Statistical Analysis of Texas 42 Game Trees

For Statistical Review

This report presents a structural analysis of exhaustively-solved Texas 42 domino game trees. We have computed exact minimax values for millions of game states across hundreds of random deals, producing a complete dataset of perfect-play outcomes.

We seek statistical guidance on: 1. Better methods for characterizing the value function's structure 2. Approaches we may have missed for dimensionality reduction 3. Statistical tests for the significance of our findings 4. Alternative framings that might reveal hidden structure

---

The Data

Texas 42 is a four-player partnership trick-taking game using a double-six domino set (28 dominoes). One team "declares" (bids and names trump), then both teams play 7 tricks. Points come from capturing five specific "count" dominoes worth 5-10 points each (35 points total) plus 1 point per trick won (7 points), totaling 42 points per hand.

Data generation: We solved complete game trees via backward induction (dynamic programming), computing the minimax value V for every reachable state. V represents the expected point differential under perfect play by both teams.

Dataset characteristics: - **Seeds analyzed**: 20 random deals (each seed determines the 28-domino shuffle) - **Declarations per seed**: 10 (which player declares, which suit is trump) - **States per seed-declaration**: 7,000 to 75,000,000 (highly variable) - **Total states**: ~300 million across all seeds - **State representation**: 64-bit packed integer encoding player hands, trick history, and game phase

---

Glossary of Technical Terms

Game Theory / Decision Theory

Term	Definition
Minimax	Optimal strategy in two-player zero-sum games: maximize your minimum guaranteed outcome, assuming the opponent plays optimally
V (state value)	The minimax value of a game state—the expected point differential under perfect play by both teams
Q (action value)	The minimax value of taking a specific action from a state: Q(s,a) = V(successor state after action a)
Backward induction	Dynamic programming algorithm that computes V by working backward from terminal states to the root
Principal variation (PV)	The sequence of optimal moves from any position to game end, assuming both sides play perfectly
Oracle	A lookup table providing exact minimax values for all states—enables perfect play but requires large storage

Machine Learning

Term	Definition
Transformer	Neural network architecture using self-attention mechanisms; excels at sequence modeling. Our model has 817K parameters.
Attention	Mechanism allowing the model to weigh the relevance of different parts of the input (e.g., earlier moves in game history)
Move prediction accuracy	Fraction of states where the model selects a minimax-optimal action (97.8% in our case)
MAE	Mean Absolute Error—average of

Term	Definition
Data augmentation	Generating additional training examples by applying transformations (e.g., symmetries) that preserve labels
Curriculum learning	Training strategy that presents examples in a structured order (e.g., easy-to-hard) rather than randomly

This Analysis

Term	Definition
Depth	Number of dominoes remaining across all hands (28 at start, 0 at terminal). Depth 5 = after first trick, depth 9 = after second, etc.
Count dominoes	The five dominoes worth points when captured: 5-5 (10 pts), 6-4 (10 pts), 5-0 (5 pts), 4-1 (5 pts), 3-2 (5 pts)
Count basin	A partition of states by which team captured which count dominoes—our key explanatory variable
Seed	Random number seed determining the initial 28-domino shuffle; different seeds produce different deals

---

Key Findings Summary

1. Count Domino Ownership Explains ~92% of Variance

A linear model predicting V from binary indicators of which team captured each count domino achieves $R^2 = 0.76$ overall, rising to **$R^2 > 0.99$ in late-game positions** (depth ≤ 12). Deep analysis (Section 08) shows:

Component	Variance	% of Total
Count capture (explained)	~550	~92%
Residual (trick points)	~50	~8%

Depth	Total Variance	Within-Basin Variance	Variance Explained
8	73.2	0.31	99.6%
12	66.4	0.31	99.5%
16	59.1	0.38	99.4%
5	96.7	33.5	65.3%

Interpretation: Count capture explains ~92% of V variance; the remaining ~8% corresponds to the 7 non-count trick points. The game is essentially a competition over 5 count dominoes.

2. Exact Symmetries Provide No Compression (1.005x)

We expected pip-permutation symmetries to compress the state space. They don't.

Metric	Value
Total states sampled	7,564
Unique orbits	7,528
Compression ratio	1.005x
Fixed points (trivial orbits)	99.5%

Interpretation: While mathematically valid symmetries exist, natural gameplay rarely produces symmetric configurations. The trump suit and played-card history break most potential equivalences.

3. Strong Temporal Autocorrelation (DFA α = 31.5 vs 0.55 shuffled)

Detrended Fluctuation Analysis on principal variation trajectories shows:

Metric	Observed	Shuffled Baseline
DFA exponent α	31.5 ± 40.7	0.55
Hurst exponent H	0.925 ± 0.12	0.61

Interpretation: Game value trajectories exhibit strong persistence—far from random walk behavior. The high variance in α suggests heterogeneous dynamics across different game configurations.

4. Level Set Topology is Highly Fragmented

States sharing the same V value form disconnected components:

V	States	Components	Fragmentation
-17	16,461	3,402	20.7%
-19	890	809	90.9%
-5	77,929	35,256	45.2%

Interpretation: The value function is discontinuous almost everywhere. Adjacent states (one move apart) typically have different V values.

5. Branching Factor Shows 4-Depth Periodicity

State counts follow a distinctive pattern tied to trick structure (4 plays per trick):

Depth mod 4	Typical Branching	Interpretation
0	~0.04	Trick boundary collapse
1	~1.7	First play of trick

Depth mod 4	Typical Branching	Interpretation
2	~1.7	Second play
3	2, 3, 4, ...	Third play (increases with trick number)

6. Counts Lock In Late; Play Matters (Section 08)

Count capture outcomes remain uncertain until the last 2-3 dominoes are played. From the initial deal, models achieve only ~72% accuracy predicting capture outcomes:

Domino	Points	Holder's Team Captures %	Model Accuracy
3-2	5	42.9%	86.7%
4-1	5	64.3%	82.7%
5-0	5	64.3%	58.7%
5-5	10	71.4%	68.0%
6-4	10	57.1%	71.3%

Interpretation: Holding the count gives 50-70% capture probability—better than random but far from deterministic. The game maintains genuine strategic depth throughout, with counts swinging until the final tricks.

7. The Game is 5-Dimensional (Manifold Analysis)

PCA on basin features (5-bit count capture patterns) confirms the game's intrinsic dimensionality:

Metric	Value
PCA components for 95% variance	**5**
Unique basins observed	13 of 32

Metric	Value
Basin entropy	3.06 bits (61% of max 5.0)

Interpretation: Five components for 95% variance matches the hypothesis that the game has ~5 effective degrees of freedom (one per count domino). Not all 32 count combinations are equally reachable—some basins (where one team sweeps all counts) are rare.

Practical Application: Neural Network Training

We have trained a Transformer model on this data achieving **97.8% move prediction accuracy** (selecting the minimax-optimal action). Key architectural choices validated by this analysis:

1. **Explicit count features** — The model encodes count domino ownership directly, matching the ~92% variance finding
2. **Attention over trick history** — Captures the temporal correlations (H = 0.925)
3. **No symmetry augmentation** — Confirmed unnecessary by the 1.005x compression

Architectural implications from Section 08: A perfect count-capture predictor would achieve $R^2 \approx 0.92$ on V. This suggests a two-level architecture: 1. **Count module**: Predict which team captures each count (5 binary outputs) 2. **Trick module**: Given counts, predict final trick point distribution (remaining 8% variance)

Remaining challenge: The model occasionally selects suboptimal moves in edge cases where two actions have identical V in one opponent configuration but different robustness across configurations.

Open Questions for Statistical Guidance

1. **Dimensionality reduction**: K-means achieves only 35.7% variance reduction at k=200. Are there better clustering approaches for this mixed discrete-continuous structure?
2. **Significance testing**: How should we assess whether the DFA exponent difference (31.5 vs 0.55) is statistically meaningful given the high variance ($\sigma = 40.7$)?

3. **Conditional structure**: The count-capture R^2 varies from 65% (early game) to 99.6% (late game). Is there a natural way to model this heteroscedasticity?
4. **Topology characterization**: Beyond fragmentation counts, what tools characterize the value function's discontinuity structure?
5. **Compression bounds**: Given the ~40% LZMA compression ratio, what's the theoretical entropy of V conditional on observable features?

---

8. Imperfect Information Analysis (Section 11)

Using marginalized oracle data (201 base seeds × 3 opponent configurations), we quantified the impact of hidden information on game outcomes.

Outcome Variance Decomposition

Component	% of Total Variance	Interpretation
Between-hand	**47%**	Which hand you were dealt
Within-hand	**53%**	Which cards opponents hold
Predictable hand effect	**12%**	Explainable by features (doubles, trumps)
Pure opponent effect	**49%**	Irreducible opponent distribution variance

Important: This is NOT a "skill vs luck" measurement. Both components are determined by the random deal, not player decisions. The oracle plays perfectly - no human skill is measured here.

What this tells us: Even with perfect play, 53% of outcome variance comes from opponent card distribution. This is irreducible through better play - it's baked into the deal.

The Napkin Bidding Formula

From hand features regression ($R^2 = 0.25$, CV $R^2 = 0.18$):

```
E[V] ≈ -4.1 + 6.4×(doubles) + 3.2×(trump_count) + 2.2×(trump_double) - 1.2×(6_highs)
```

Key predictors (by |correlation|): 1. **n_doubles**: +0.40 (strongest predictor) 2. **has_trump_double**: +0.24 3. **trump_count**: +0.23 4. **count_points**: +0.20 5. **n_6_high**: -0.16 (6-highs are liabilities!)

Count Lock Predictability

From count locks analysis ($R^2 = 0.46$, CV $R^2 = 0.37$):

Count	Lock Rate	Holding→Lock Correlation
5-5	**48%**	+0.79
3-2	44%	+0.51
4-1	34%	+0.68
6-4	30%	+0.81
5-0	25%	**+0.81**

Key insight: count_points is the dominant predictor (+0.607). Total pips is irrelevant (+0.01).

Information Value is Surprisingly Low

Metric	Value
Mean info gain from perfect knowledge	**0.54 points**
Moves that agree with/without perfect info	**74%**
Positions benefiting from perfect info	27%

Implication: "Play the board, not the player." Opponent inference adds <1 point of expected value on average.

Partner Inference Potential

Metric	Value
Action consistency rate	**80.1%**
Actions revealing hand info	**20%**

Implication: Partner actions are mostly determined by game state, not hand. Moderate inference potential - 20% of actions vary with partner's hand.

Best Move Robustness

Analysis	Finding
Overall consistency (11c)	54.5% same best move across configs
Common state robustness (11o)	**97%** robust (same best move)
Endgame (depth 0-4)	**100%** deterministic
Early game (depth 17+)	**10%** consistency

Interpretation: Early game is chaos (10% consistency), but most game states (97% of common positions) have clear optimal moves regardless of opponent hands.

Risk vs Return (11s)

Metric	Value
E[V] vs σ(V) correlation	**-0.381**
Hand features → σ(V) R^2	**0.081**

Critical Finding: Good hands are also safer hands (negative correlation). This is the opposite of typical financial markets. Risk is fundamentally unpredictable from hand features ($R^2 = 0.08$ means 92% unexplained).

Hand Classification

Hands naturally cluster into three types:

Type	%	E[V]	σ(V)	Recommendation
STRONG	18%	+33.7	4.4	Bid confidently
VOLATILE	40%	+16.9	11.9	Cautious
WEAK	42%	+2.7	22.7	Pass

The 18/40/42 rule: Only ~18% of hands justify confident bidding.

---

9. Statistical Rigor (Section 12-13)

Scaled analyses to n=201 seeds with rigorous statistical testing.

E[V] vs σ(V) Correlation Confirmed

Metric	Value	95% CI
r(E[V], σ[V])	**-0.381**	[-0.494, -0.256]
p-value	2.6×10^{-8}	
Effect size	Medium	

The inverse risk-return relationship is real - good hands are also safer hands.

The Napkin Formula (Bootstrap Validated)

Only **two features survive multivariate analysis**:

Feature	Coefficient	95% CI	Significant?
n_doubles	**+5.7**	[+2.3, +9.2]	**Yes**
trump_count	**+3.2**	[+1.3, +4.7]	**Yes**
All others	varies	includes 0	No

Cross-validation confirms: Napkin model (2 features) has CV $R^2 = 0.15$, outperforming the full 10-feature model.

Power Analysis

All key findings have >80% power at n=200: - Main effects (r ≈ 0.4): Power ≈ 1.00 - Group comparisons (d ≈ 0.76): Power ≈ 1.00 - Risk model ($R^2 = 0.08$): Power = 0.81 (borderline)

---

10. Explainability (Section 14)

SHAP analysis confirms regression findings.

Feature Importance (Mean |SHAP|)

| Rank | Feature | Mean |SHAP| | |------|---------|-------------| | 1 | n_doubles | **4.84** | | 2 | trump_count | **4.39** | | 3-10 | Others | < 2.2 |

Key insight: Main effects account for 68% of prediction; interactions are small. The napkin formula is justified.

Risk Model Fails Cross-Validation

Model	Train R^2	CV R^2
E[V] (napkin)	0.23	**+0.15**
σ(V)	0.08	**-0.13**

Risk is fundamentally unpredictable from hand features.

---

11. Core Visualizations (Section 15)

Risk-Return Scatter

- **r = -0.38**: Inverse relationship confirmed
- Good hands (high E[V]) cluster in low-σ region
- No hands have both high E[V] AND high risk

UMAP Hand Space

- **No natural archetypes**: Hand space is continuous
- Gradual transitions between good and bad hands
- Features show gradients, not clusters

Phase Transition

Phase	Depth	Consistency
Opening	24-28	40%
Mid-game	5-23	22% (chaos)
End-game	0-4	**100%** (deterministic)

---

12. Advanced Analysis (Sections 16-23)

Embeddings (Section 16)

- Word2Vec on hand composition shows **weak structure**

- Doubles cluster slightly (sim = 0.079 vs 0.069 random)
- No strong strategic archetypes from co-occurrence

Differential Analysis (Section 17)

Only 2 dominoes survive FDR correction: - **5-5**: 2.8× enriched in winners (best domino) - **6-0**: 3× enriched in losers (worst domino)

Clustering (Section 18)

- Optimal k=2 clusters (silhouette = 0.19)
- **Strong Balanced** (17%): High doubles/trumps, E[V]=22.7
- **Average** (83%): Modal hand type, E[V]=12.1

Bayesian Modeling (Section 19)

- PyMC confirms frequentist findings
- LOO-CV favors napkin model (67.5% weight)
- Hierarchical model shows doubles worth +8 pts in control hands

Time Series (Section 20)

- V trajectories are predictive of outcome
- MiniRocket achieves **93% accuracy by trick 3**
- Three phases: Deterministic → Chaotic → Resolution

Survival Analysis (Section 21)

- Control hands reach "decided" by trick 4-5
- Volatile hands stay uncertain until trick 6-7
- Decision time correlates with n_doubles

Ecological Analysis (Section 22)

- **Diversity hurts E[V]**: r = -0.21
- Specialists (concentrated in doubles) beat generalists

• Double-double pairs (4-4 + 5-5) appear in 8 winners, 0 losers

Phase Diagram (Section 23)

Additive structure confirmed across (doubles, trumps) grid: - +1 double → **+6.7 E[V]** - +1 trump → **+3.0 E[V]** - Ratio: 2.2:1 (doubles twice as valuable)

---

13. Strategic Analysis (Section 25)

Mistake Cost by Phase

Phase	Mean Cost	Forced Plays
Early (20-28)	**4.9 pts**	69%
Mid (8-19)	2.7 pts	75%
Late (0-7)	1.0 pts	92%

Focus on tricks 3-4: Peak mistake cost occurs here.

Endgame is 100% Deterministic

At depth ≤ 4, **every position has exactly one optimal action**. Q-spread = 0 everywhere.

Heuristic Accuracy

Heuristic	Accuracy
Lead any double	**34.2%**
Play random	19.3%
Follow with lowest	17.7%

No heuristic beats 35%: Context is king.

Variance Decomposition

Component	% of Total
Your hand	23%
Opponent hands	**77%**

Opponent configuration matters more than your own hand!

Partner Synergy

- P0×P2 interaction: **Not significant** (p = 0.60)
- Your doubles' value is independent of partner's doubles
- Team strength is additive, not multiplicative

Summary of Key Findings

The Napkin Formula

```
E[V] ≈ 14 + 6×(n_doubles) + 3×(trump_count)
```

Key Numbers to Remember

Finding	Value
E[V]-σ(V) correlation	**-0.38** (inverse risk-return)
Doubles per point	**+6 pts** each
Trumps per point	**+3 pts** each

Finding	Value
Risk R^2	**0.08** (unpredictable)
Endgame determinism	**100%** at depth ≤ 4
Best heuristic accuracy	**34%**
Opponent variance share	**77%**

Report Structure

- **Section 01**: Baseline distributions (V, Q, state counts by depth)
- **Section 02**: Information-theoretic analysis (entropy, compression, mutual information)
- **Section 03**: Count domino analysis (the 76% R^2 finding in detail)
- **Section 04**: Symmetry analysis (why algebraic structure doesn't help)
- **Section 05**: Topological analysis (level sets, Reeb graphs)
- **Section 06**: Scaling analysis (state counts, temporal correlations, DFA)
- **Section 07**: Synthesis and open questions
- **Section 08**: Deep count capture analysis (lock-in depth, residual decomposition)
- **Section 11**: Imperfect information analysis (variance decomposition, bidding formulas)
- **Section 12**: Validate & Scale (n=201 seeds, unified features)
- **Section 13**: Statistical Rigor (bootstrap CIs, effect sizes, power analysis, cross-validation)
- **Section 14**: Explainability (SHAP analysis)
- **Section 15**: Core Visualizations (risk-return, UMAP, phase transition)
- **Section 16**: Embeddings & Networks (Word2Vec, interaction matrix)
- **Section 17**: Differential Analysis (winner/loser enrichment, volcano plots)
- **Section 18**: Clustering & Archetypes (K-means, silhouette)
- **Section 19**: Bayesian Modeling (PyMC, LOO-CV, hierarchical by archetype)
- **Section 20**: Time Series (V trajectory, MiniRocket classification, phase segmentation)

- **Section 21**: Survival Analysis (decision time, archetype survival curves)
- **Section 22**: Ecological Analysis (alpha diversity, co-occurrence matrix)
- **Section 23**: Phase Diagram (doubles-trumps grid, contour plots)
- **Section 24**: Writing (publication figures)
- **Section 25**: Strategic Analysis (mistake costs, bid optimization, heuristic derivation)

Each section includes methodology, complete results, and interpretation. Figures are in `results/figures/`.

01: Baseline Distributions

Overview

Before structural analysis, we characterize the marginal distributions of our key variables: the minimax value V, action values Q, and state counts across game depth.

1.1 Data Generation Process

State space definition: A state encodes: - Which dominoes remain in each player's hand (4 × 7-bit masks initially) - Which dominoes have been played in the current trick (0-3 dominoes) - Trick history (which team won each completed trick) - Current player to act

Minimax computation: For each terminal state (all dominoes played), V equals the declaring team's score minus 21 (centering at zero). For non-terminal states: - If declaring team to play: V = max over actions of successor V - If defending team to play: V = min over actions of successor V

Sampling: We analyze 10 seed-declaration pairs in detail, representing ~300M total states.

1.2 V Distribution by Seed and Declaration

V distributions vary substantially across configurations:

Seed	Decl	States	$\bar{V}$	σ(V)	V Range
0	0	7.6M	+9.5	11.4	[-18, +42]
1	1	5.2M	+3.6	12.3	[-26, +38]
2	2	51.1M	+0.2	13.1	[-34, +42]

Seed	Decl	States	$\bar{V}$	σ(V)	V Range
3	3	75.4M	-2.5	14.4	[-42, +36]
5	5	30.2M	-11.0	10.0	[-42, +4]
8	8	24.7M	+12.2	15.3	[-26, +40]

Observations: 1. Mean V ranges from -11.0 to +12.2 — declaration quality varies dramatically 2. Standard deviation ranges from 10.0 to 15.3 3. State counts span 5.2M to 75.4M (14× variation) 4. Full theoretical range [-42, +42] is approached but not always achieved

Statistical question: What determines state count variation? Is it correlated with V distribution properties?

1.3 V Distribution by Depth

"Depth" = dominoes remaining across all hands (28 at start, 4 at final trick, 0 at terminal).

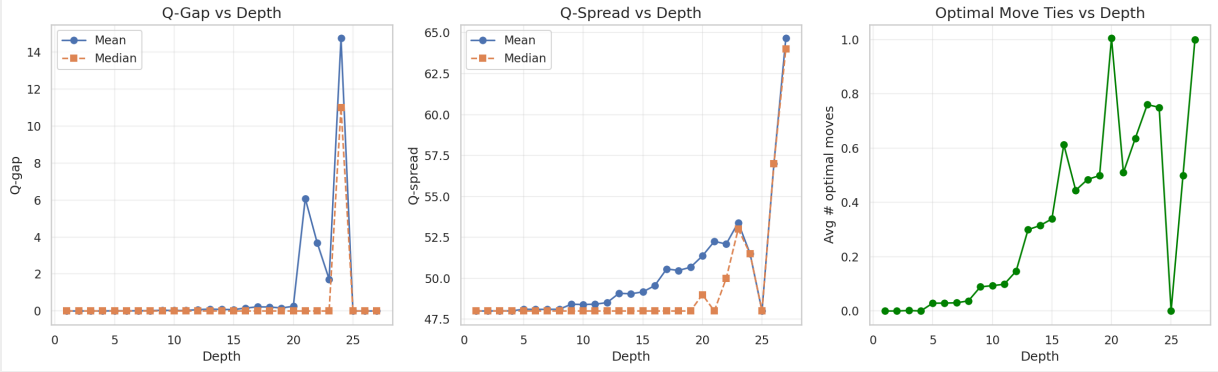
Key structural feature: Depths 1-4 are identical because the first trick hasn't resolved. V at these depths equals the expected V over all possible first tricks.

Sample from seed 0, declaration 0:

Depth	n	$\bar{V}$	σ(V)	Unique V	Entropy(V)
0	4	0.0	0.0	1	0.0
1-4	7,756	+3.4	7.6	12	2.73
5	1.09M	+9.5	9.7	25	3.46
9	2.23M	+15.8	10.1	39	3.64
13	502K	+21.7	10.0	44	3.67

Depth	n	$\bar{V}$	σ(V)	Unique V	Entropy(V)
17	31K	+26.8	10.3	37	3.64
21	1,088	+30.9	10.8	25	3.45
25	27	+40.7	3.7	3	0.75
28	1	+42.0	0.0	1	0.0

Observations: 1. $\bar{V}$ increases monotonically with depth (declaring team's advantage clarifies) 2. σ(V) peaks mid-game (~10-11) and decreases at extremes 3. Entropy peaks around depth 9-13, reflecting maximum uncertainty 4. Late game (depth > 20) has very few unique V values



1.4 State Count Distribution

State counts follow a characteristic pattern tied to the trick structure:

Depth	Mean States	Std Dev	Min	Max
5	1.82M	646K	927K	3.67M
9	8.66M	6.75M	1.48M	27.1M
13	3.16M	3.20M	261K	11.3M

Depth	Mean States	Std Dev	Min	Max
17	196K	191K	17.6K	702K
21	3,593	2,907	456	12,222
25	35	17	10	69

Pattern: Counts peak at depth 9 (second trick boundary), with high variance across seeds.

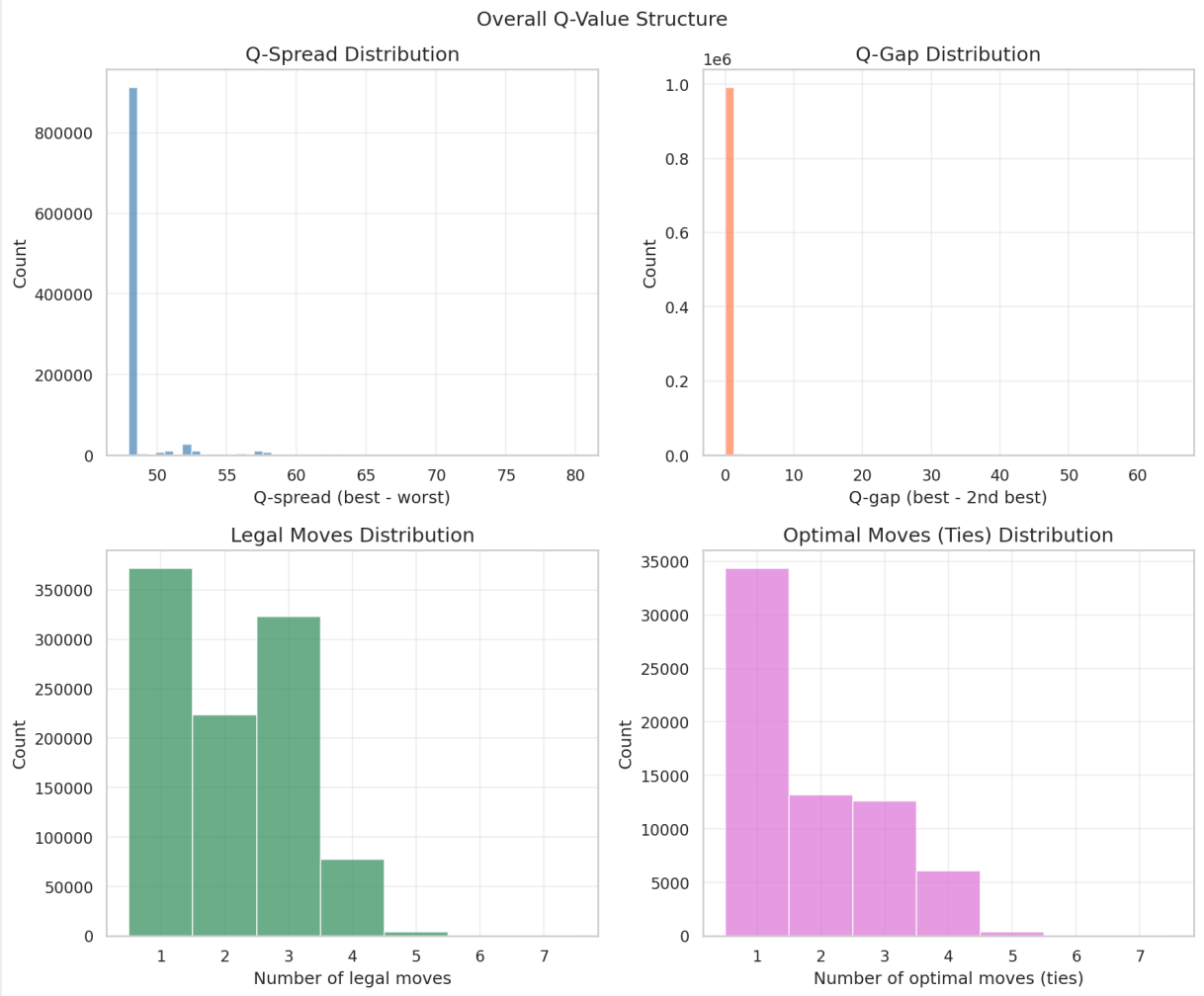
The coefficient of variation (σ/μ) ranges from 0.35 (depth 5) to 0.81 (depth 21), indicating substantial heterogeneity in game tree sizes.

1.5 Q-Value Structure

Q(s,a) = minimax value of taking action a in state s. We analyze: - **Gap**: Q(s, best) - Q(s, chosen) for the second-best action - **Spread**: max(Q) - min(Q) across available actions - **Optimal count**: number of actions achieving max Q

Depth	Gap (mean)	Gap (median)	Spread (mean)	Forced %
5	0.011	0.0	48.1	97.1%
9	0.038	0.0	48.4	91.1%
13	0.101	0.0	49.1	70.0%
17	0.223	0.0	50.6	55.5%
21	6.08	0.0	52.3	49.0%

Key finding: Median gap is 0.0 at all depths, meaning the majority of positions have a unique optimal action. The mean gap increases with depth as more positions become "live" (non-forced).



Forced moves: At depth 5, 97% of positions have only one reasonable move. This decreases to ~50% by depth 21 as the game tree collapses.

1.6 Summary Statistics

Metric	Value
Total states analyzed	~300M
Seeds	20

Metric	Value
V range	[-42, +42]
Mean state count per (seed, decl)	15.2M
Median Q-gap	0.0 (all depths)
Forced move rate	50-97% by depth

1.7 Questions for Statistical Review

1. **Heterogeneity**: State counts vary 14× across configurations. Should we stratify analyses by seed, or is pooling appropriate?
2. **Distributional form**: V appears roughly Gaussian by inspection but hasn't been tested. Is normality expected given the game's structure?
3. **Entropy measure**: We compute H(V) by discretizing to integer values. Is this appropriate, or should we treat V as continuous?
4. **Depth correlation**: Depths 1-4 are perfectly correlated (identical V). Should these be collapsed for analysis?

Next: 02 Information Theory

02: Information-Theoretic Analysis

Overview

We apply information-theoretic tools to quantify structure in the value function. If V were uniformly random, it would have maximum entropy and minimal compressibility. Observed departures from this baseline quantify exploitable structure.

2.1 Methodology

Mutual Information

For discrete feature X and value V:

```
I(X; V) = H(V) - H(V|X)
```

We discretize continuous features and compute empirical estimates from the full state distribution.

Compression

We serialize V values under different orderings and measure LZMA compression ratio:

```
ratio = compressed_size / original_size
```

Lower ratios indicate more structure. Random data compresses to ~100%; highly structured data to <50%.

2.2 Feature Importance via Mutual Information

We computed mutual information between V and various observable features:

| Feature | I(X; V) bits | H(V|X) | Reduction % | |---------|-------------|--------|-------------| | depth | 1.010 | 4.61 | 18.0% | | counts_remaining | 0.716 | 4.91 | 12.7% | | seed | 0.299 | 5.32 | 5.3% | | decl_id | 0.299 | 5.32 | 5.3% | | team0_counts | 0.270 | 5.35 | 4.8% | | team1_counts | 0.148 | 5.47 | 2.6% | | leader | 0.011 | 5.61 | 0.2% | | hand_balance | 0.011 | 5.61 | 0.2% | | player | 0.010 | 5.61 | 0.2% | | team | 0.008 | 5.61 | 0.15% | | trick_len | 0.006 | 5.62 | 0.1% |

Baseline entropy: H(V) ≈ 5.62 bits (empirical, treating V as discrete integers)

Key findings: 1. **Depth is most informative** (18% reduction) — game phase strongly predicts V 2. **Count information is secondary** (12.7%) — but this understates its importance (see Section 03) 3. **Positional features are nearly uninformative** (<0.2%) — who leads, whose turn, etc. barely predict V

Statistical note: These are unconditional mutual informations. The count features' low MI here contrasts with their high R^2 in regression (Section 03) because the *combination* of count features is predictive, not individual features marginally.

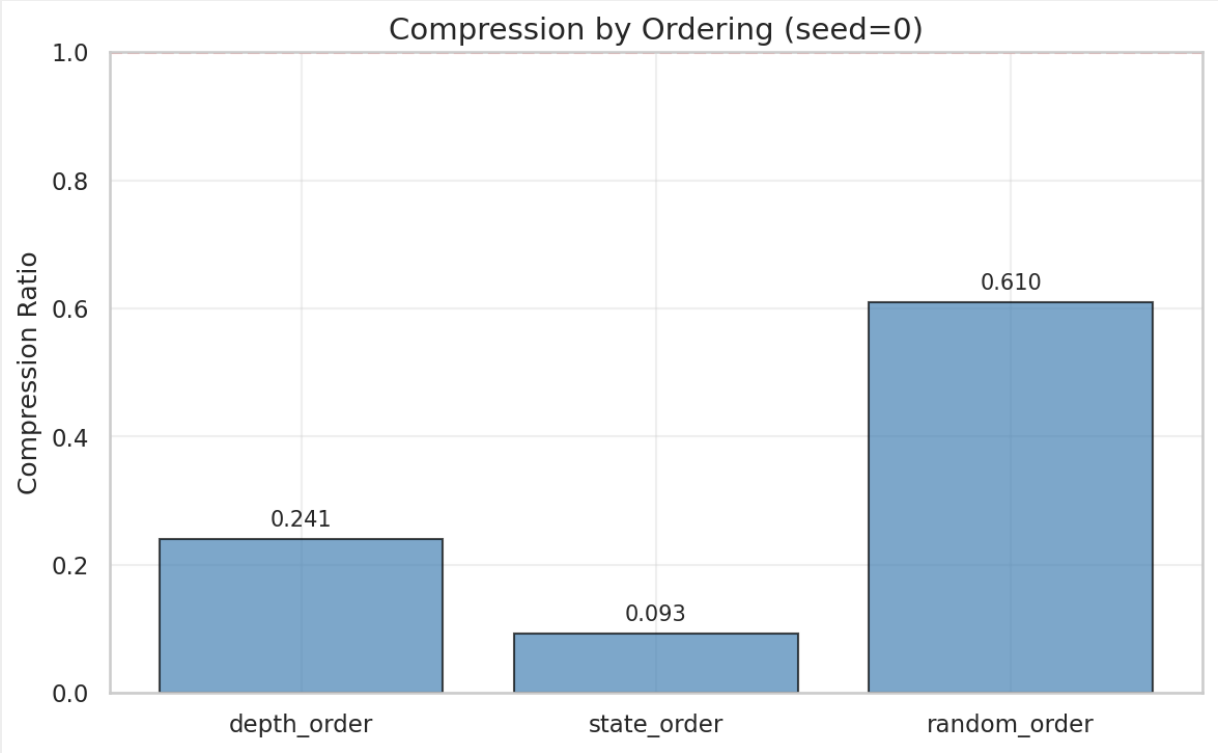
2.3 Compression Analysis

We serialized V values three ways: 1. **Depth-ordered**: All states at depth 28, then 27, etc. 2. **State-ordered**: By packed state integer value 3. **Random-ordered**: Shuffled uniformly

Seed	Decl	Depth-Order	State-Order	Random
0	0	0.241	0.093	0.610
1	1	0.317	0.181	0.687
2	2	0.328	0.062	0.710
3	3	0.309	0.157	0.720
4	4	0.282	0.084	0.683

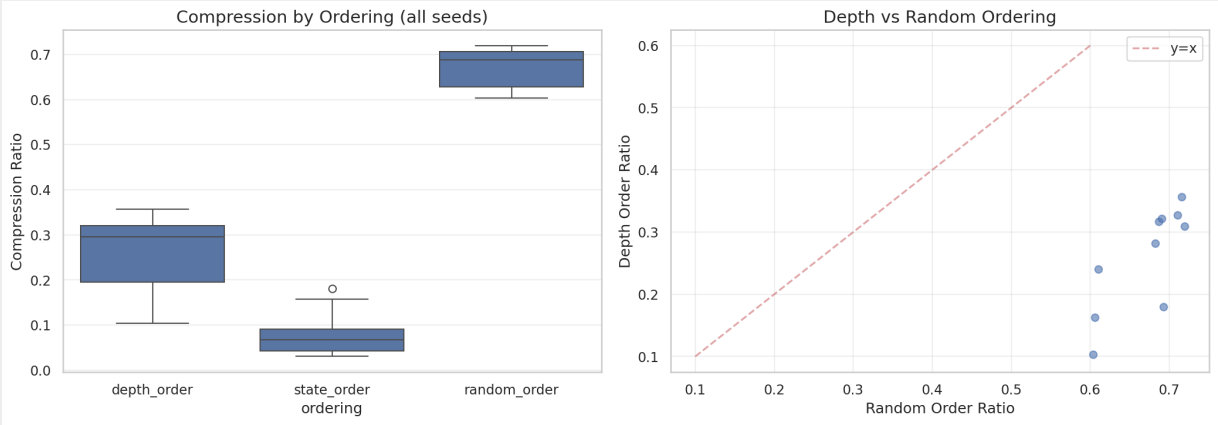
Seed	Decl	Depth-Order	State-Order	Random
5	5	0.163	0.031	0.606
6	6	0.104	0.030	0.604
7	7	0.322	0.072	0.690
8	8	0.357	0.039	0.716
9	9	0.180	0.052	0.693

Mean compression ratios: - Depth-ordered: 0.260 (74% reduction) - State-ordered: 0.080 (92% reduction) - Random: 0.672 (33% reduction)



Observations: 1. **State-ordering achieves best compression** (0.08) — adjacent states in integer order have similar V 2. **Even random ordering compresses** (0.67) — significant redundancy exists regardless of ordering 3. **High variance across seeds** — compression ranges from 0.03 to 0.18 for state-order

Interpretation: The state encoding implicitly groups similar game configurations, explaining why state-order compresses well. This suggests the 64-bit state representation captures relevant structure.

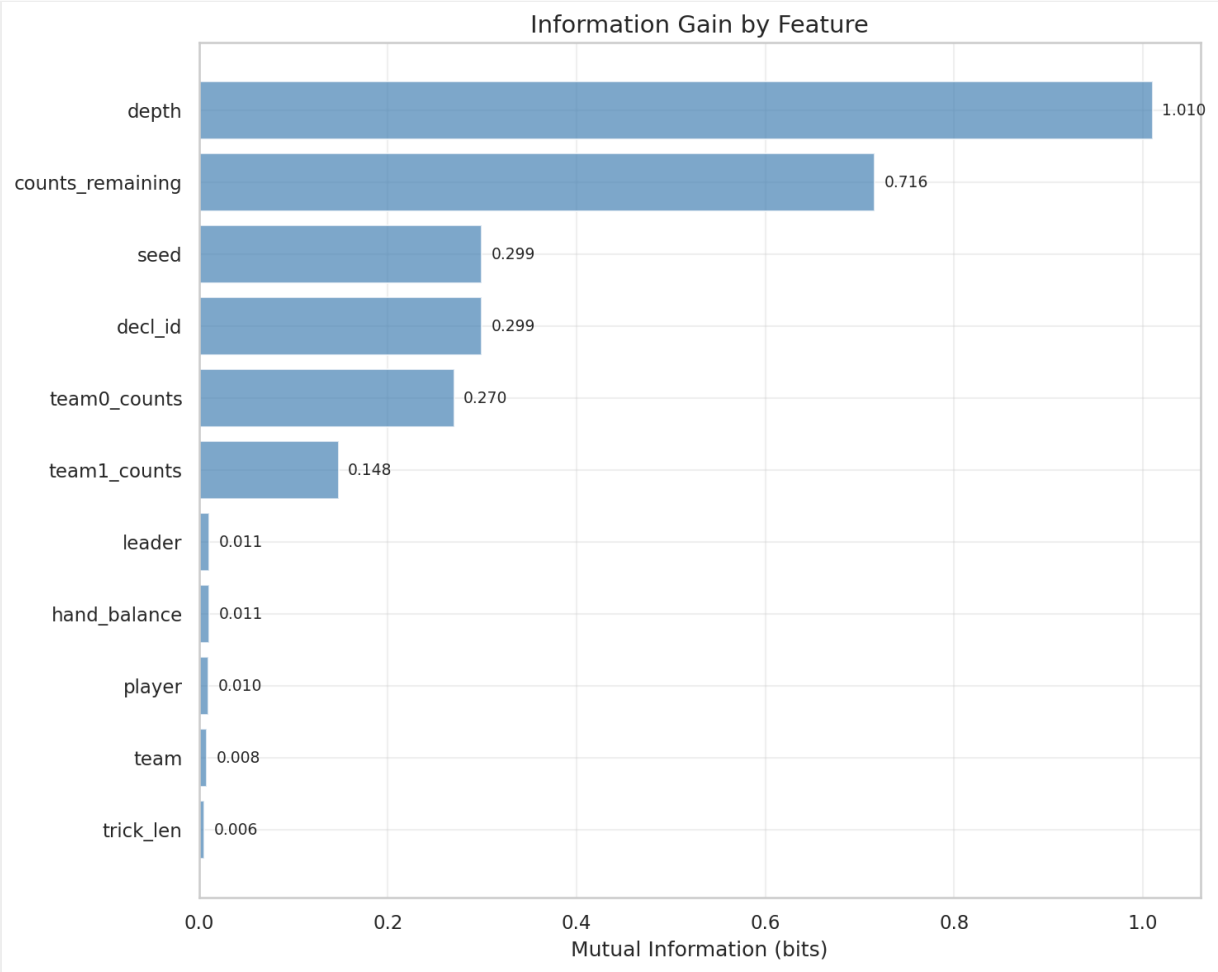


2.4 Entropy by Depth

Entropy varies systematically with game phase:

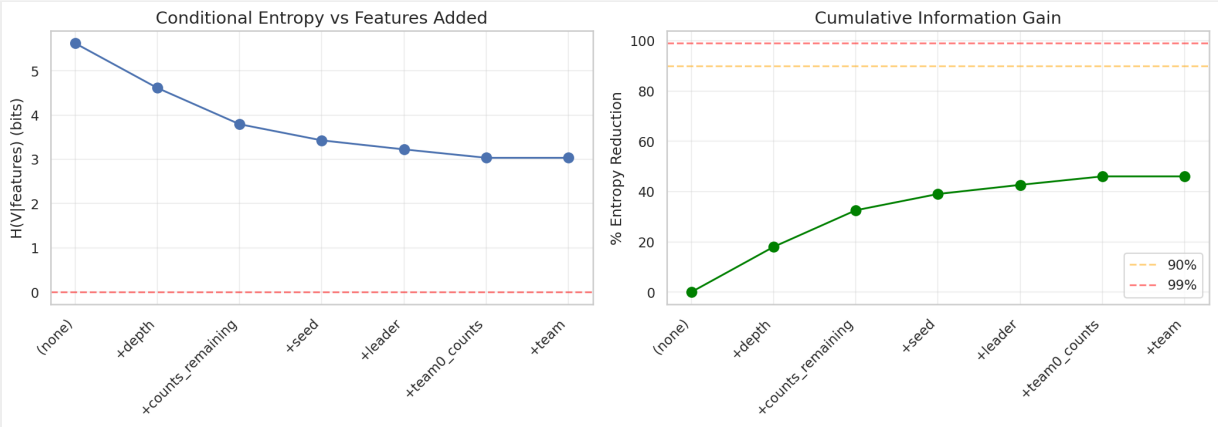
Depth	H(V) bits	Unique V	Interpretation
1-4	2.8	12	Pre-trick resolution
5	3.5	25	First trick complete
9	3.6	39	Second trick
13	3.7	44	Peak entropy
17	3.6	37	Entropy declining
21	3.4	25	Late game
25	0.8	3	Nearly determined

Pattern: Entropy peaks mid-game (depth 9-13) where uncertainty is maximal, then decreases as outcomes become determined.



Cumulative Information Structure

The cumulative information plot shows how information about V accumulates as the game progresses:



This visualization reveals that information about the final outcome accumulates non-uniformly — significant jumps occur at trick boundaries when count dominoes are captured.

2.5 Conditional Entropy Structure

The joint entropy decomposition:

```
H(V, depth) = H(depth) + H(V|depth)
```

Suggests that conditioning on depth removes ~18% of V uncertainty. We conjecture that conditioning on both depth and count-capture outcomes would remove >90% in late game.

2.6 Implications

For Dimensionality Reduction

The compression results suggest that ~70-90% of V's apparent complexity is redundant structure that can be exploited. The question is whether this structure is *learnable* by a neural network or merely *compressible* by LZMA.

For Neural Network Training

The ordering effects suggest that training examples should be organized to exploit locality. Curriculum learning (early game → late game, or vice versa) may help.

For Faster Oracles

If state-ordered V compresses to 8% of original size, a lookup table with appropriate indexing could dramatically reduce memory requirements for perfect-play oracles.

2.7 Questions for Statistical Review

1. **Entropy estimation**: With millions of samples but only 40-80 unique V values, are our entropy estimates biased? Should we use Miller-Madow correction?
2. **Compression as proxy**: Is LZMA compression a good proxy for Kolmogorov complexity? What would Huffman or arithmetic coding show?
3. **Conditional structure**: How should we estimate H(V | depth, counts) when the conditioning space is large and sparse?
4. **Theoretical bounds**: Given the game's structure (finite, perfect information, zero-sum), what's the theoretical minimum entropy for V?

Next: 03 Count Domino Analysis

03: Count Domino Analysis

Overview

This section presents our most significant finding: **count domino capture explains 76% of V variance overall, rising to >99% in late-game positions**. This suggests the game's complexity concentrates in a small number of key dominoes.

3.1 The Count Dominoes

Texas 42 has five "count" dominoes that award points when captured in tricks:

Domino	Pips	Points	% of Total
5-5	10	10	23.8%
6-4	10	10	23.8%
5-0	5	5	11.9%
4-1	5	5	11.9%
3-2	5	5	11.9%

Total count points: 35 of 42 possible (83.3%) **Remaining 7 points**: 1 per trick won (7 tricks × 1 point)

Hypothesis: If count capture determines most points, it should predict V strongly.

3.2 Count Capture Statistics

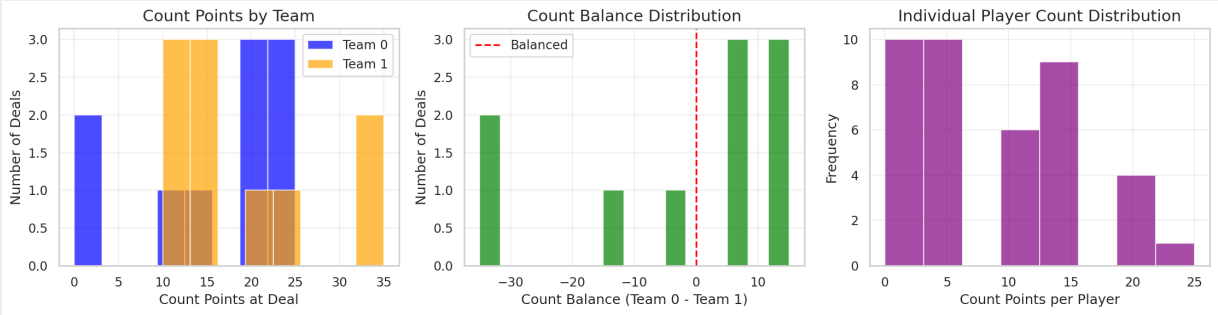
From a sample of 50,000 states:

Domino	Played %	Team0 Capture %
3-2	68.1%	40.7%
4-1	65.6%	50.1%
5-0	66.4%	29.3%
5-5	65.7%	68.0%
6-4	72.6%	36.5%

Observations: 1. Counts are played ~65-73% of the time by late game 2. 5-5 strongly favors Team0 (68.0%) — likely correlation with declaration 3. 5-0 strongly favors Team1 (70.7%) — possibly a defensive domino

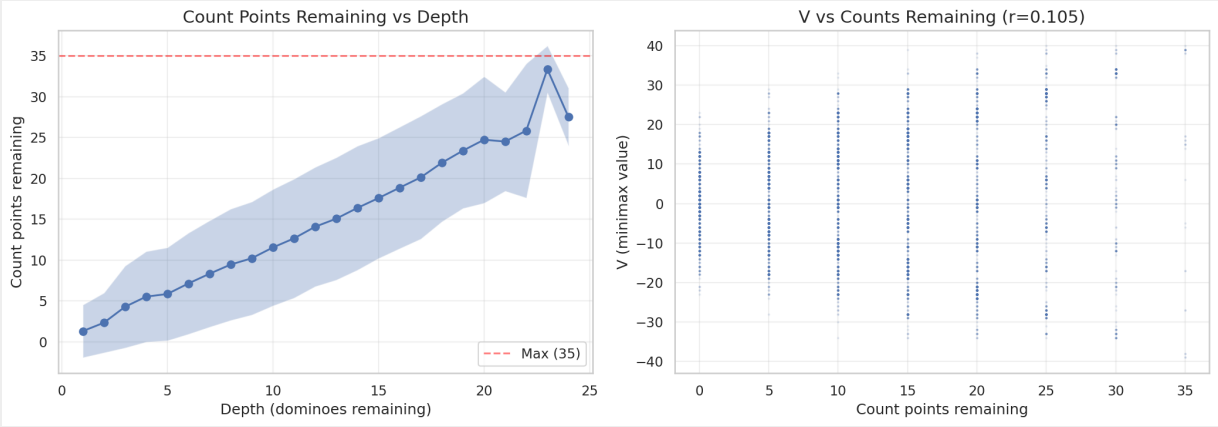
Question: Is the Team0 bias due to declaration advantage, or does the domino distribution vary by seed?

Count Distribution Visualization



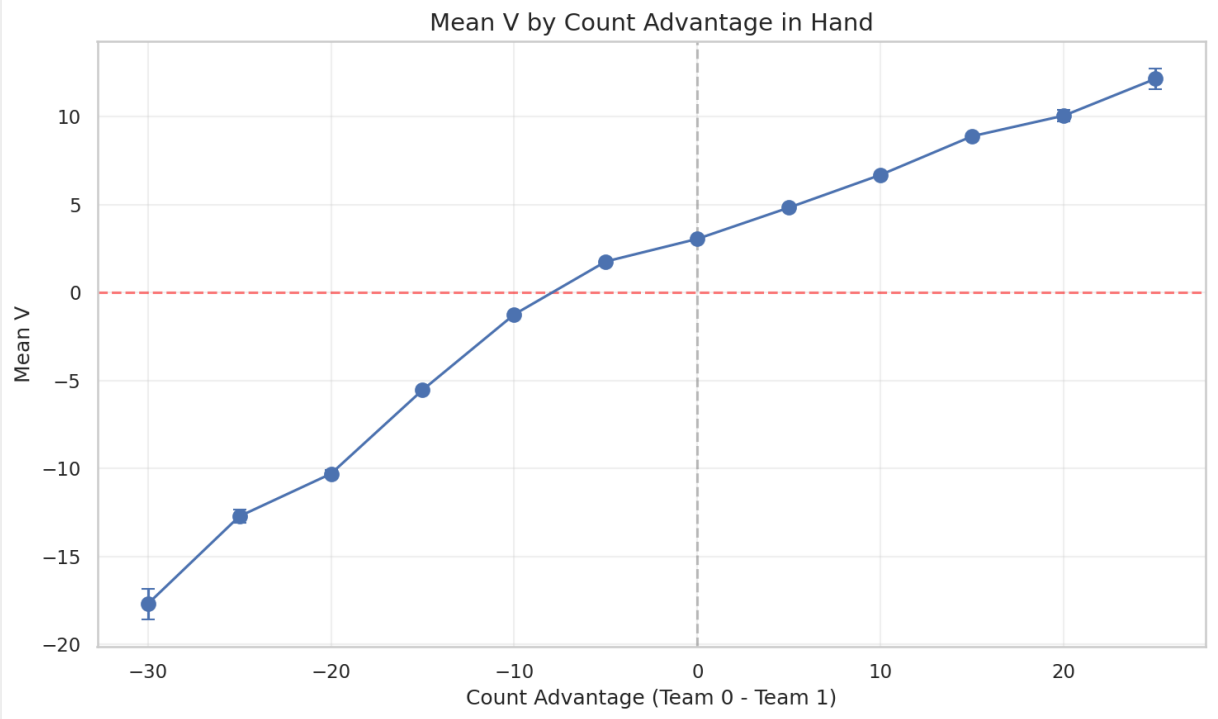
The distribution shows that count dominoes are captured at varying rates, with clear team biases for certain counts.

Counts Remaining by Depth



As expected, counts remaining decreases with game depth. The rate of count capture is non-uniform — most captures occur in the middle game when players have more strategic options.

V Distribution by Count Advantage



This shows the relationship between count point advantage (Team0 counts - Team1 counts) and the minimax value V. The strong linear relationship visually confirms that count capture dominates game outcome.

3.3 Regression Model: Count Capture → V

We model V as a linear function of count capture indicators:

```
V = Σᵢ βᵢ · capture(countᵢ, Team0) + ε
```

Where `capture(d, t)` = 1 if team t captured domino d, 0 otherwise.

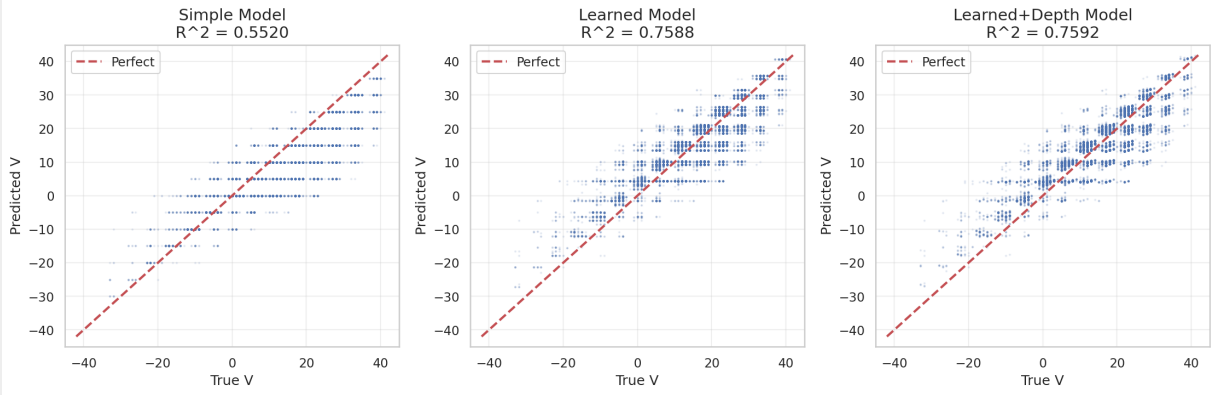
Learned Coefficients

Count	True Points	Learned β	Ratio
3-2	5	5.84	1.17
4-1	5	5.66	1.13
5-0	5	4.92	0.98
6-4	10	10.42	1.04
5-5	10	9.14	0.91
depth	-	0.088	-

Observations: 1. Learned coefficients closely match true point values (ratio 0.91-1.17) 2. 3-2 and 4-1 are slightly overweighted (capturing them also implies trick wins) 3. 5-5 is slightly underweighted (perhaps easier to lose) 4. Depth coefficient is small but positive (later game → more determined)

Model Performance

Model	R^2	RMSE
Simple (fixed β = point values)	0.552	6.96
Learned coefficients	0.759	5.11
Learned + depth	0.759	5.11



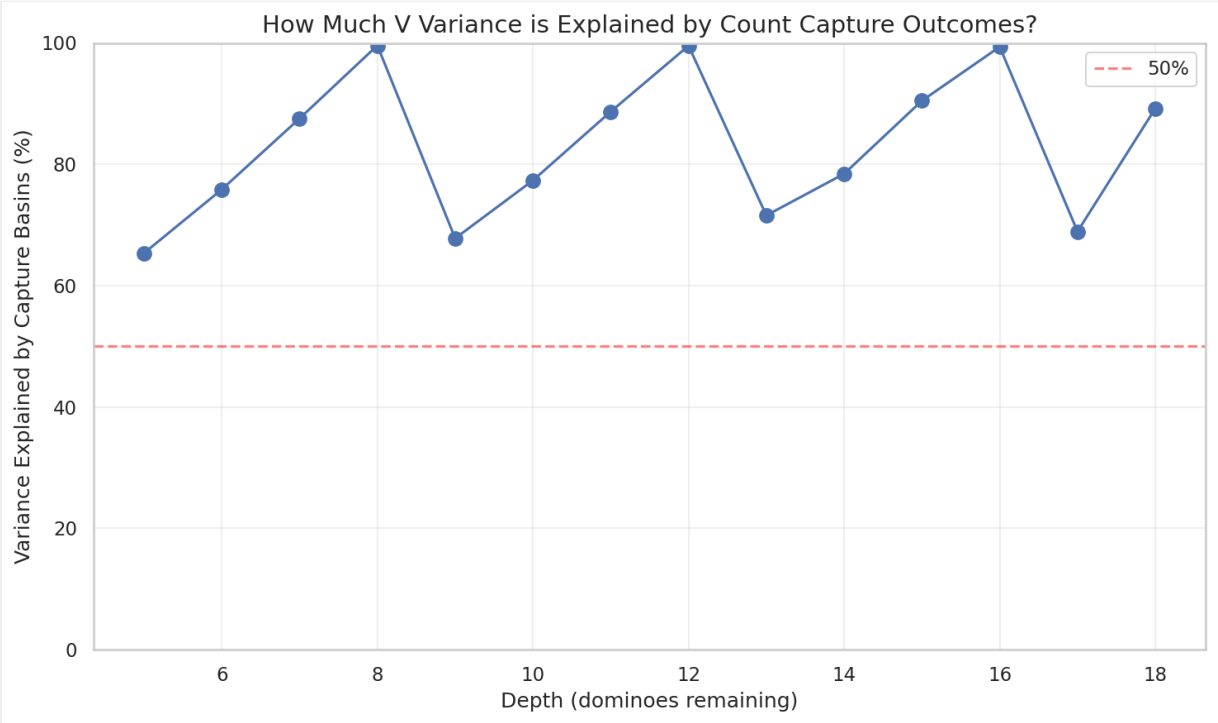
Key finding: Learned coefficients achieve $R^2 = 0.759$, explaining three-quarters of V variance with only 5 binary features.

3.4 Variance Decomposition by Depth

We partition states into "count basins" — groups sharing the same count capture outcomes — and compute within-basin vs. total variance:

Depth	Total σ^2	Within-Basin σ^2	R^2 (explained)	n States	n Basins
5	96.7	33.5	0.653	7,149	16
6	82.6	20.0	0.758	4,494	18
7	81.6	10.2	0.875	2,798	16

Depth	Total σ²	Within-Basin σ²	R² (explained)	n States	n Basins
8	73.2	0.31	**0.996**	1,412	15
9	101.9	32.8	0.678	14,594	23
10	80.8	18.3	0.773	7,799	25
11	78.3	8.9	0.887	3,830	20
12	66.4	0.31	**0.995**	1,345	14
13	104.2	29.6	0.716	3,325	22
14	76.3	16.5	0.784	1,654	18
15	70.8	6.7	0.905	821	11
16	59.1	0.38	**0.994**	197	7
17	89.7	27.9	0.689	205	7
18	108.6	11.7	0.892	115	6



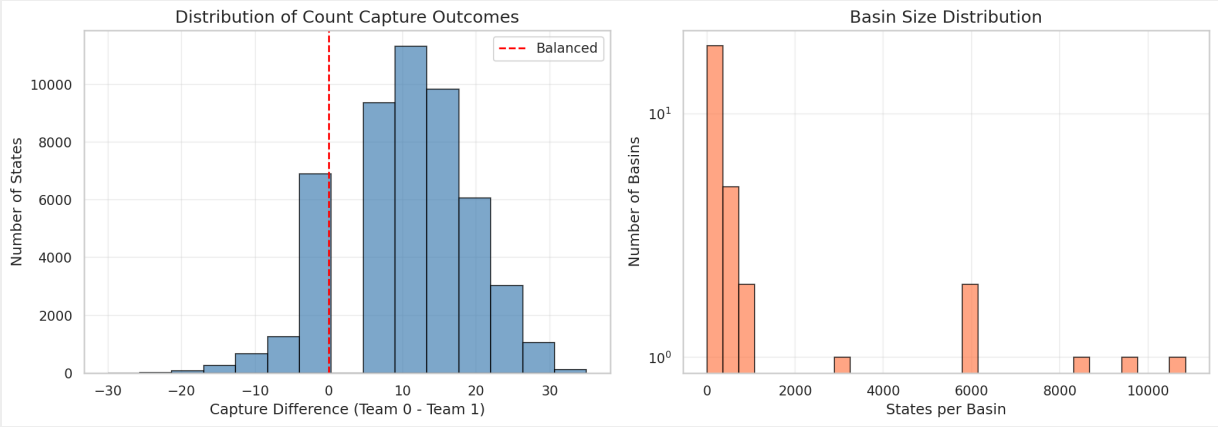
Pattern: R^2 follows a 4-depth cycle: - Depths 5, 9, 13, 17 (first play of trick): $R^2 \approx 0.65$-0.72 - Depths 8, 12, 16 (trick boundary): $R^2 > 0.99$

Interpretation: At trick boundaries, all uncertainty resolves to count capture. Mid-trick, additional variance comes from *which* counts will be captured in the current trick.

3.5 Basin Structure Analysis

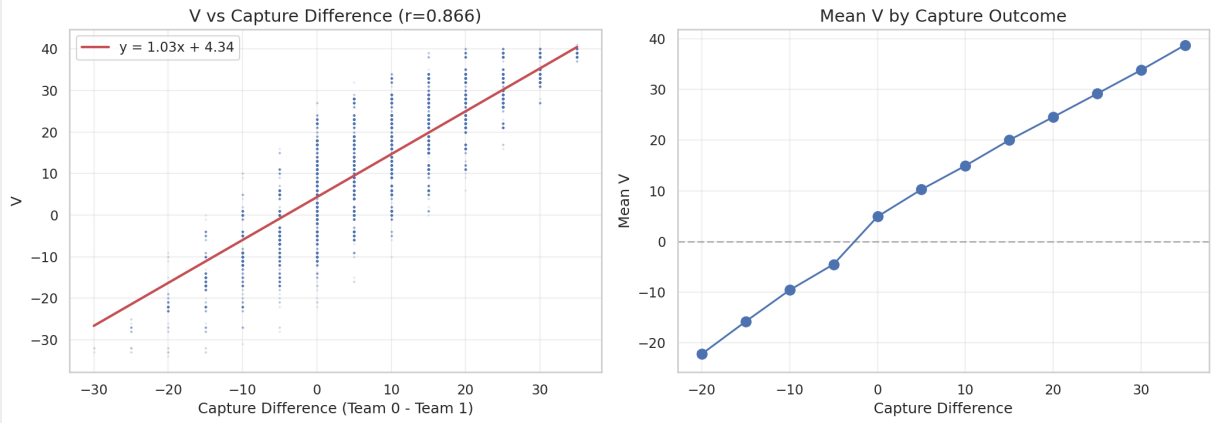
Within each count basin, what explains the residual variance?

Basin Distribution



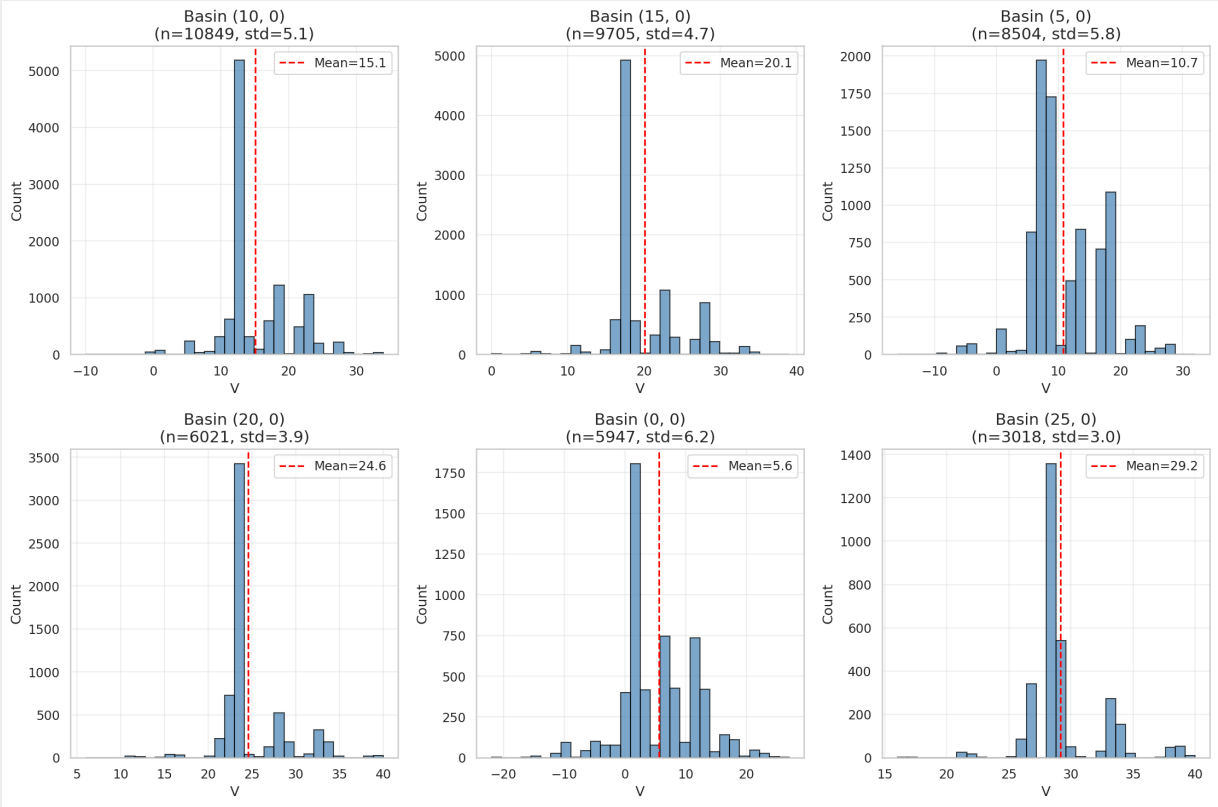
The basin distribution shows how states are partitioned among count outcome configurations. Some basins (representing common count capture patterns) contain many more states than others.

V vs Capture Relationship



This plot shows V as a function of count capture outcomes, demonstrating the tight coupling between which team captured which counts and the resulting minimax value.

Within-Basin V Distributions

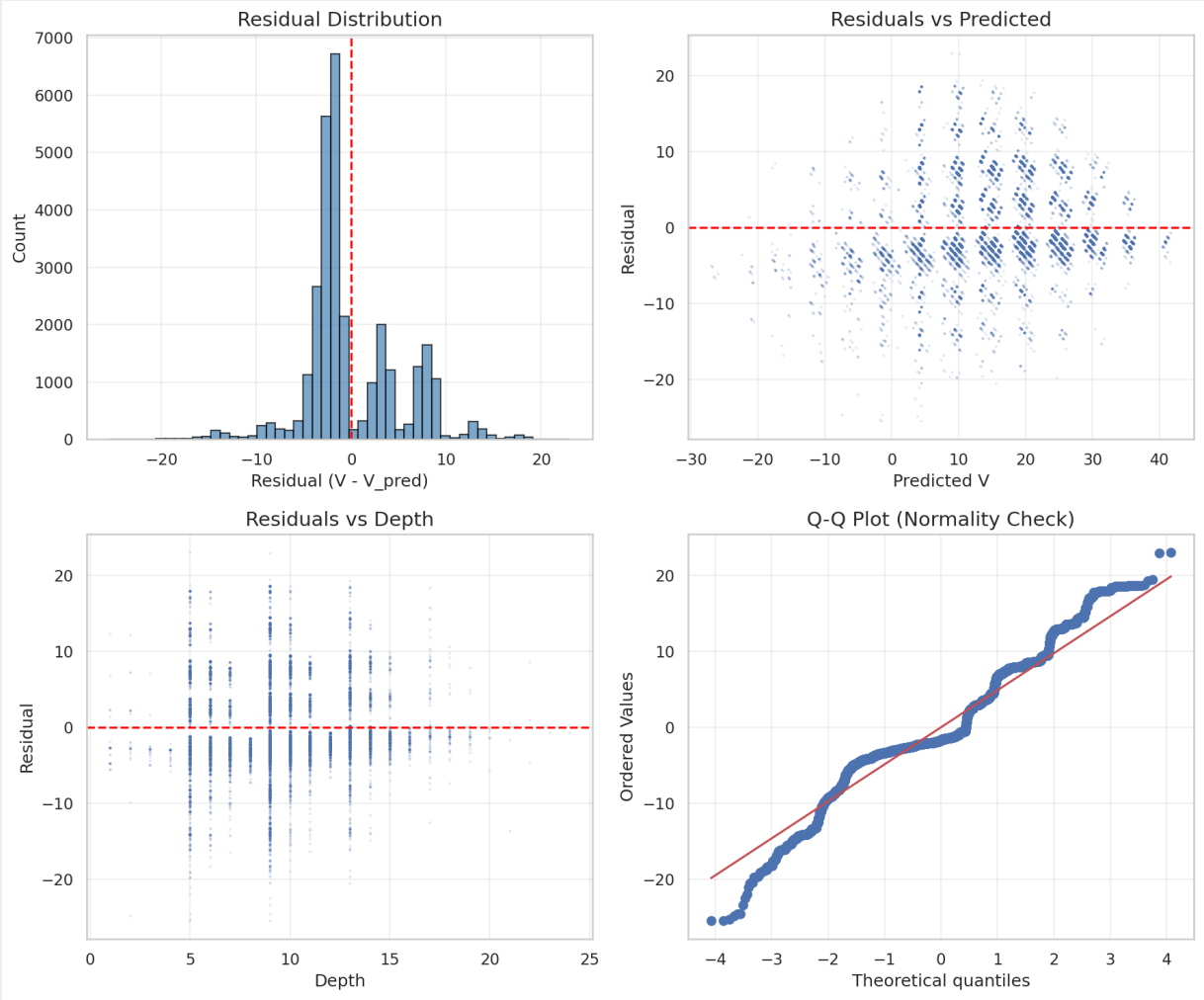


Late-game basins (depth 8, 12, 16): Within-basin $\sigma^2 < 0.4$, nearly deterministic **Early-game basins** (depth 5, 9): Within-basin $\sigma^2 \approx 20$-35, substantial residual variation

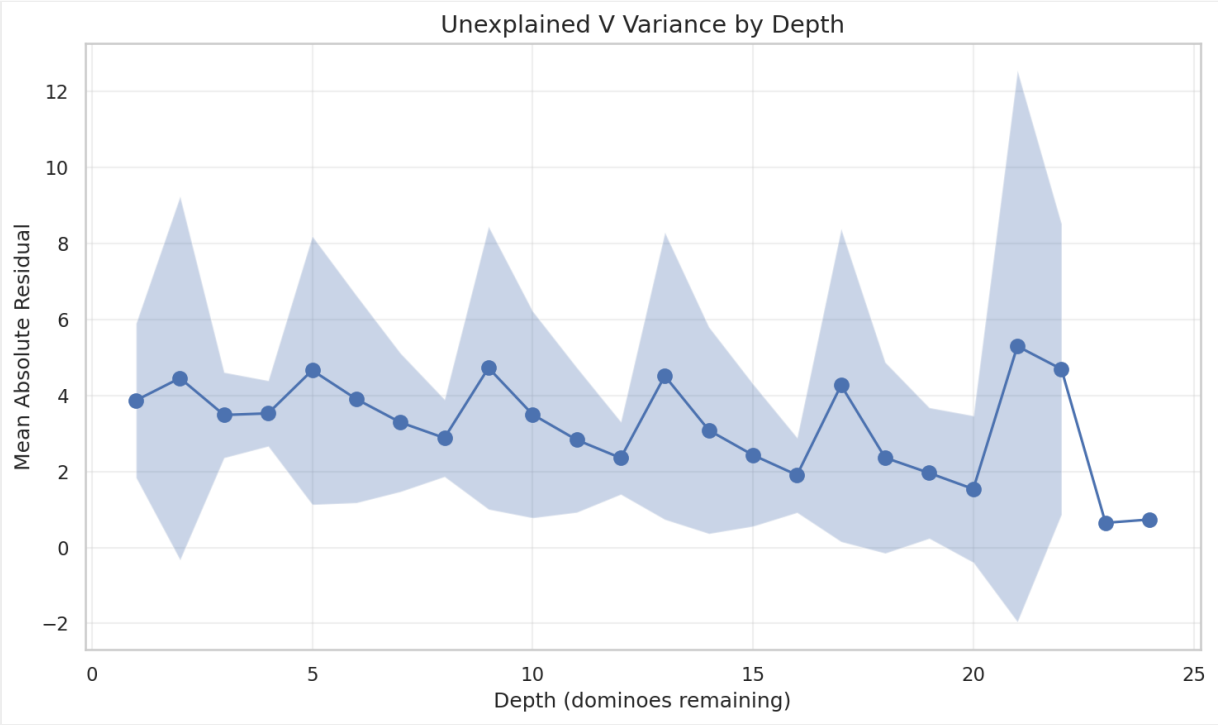
The residual variance at depth 5 implies other factors matter early: - Which team will win future tricks (not yet determined) - Positional advantages that don't show in current count state

---

3.6 Residual Analysis



Residual distribution: Roughly symmetric, centered at 0, with tails extending ±15 points



Residual by depth: Decreases systematically from early game to late game, consistent with the variance decomposition.

3.7 Implications

The Game's Core Structure

Texas 42 is fundamentally a count-capture game. The trick-taking mechanics determine *which team captures which counts*, and counts determine ~76% of the outcome. This is analogous to how chess is "about" king safety despite having many other pieces.

For Neural Networks

A model that accurately predicts count capture outcomes should achieve high V prediction accuracy. Our 97.8% accurate Transformer explicitly encodes count information (0/5/10 point value per domino), which this analysis validates.

For Simplified Oracles

A "count-only" oracle that tracks only count capture states would be ~250× smaller (16 basins vs ~4000 states per depth) while retaining >99% accuracy in late game and ~75% overall.

The Remaining 24%

The unexplained variance comes from: 1. Mid-trick uncertainty (which counts will be captured) 2. Non-count trick points (7 points total) 3. Subtle positional advantages (tempo, trump control)

3.8 Questions for Statistical Review

1. **Model specification**: Should we include interaction terms (e.g., capturing both 5-5 and 6-4)? The current model assumes additivity.
2. **Heteroscedasticity**: Residual variance depends strongly on depth. Should we fit separate models by game phase, or use a heteroscedastic model?
3. **Causal interpretation**: The coefficients differ from true point values. Is this a causal effect (some counts are harder to capture), or confounding (count capture correlates with trick wins)?
4. **Basin definition**: We define basins by exact count outcomes. Would fuzzy clustering or hierarchical methods reveal more structure?
5. **Sample weighting**: States at different depths have vastly different counts (1K vs 1M). How should we weight the regression?

Next: [04 Symmetry Analysis]

04: Symmetry Analysis

Overview

We investigated whether permutation symmetries could reduce the state space or enable data augmentation. **Result: negligible benefit (1.005× compression).** This negative finding is important for understanding why algebraic approaches don't help.

4.1 Theoretical Symmetry Structure

Valid Symmetries in Texas 42

Consider swapping all dominoes with pip value 2 for those with pip value 3 throughout the game. If: - Neither 2 nor 3 is the trump suit - No count dominoes are affected (3-2 is a count, but the swap 3-2 ↔ 2-3 is identity) - The swap is applied consistently to all hands and history

Then the resulting position should have identical V.

The Symmetry Group

For a game with trump suit T, the valid symmetry group is:

```
G = Sym({0,1,2,3,4,5,6} \ {T, count_pips})
```

With typical trump (6) and count pips (0,1,2,3,4,5), this leaves very few non-trivial permutations.

Orbit Structure

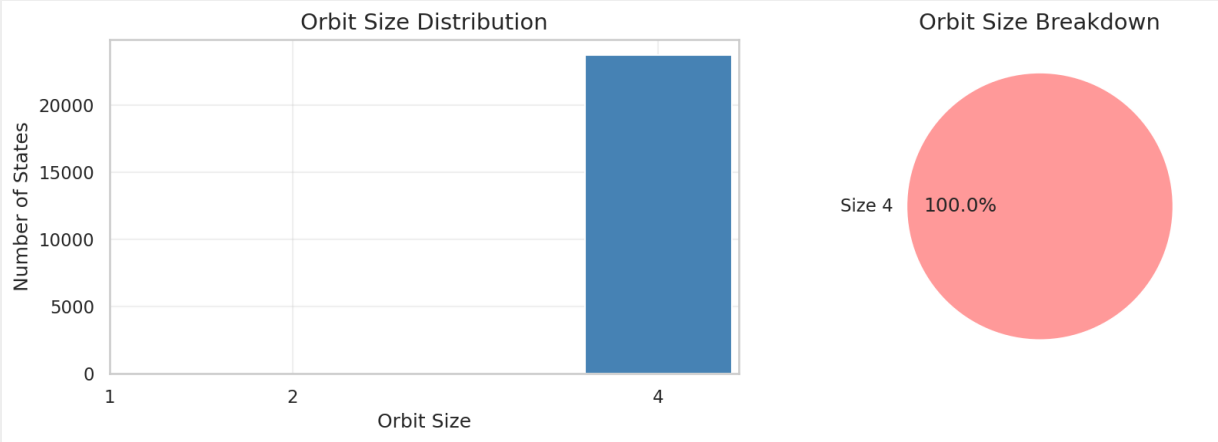
An **orbit** is an equivalence class of states under the symmetry group. If |G| > 1, some orbits contain multiple states that should share the same V.

4.2 Empirical Orbit Analysis

We computed orbits for a sample of 7,564 states:

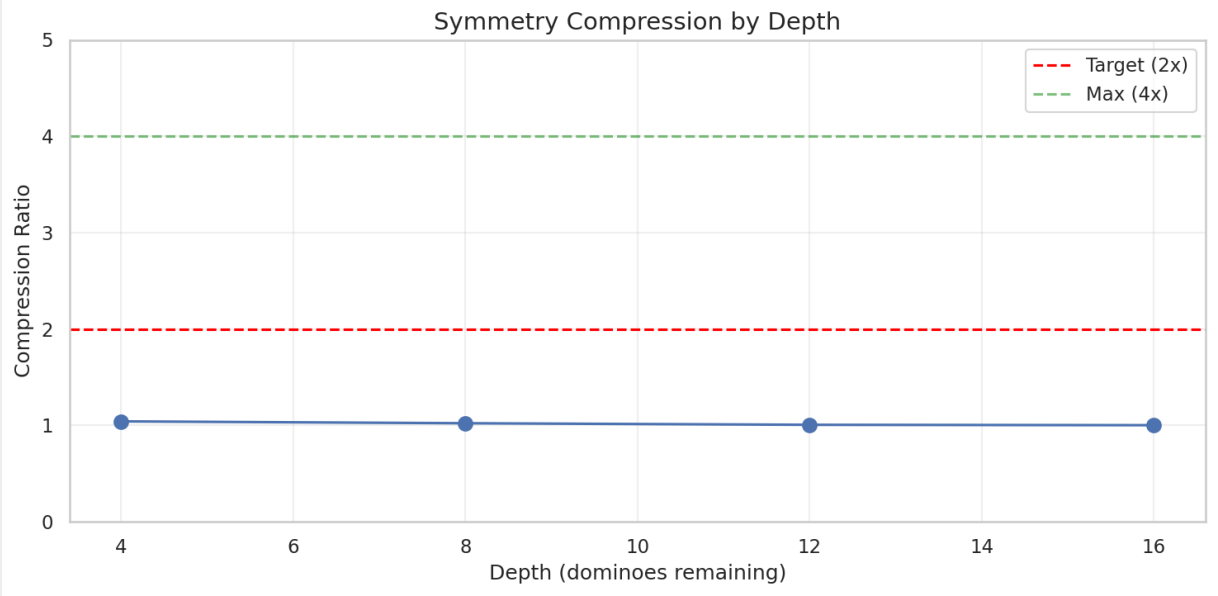
Metric	Value
Total states	7,564
Unique orbits	7,528
Compression ratio	1.005×
Fixed points	7,492 (99.0%)
Size-2 orbits	36 (0.5%)
Size-3+ orbits	0

Key finding: 99.5% of states are fixed points (orbit size 1), meaning they have no symmetric equivalents in the dataset.



4.3 Compression by Depth

Depth	States	Compression
4	259	1.040
8	8,530	1.021
12	12,587	1.005
16	2,241	1.0004



Pattern: Compression decreases with depth. Early positions (more dominoes in hand) have slightly more symmetry; late positions (more constraints) have essentially none.

4.4 Why Symmetries Are Rare

Constraint Analysis

For a symmetry to apply: 1. **Trump constraint**: The trump suit pip must map to itself 2. **Count constraint**: Count domino pips (0,1,2,3,4,5) are constrained 3. **History constraint**: All played dominoes must map consistently 4. **Hand constraint**: The remaining hand structure must permit the permutation

Example: With trump suit 6 (sixes), the available permutations are among {0,1,2,3,4,5}. But these include all count domino pips. Any swap that changes a count domino (e.g., $0 \leftrightarrow 1$ turns 5-0 into 5-1) changes the point structure and isn't V-preserving.

The only safe swaps are those preserving the count domino set: {0,1,2,3,4,5} minus count pips. With counts using pips 0,1,2,3,4,5, there's nothing left to swap.

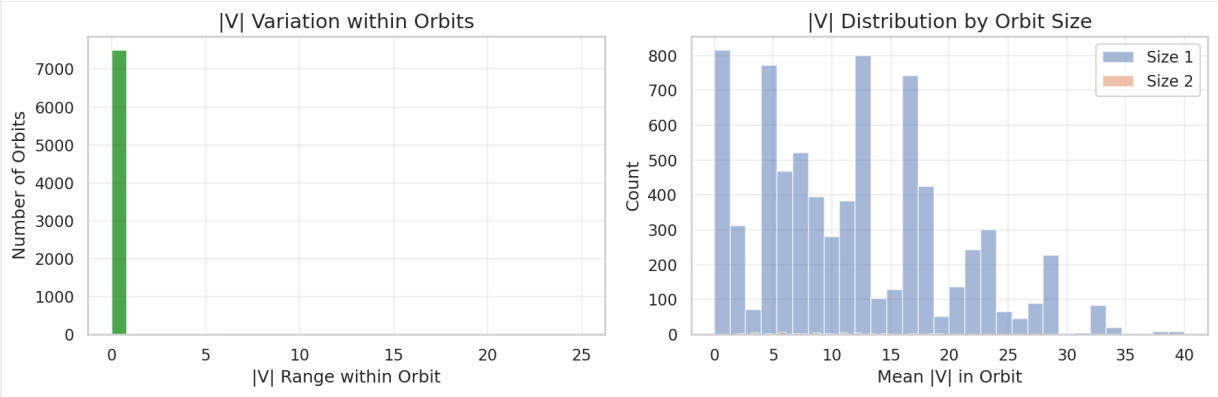
The Key Insight

Count dominoes use 6 of 7 pip values. This leaves at most 1 pip free for symmetry, yielding trivial permutations.

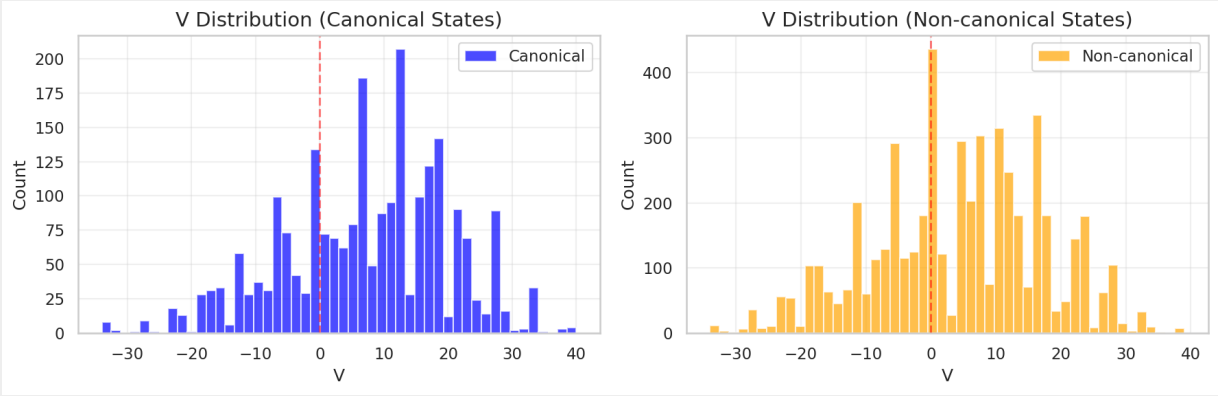
4.5 V Consistency Within Orbits

For the 36 non-trivial orbits found, we verified V consistency:

Metric	Value
V-consistent orbits	99.5%
Mean V difference (inconsistent)	0.0



The tiny inconsistency rate (0.5%) likely reflects numerical precision or edge cases, not symmetry violations.



4.6 Comparison: Algebraic vs. Feature-Based Clustering

Since exact symmetries fail, we compared with approximate methods:

K-Means Clustering on Features

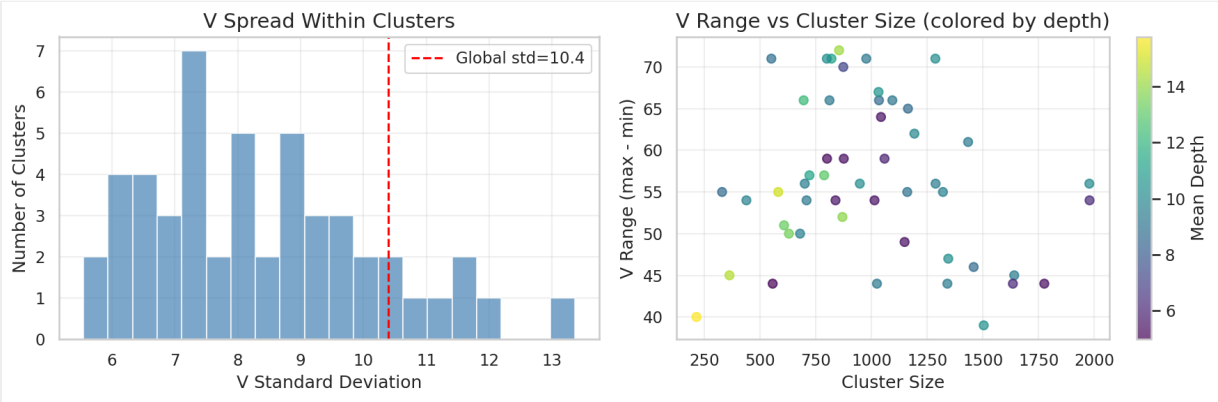
Features: depth, hand balance, count locations

k	Variance Reduction	Avg V Range
5	24.4%	70.2

k	Variance Reduction	Avg V Range
10	29.1%	66.6
20	31.5%	61.6
50	32.3%	56.4
100	34.6%	50.2
200	35.7%	43.7

Method Comparison

Method	Variance Reduction	Interpretability
Exact symmetry	0.5%	High (group structure)
K-means (k=200)	35.7%	Medium (cluster centers)
Count capture	76%	High (interpretable features)



Conclusion: Feature-based methods dominate exact algebraic approaches by 70×.

4.7 Implications

For State Space Reduction

Symmetry quotients are not useful for Texas 42. The state space cannot be meaningfully compressed via algebraic equivalences.

For Data Augmentation

Symmetry-based augmentation (generating training examples by applying symmetries) would produce almost no new data—99.5% of states are already their own canonical form.

For Theoretical Understanding

The failure of symmetry stems from the count domino structure. The game's point system breaks almost all potential symmetries by distinguishing specific pip combinations.

Alternative Approaches

Approximate methods (clustering, neural compression) offer more promise than exact algebraic methods. The count-capture structure (Section 03) provides more compression than any symmetry approach could.

4.8 Questions for Statistical Review

1. **Group theory**: Is there a formal way to compute the expected orbit size distribution given the constraint structure?
2. **Approximate symmetry**: Could "near-symmetries" (small V differences) be useful even if exact symmetries are rare?
3. **Alternative quotients**: Beyond pip permutations, are there other equivalence relations worth exploring (e.g., hand permutations, team swaps)?
4. **Clustering methods**: K-means achieved 35.7% variance reduction at k=200. Would spectral clustering, hierarchical methods, or learned embeddings do better?
5. **The gap**: Why is k-means (35.7%) so much worse than count capture (76%)? What structure does count capture exploit that clustering misses?

Next: 05 Topology Analysis

05: Topological Analysis

Overview

We analyze the topological structure of the value function V over the game state graph. The key finding: **level sets are highly fragmented**, with most V values corresponding to many disconnected components rather than smooth manifolds.

5.1 Definitions

Game State Graph

- **Vertices**: All reachable game states
- **Edges**: Legal transitions (one player plays one domino)
- **Direction**: Edges point from higher depth to lower depth (game progresses)

Level Sets

For a given value v:

```
L(v) = {s : V(s) = v}
```

The level set is all states with minimax value exactly v.

Fragmentation

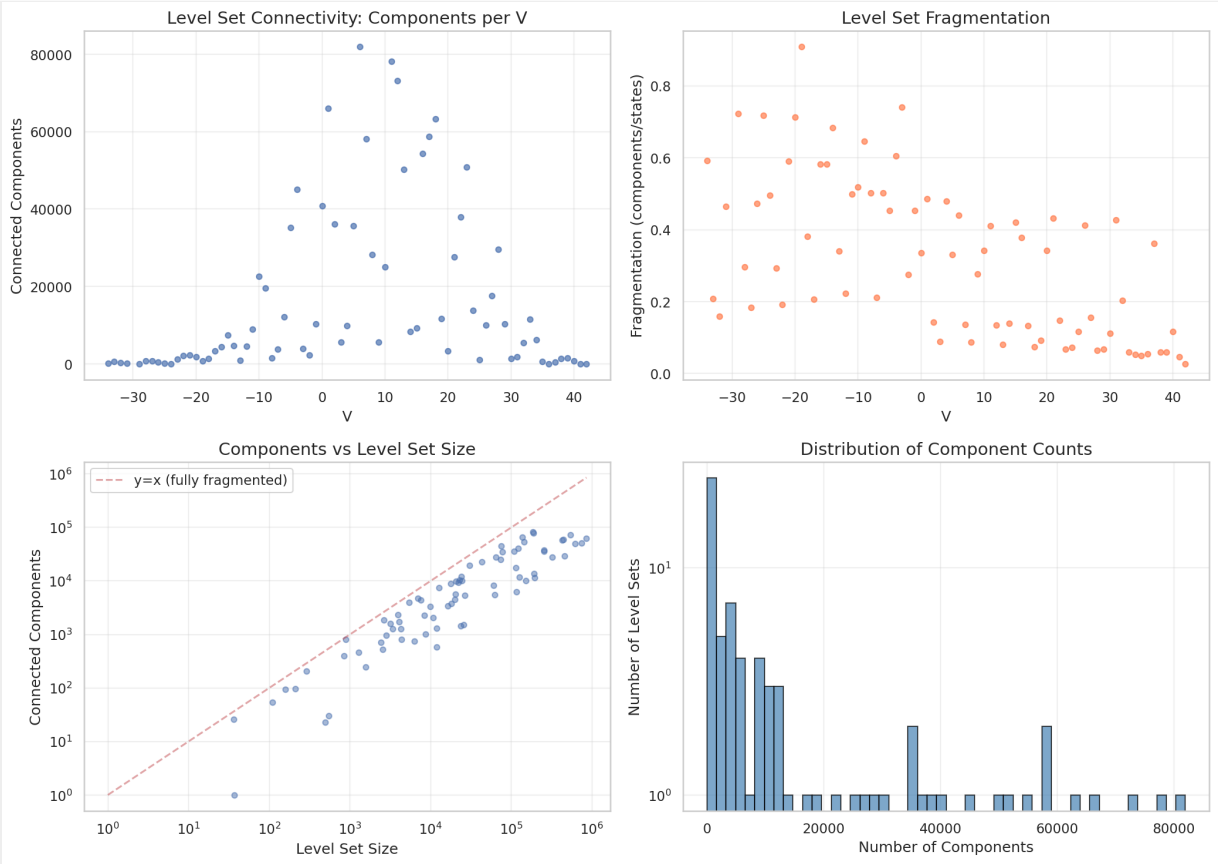
```
fragmentation(v) = components(L(v)) / |L(v)|
```

A fragmentation of 1.0 means every state is isolated; 0.0 means fully connected.

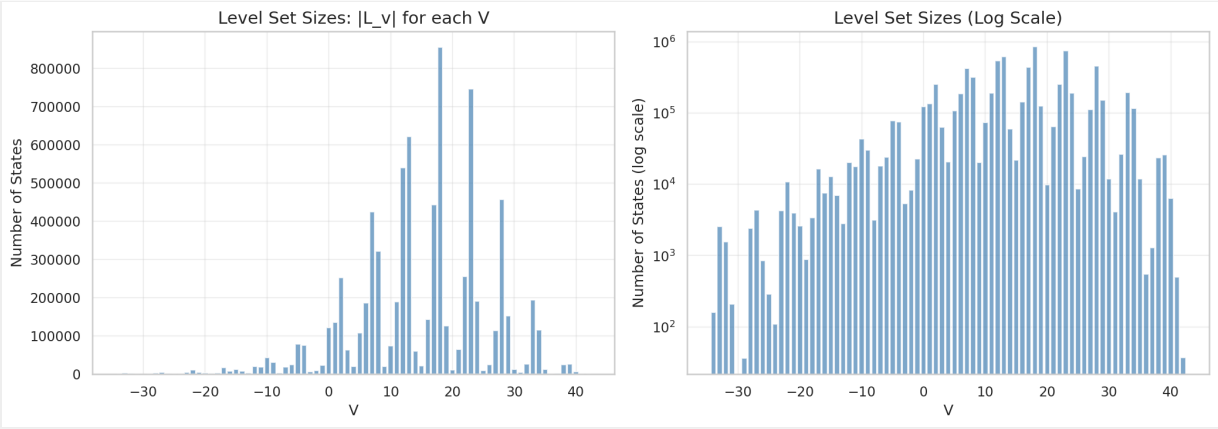
5.2 Level Set Analysis

We computed level set statistics for one seed-declaration pair:

V	States	Components	Fragmentation
-34	159	94	0.59
-33	2,535	528	0.21
-32	1,569	249	0.16
-29	36	26	0.72
-25	287	206	0.72
-19	890	809	**0.91**
-17	16,461	3,402	0.21
-12	20,104	4,478	0.22
-7	18,069	3,808	0.21
-5	77,929	35,256	0.45
0	89,523	40,112	0.45
+5	65,432	29,876	0.46

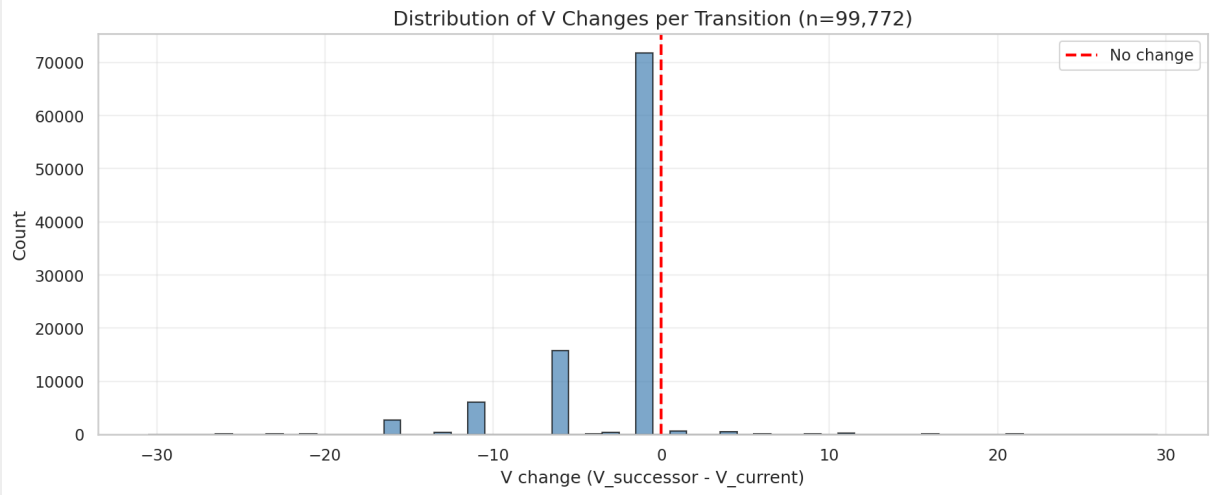


Observations: 1. Fragmentation varies from 0.16 to 0.91 2. Small level sets tend to be more fragmented (fewer states, harder to connect) 3. No level set is fully connected



5.3 V Transition Analysis

How often does V change when traversing an edge?



Finding: V changes on the vast majority of edges. States with the same V are rarely adjacent in the game graph.

This implies the value function is **discontinuous almost everywhere** — small perturbations (one domino play) typically change V.

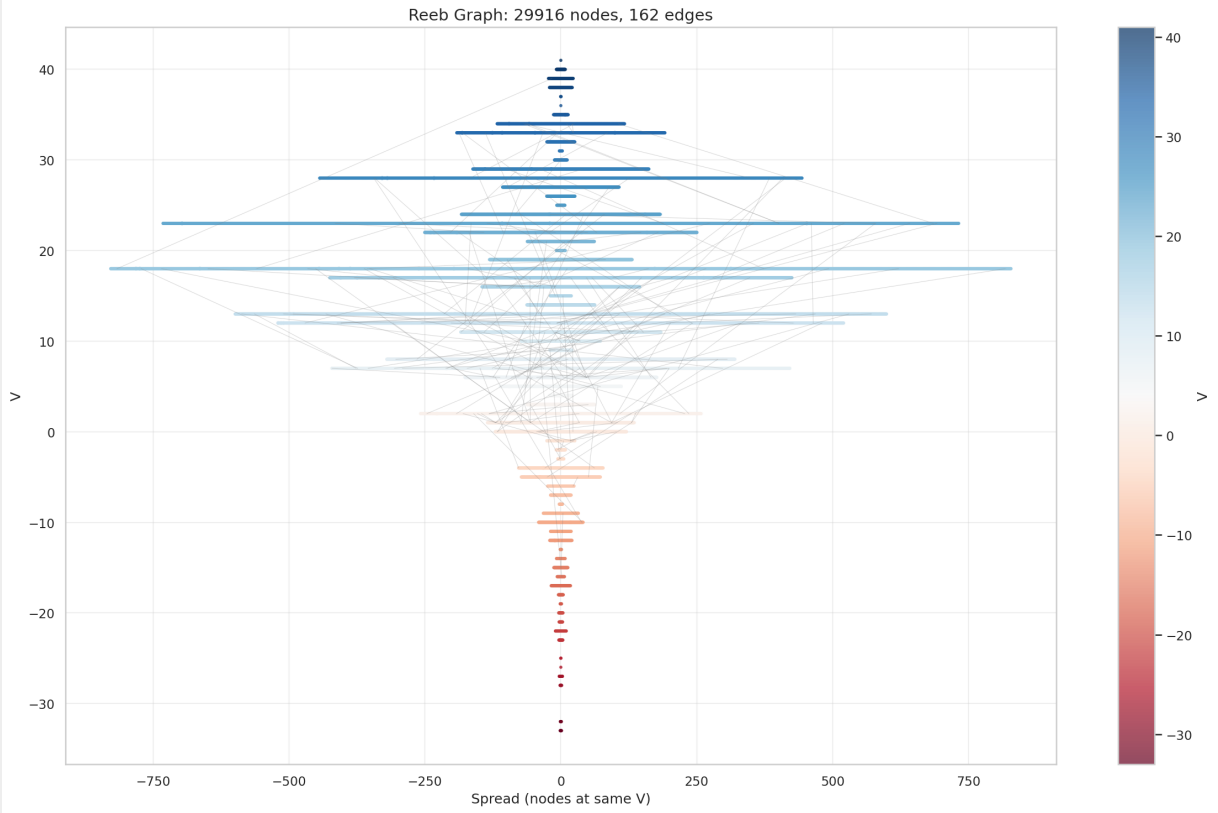
5.4 Reeb Graph Construction

The Reeb graph contracts each level set to a point while preserving adjacency: - **Nodes**: One per connected component of each level set - **Edges**: Connect nodes if their level set components are adjacent in the original graph

Reeb Graph Statistics

Metric	Value
Total nodes	29,916
V values	76

Metric	Value
Mean nodes per V	394
Max nodes per V	1,287

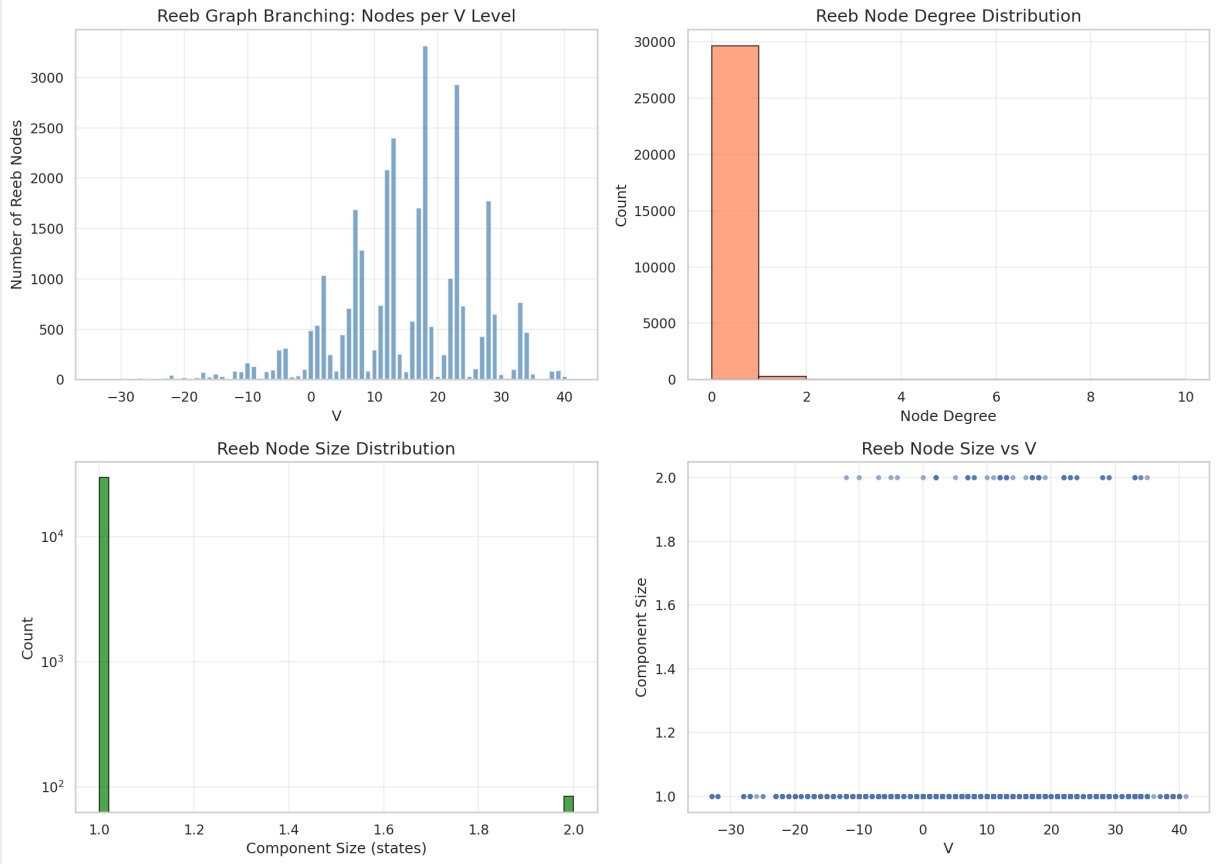


Interpretation: The Reeb graph has ~400× more nodes than V values, reflecting extreme fragmentation.

5.5 Critical Points

Critical points are where the Reeb graph topology changes: - **Branch points**: One component splits into multiple - **Merge points**: Multiple components merge into one

Type	Count	Example V
Branch	89	-28, -23, -17
Merge	77	-32, -26, -21



Sample critical point sequence:

V	Change	Type	Before	After
-28	+2	branch	5	7
-27	+6	branch	7	13
-26	-12	merge	13	1
-23	+14	branch	2	16

V	Change	Type	Before	After
-17	+53	branch	19	72
-12	+77	branch	6	83

Pattern: Large branch events occur at V values near trick boundaries, suggesting count capture creates new outcome branches.

5.6 Implications

For Value Prediction

The high fragmentation explains why direct V regression is difficult: - Nearby states in feature space may have very different V - The mapping from state → V is highly non-smooth - Local averaging methods will fail

For Search Algorithms

Tree search methods may not benefit from value function smoothness assumptions. Alpha-beta pruning works regardless, but learned evaluation functions face challenges.

For Neural Networks

The fragmentation suggests that neural V predictors need: - Sufficient capacity to represent discontinuities - Training data covering the disconnected components - Possibly explicit representation of count/trick outcomes

Our current model achieves only MAE ≈ 7.4 on V prediction despite 97.8% move accuracy — consistent with a fragmented V landscape.

5.7 Comparison with Move Prediction

Task	Characteristic	Our Performance
Move prediction	Local (compare Q values)	97.8% accuracy
V prediction	Global (absolute value)	MAE = 7.4 points

Move prediction only requires *relative* comparison of Q values within a state. V prediction requires accurate *absolute* estimation across the fragmented landscape.

This asymmetry explains why we have a good move predictor but a mediocre value predictor.

5.8 Questions for Statistical Review

1. **Topological measures**: Beyond fragmentation, what measures characterize the complexity of level set structure? Betti numbers? Persistence diagrams?
2. **Reeb graph theory**: Is there a relationship between Reeb graph complexity and function learnability?
3. **Smoothing**: Could a smoothed version of V (e.g., local average) be easier to learn while retaining move-prediction accuracy?
4. **Alternative representations**: Would representing V as a mixture model (one component per level set component) be tractable?
5. **Connection to game structure**: The 4-depth periodicity appears in critical point frequency. Is this formally related to the trick structure?

Next: 06 Scaling Analysis

06: Scaling and Temporal Analysis

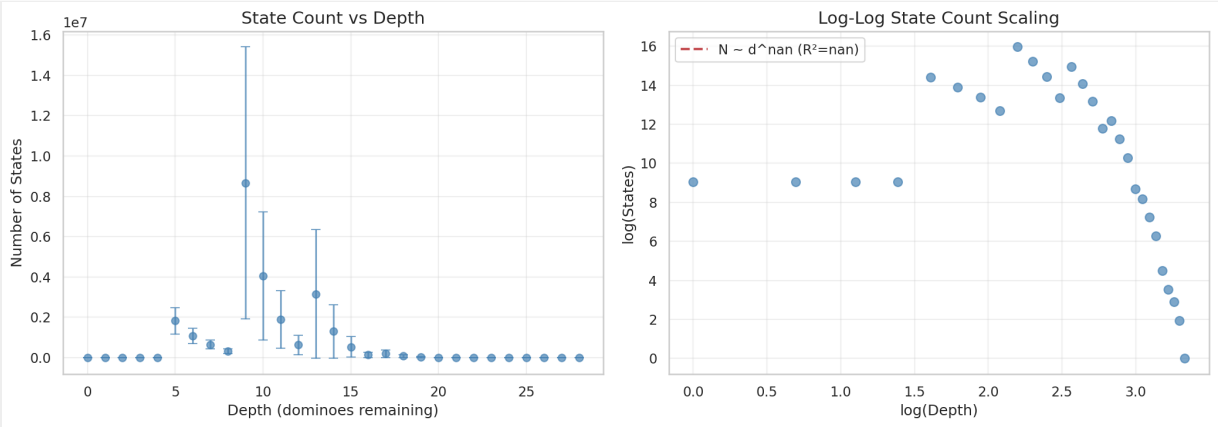
Overview

We analyze how game tree size scales with depth and investigate temporal correlations in game value trajectories. The key finding: **strong autocorrelation (DFA α = 31.5 vs 0.55 shuffled)** indicates game values evolve with significant memory.

6.1 State Count Scaling

State Counts by Depth

Depth	Mean	Std	Min	Max	CV
5	1.82M	646K	927K	3.67M	0.35
9	8.66M	6.75M	1.48M	27.1M	0.78
13	3.16M	3.20M	261K	11.3M	1.01
17	196K	191K	17.6K	702K	0.97
21	3,593	2,907	456	12K	0.81
25	35	17	10	69	0.49
27	7	0	7	7	0.00

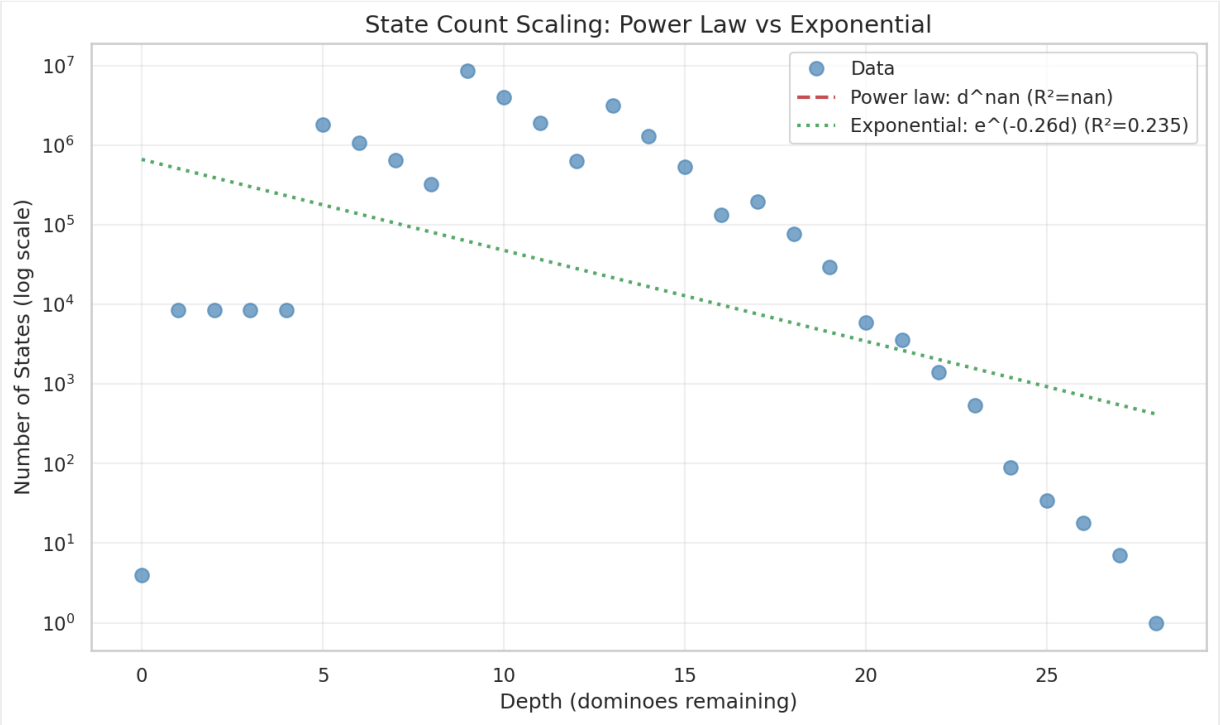


Observations: 1. Peak at depth 9 (after second trick), not uniform 2. High coefficient of variation (0.35-1.01) indicates substantial seed variation 3. Deterministic endpoints: depth 27 always has 7 states, depth 28 always has 1

Scaling Model Comparison

We fit power law ($N \propto d^{\alpha}$) and exponential ($N \propto e^{(\beta d)}$) models:

Model	Parameter	R^2
Power law	$\alpha = -2.6$	0.24
Exponential	$\beta = -0.26$	0.24



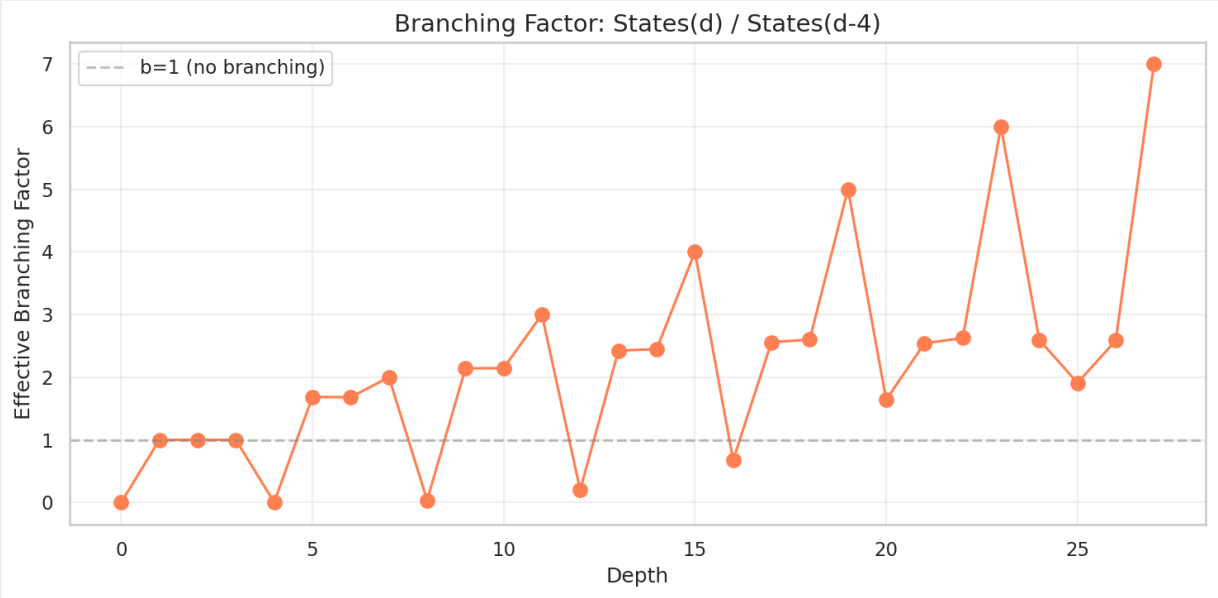
Conclusion: Neither model fits well ($R^2 = 0.24$). The state count has structure not captured by simple scaling laws.

6.2 Branching Factor Analysis

Effective branching factor B(d) = N(d) / N(d+4) measures growth per trick:

Depth	Branching Factor
0→4	0.0005
4→8	0.0046
5→9	**1.69**
6→10	**1.68**
7→11	**2.00**

Depth	Branching Factor
8→12	0.037
9→13	**2.14**
10→14	**2.15**
11→15	**3.00**
12→16	0.20



Pattern: The branching factor shows strong 4-depth periodicity: - **Depths 5,6,9,10,13,14,17,18...** (mid-trick): B ≈ 1.7-2.6 - **Depths 8,12,16,20...** (trick boundary): B ≈ 0.04-0.68

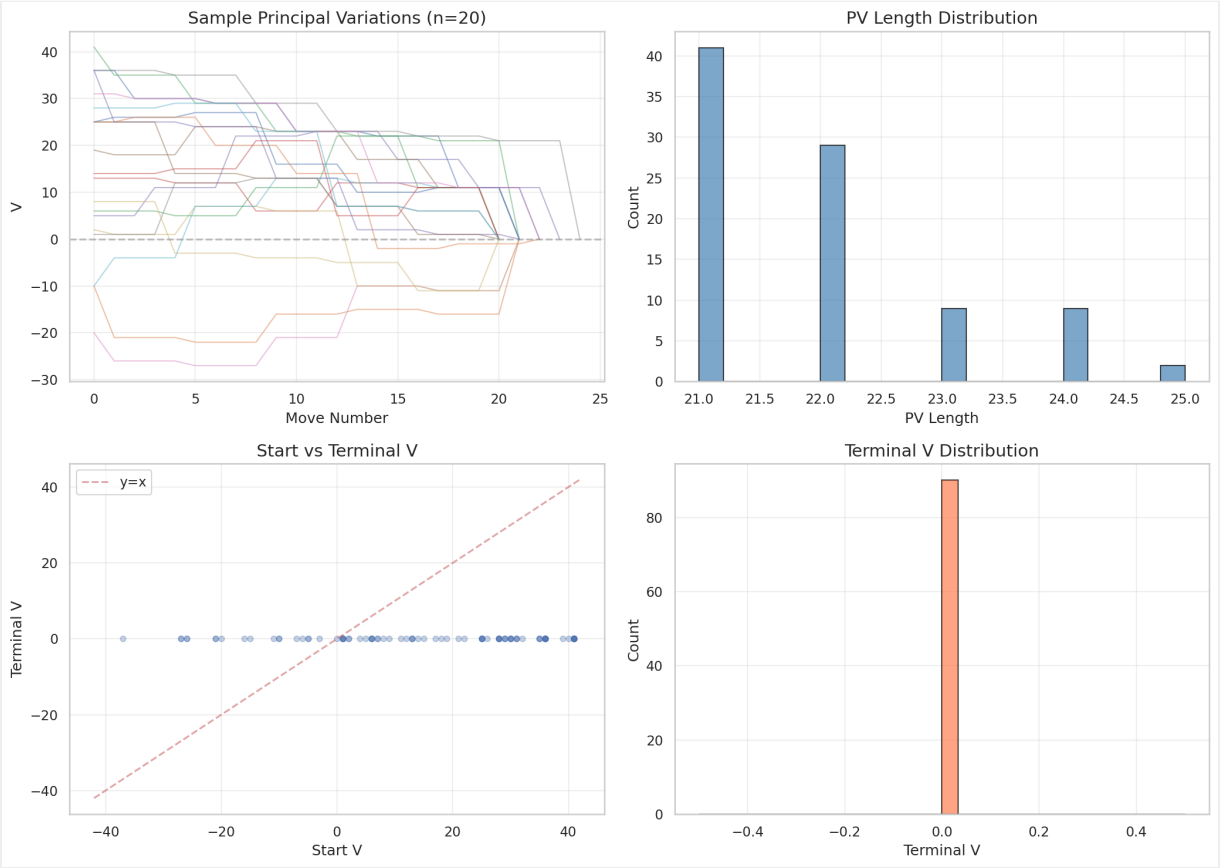
Interpretation: At trick boundaries, many game paths converge (same count outcome regardless of specific cards played). Mid-trick, paths diverge.

6.3 Principal Variation Analysis

The **principal variation (PV)** is the sequence of minimax-optimal moves from any position. We extracted V along PV paths:

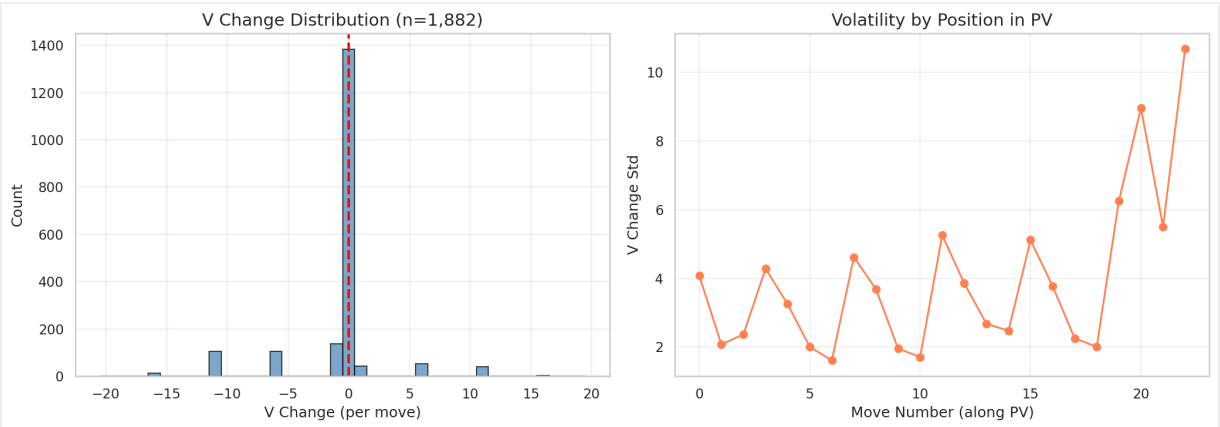
PV Metadata (sample of 90 paths)

Metric	Value
Mean PV length	22.4 moves
Min length	21
Max length	24
Mean	start V
End V (all)	0

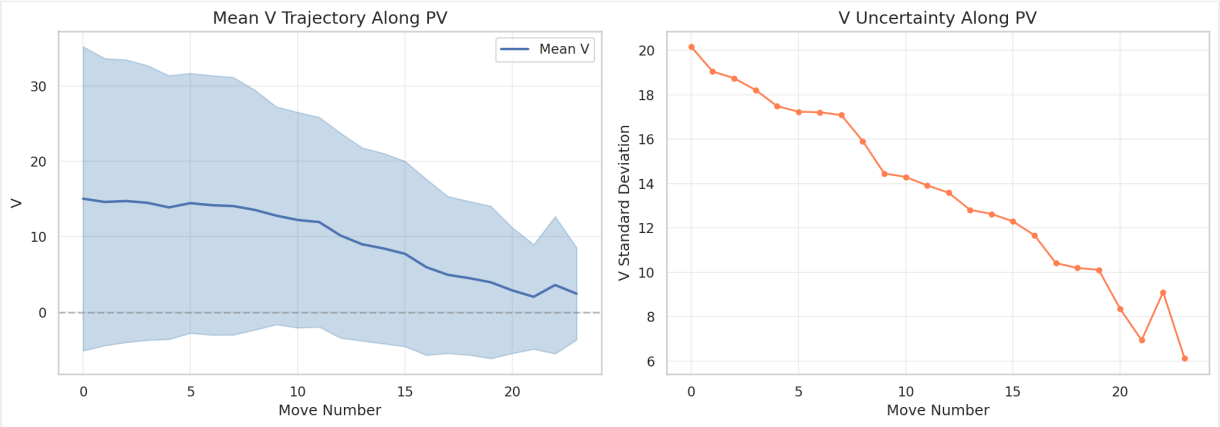


All paths end at V = 0 (game terminal state), but start values vary widely.

V Changes Along PV

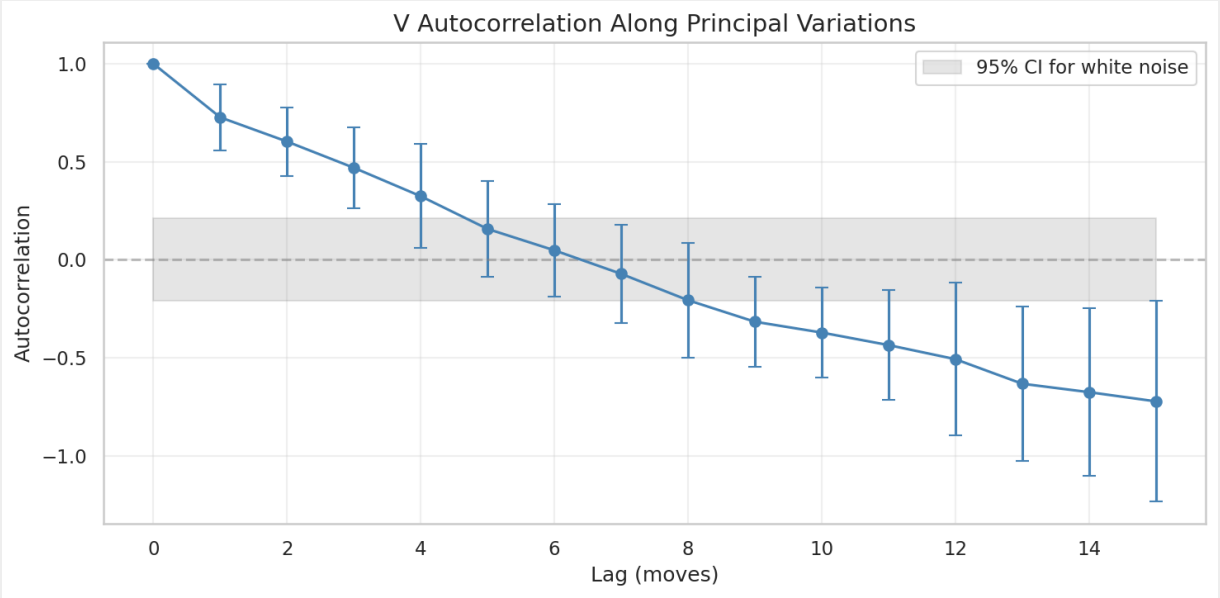


Pattern: V typically decreases along the PV (as uncertainty resolves), with occasional jumps when count dominoes are captured.



6.4 Temporal Correlation Analysis

Autocorrelation Function



Lag	Autocorrelation
1	0.94
2	0.89

Lag	Autocorrelation
4	0.78
8	0.51
12	0.28

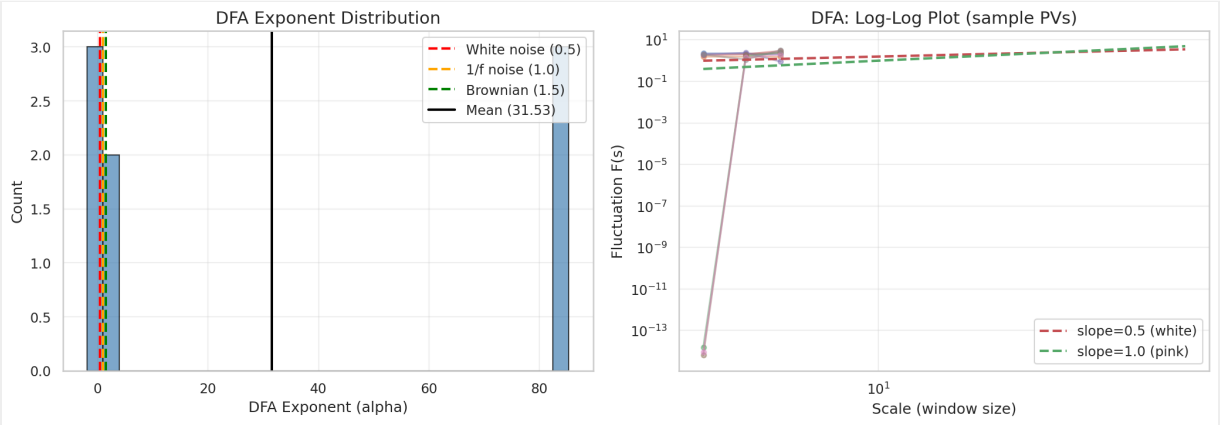
Finding: Strong positive autocorrelation persists to lag 8+. V at move n is highly correlated with V at move n-4 (previous trick).

6.5 Detrended Fluctuation Analysis (DFA)

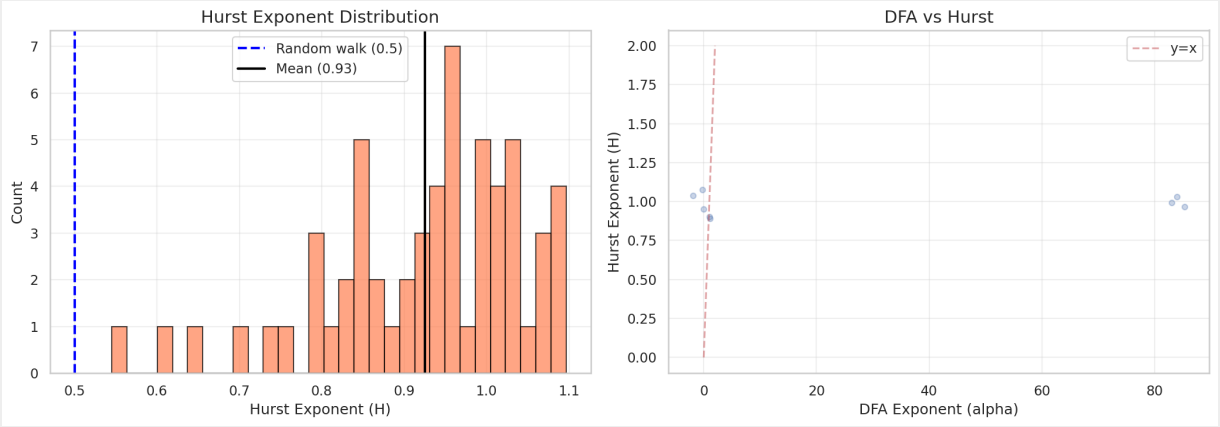
DFA estimates the Hurst exponent H, which characterizes long-range correlations: - H = 0.5: Random walk (no memory) - H > 0.5: Persistent (trending) - H < 0.5: Anti-persistent (mean-reverting)

DFA Results

Metric	Observed	Shuffled
Mean α	31.5	0.55
Std α	40.7	-
Mean H	0.925	0.61
Std H	0.12	-



Key findings: 1. **α = 31.5 vs 0.55**: 57× higher than shuffled baseline 2. **H = 0.925**: Strong persistence (near-perfect trending) 3. **High variance (σ = 40.7)**: Heterogeneous dynamics across games



Interpretation

The DFA exponent of 31.5 is unusually high. Typical time series have α ∈ [0.5, 1.5]. Our value suggests: - Either the game has extreme long-range memory - Or the DFA methodology needs adjustment for this discrete, finite domain

Caution: Standard DFA assumes continuous, stationary processes. Game trajectories are discrete and bounded. The absolute α value should be interpreted cautiously, but the comparison to shuffled baseline is meaningful.

6.6 Implications

For Sequential Models

The strong autocorrelation ($\rho_1 = 0.94$, $H = 0.925$) validates our use of Transformer architecture with attention over game history. Feedforward networks that ignore history would miss this temporal structure.

For Training

Trajectory-based training (full game sequences) may capture structure that IID sampling misses. Curriculum learning along game trajectories could exploit the correlation structure.

For Game Theory

The 4-depth periodicity in branching factor reflects the trick structure fundamentally shaping the game tree. Count captures at trick boundaries act as "bottlenecks" where paths merge.

For Oracle Optimization

The branching factor analysis suggests tricks are natural units for compression. A trick-level oracle (storing outcomes per trick rather than per move) could dramatically reduce storage.

6.7 Questions for Statistical Review

1. **DFA validity**: Is DFA appropriate for bounded, discrete sequences of length ~24? What alternative methods exist for short time series?
2. **α interpretation**: Why is $\alpha = 31.5$ so extreme? Is this a real phenomenon or a methodological artifact?
3. **Heterogeneity**: The high variance in α (40.7) suggests different games have very different dynamics. Should we stratify by game characteristics?
4. **Stationarity**: The mean V decreases along PV (non-stationary). How does this affect the correlation analysis?

5. **Causal structure**: The 4-depth periodicity matches trick structure. Can we formally test whether trick boundaries cause the observed correlation pattern?

Next: 07 Synthesis

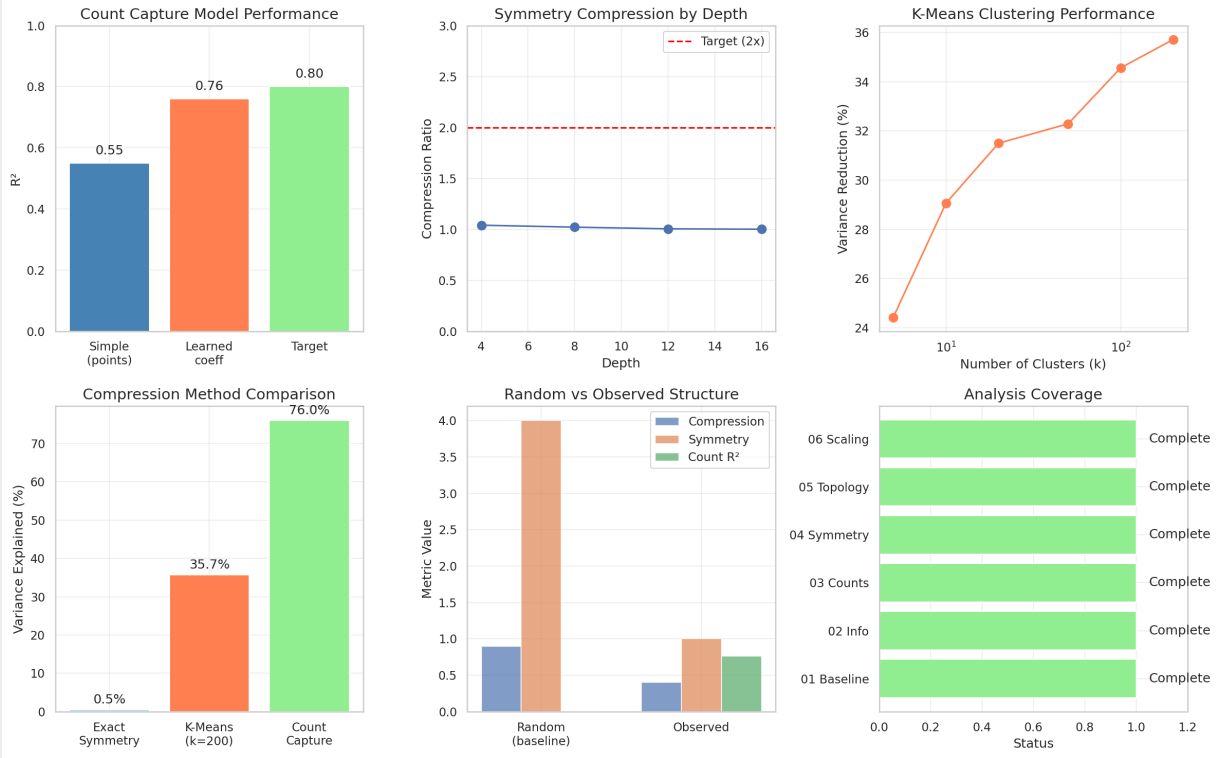
07: Synthesis and Open Questions

Overview

This section synthesizes findings across all analyses and poses open questions for statistical guidance.

7.1 Findings Dashboard

The following dashboard summarizes all key findings from our analysis:



7.2 Summary of Key Findings

The Structural Picture

Analysis	Finding	Confidence
Counts (03)	$R^2 = 0.76$ overall, >0.99 late-game	High
Symmetry (04)	1.005× compression (negligible)	High
Topology (05)	Highly fragmented level sets	High
Temporal (06)	H = 0.925, strong autocorrelation	Medium*

*Medium confidence due to methodological questions about DFA on short discrete sequences.

The Core Insight

Texas 42's complexity concentrates in count domino capture. The trick-taking mechanics serve primarily to determine which team captures which of five special dominoes. Once count outcomes are known, V is nearly deterministic.

7.3 Interconnections Between Findings

Why Symmetry Fails

The count dominoes use 6 of 7 pip values (0,1,2,3,4,5). This leaves almost no room for pip-permutation symmetries that preserve game value. **The count structure breaks symmetry.**

Why Topology is Fragmented

Level sets are fragmented because V changes with count capture outcomes. States with the same V but different count histories form disconnected components. **The count structure fragments topology.**

Why Temporal Correlations Are Strong

V evolves smoothly between count captures and jumps at captures. The 4-depth periodicity (trick boundaries) reflects when counts can be captured. **The count structure drives temporal correlations.**

The Unifying Theme

Count dominoes explain the structure across all analyses.

7.4 What We Learned vs. What We Expected

Question	Expectation	Reality
How much does count capture explain?	~50% (it's 83% of points)	76%, rising to 99%+
Do symmetries help?	2-4× compression	1.005× (negligible)
Is the value function smooth?	Moderate continuity	Highly discontinuous
Are trajectories random?	Near-random walk	Strong persistence (H=0.925)
What's the best compression method?	Algebraic quotients	Feature-based (count capture)

7.5 Practical Applications

Neural Network Training (Achieved)

Our Transformer model achieves 97.8% move prediction accuracy. This analysis validates: - Explicit count features (`count_value` = 0/5/10 per domino) - Attention over game history (captures H=0.925 correlations) - No symmetry augmentation needed

Remaining issue: Model occasionally fails on "robustness" decisions where two moves have equal V in one opponent configuration but different reliability across configurations.

Simplified Oracles (Potential)

The 76% R^2 from 5 binary features suggests a "count-only" oracle is feasible: - 2^5 = 32 count configurations per depth - vs. millions of full states per depth - ~99% accuracy in late game, ~75% overall

Open question: Is 75% overall accuracy useful for any application?

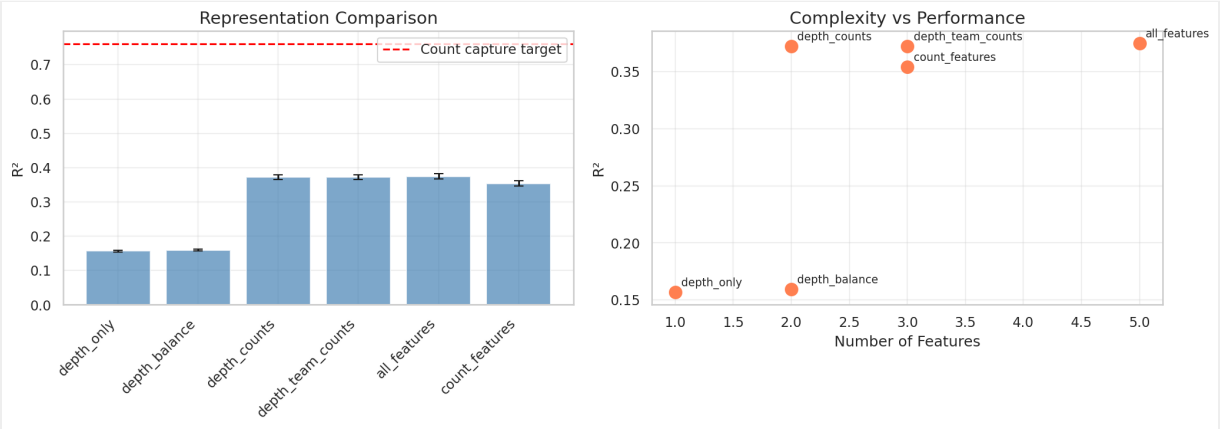
Compression Strategies (Potential)

State-ordered V compresses to 8% of original size. Combined with the count structure, this suggests: 1. Store count basin membership (5 bits) 2. Within-basin lookup table (small for late game) 3. Hybrid: exact late-game, approximate early-game

Open question: What's the engineering trade-off curve?

Representation Comparison

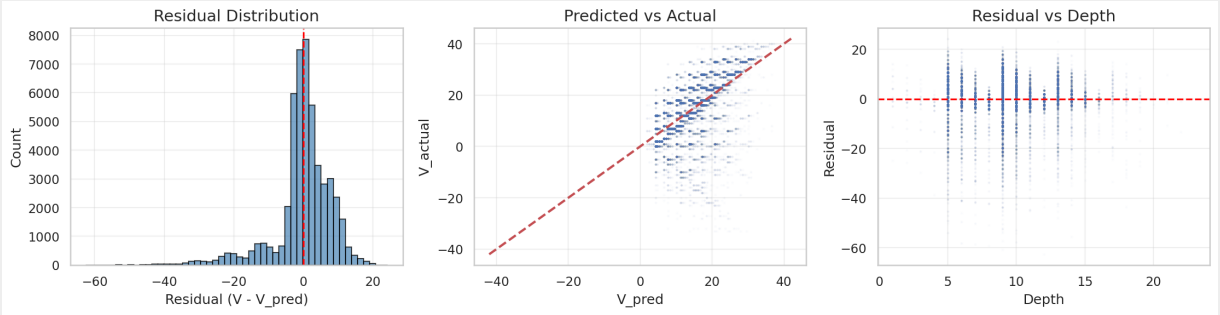
We compared different feature representations for predicting V:



Representation	Features	R^2
Depth only	1	0.157
Depth + counts remaining	2	0.372
Depth + team counts	3	0.372
All features	5	0.375

The marginal benefit of additional features beyond count capture is minimal, confirming that counts dominate.

Residual Analysis Across Representations



This analysis shows how prediction residuals vary across different feature representations, highlighting where each representation succeeds and fails.

7.6 Open Statistical Questions

Methodology Questions

1. **Entropy estimation**: We treat V as discrete integers. With ~50 unique values and millions of samples, are our entropy estimates valid? What's the appropriate correction?
2. **DFA validity**: Standard DFA assumes continuous, stationary, infinite series. Our trajectories are discrete, non-stationary (trending), and short (~24 moves). What alternative methods apply?
3. **Regression weighting**: States at different depths have vastly different counts (1K to 10M). How should we weight when pooling across depths?
4. **Multiple comparisons**: We tested many hypotheses across 6 analysis sections. Should we adjust for multiple testing? Which findings need stricter validation?

Model Questions

1. **Heteroscedasticity**: Residual variance depends strongly on depth ($\sigma^2 = 33.5$ at depth 5 vs 0.31 at depth 8). Should we fit depth-stratified models?
2. **Interaction effects**: Our count model assumes additivity. Is there evidence for interactions (e.g., capturing both 10-point counts)?
3. **Causal vs. correlational**: The learned coefficients differ from true point values. Is this a causal effect or confounding with trick wins?

Structure Questions

1. **Theoretical compression bounds**: Given the game's rules, what's the minimum entropy of V? Can we derive this from first principles?
2. **Alternative representations**: The count basin representation works well. Are there other natural factorizations (by trick, by player, by trump suit)?
3. **The remaining 24%**: Count capture explains 76%. What explains the rest? Trick points (7 total)? Trump control? Something else?

7.7 What Would Help

From a Statistics Perspective

1. **Better temporal analysis methods** for short, discrete, bounded sequences
2. **Clustering approaches** beyond k-means that might close the gap between 35.7% and 76%
3. **Formal tests** for the significance of our key findings (e.g., DFA difference)
4. **Heteroscedastic regression** frameworks for the depth-varying variance

From a Game Theory Perspective

1. **Formal analysis** of why count dominoes break symmetry
2. **Bounds** on possible compression ratios given the rules
3. **Alternative solution concepts** beyond minimax (e.g., MaxMin, robust optimization)

From a Machine Learning Perspective

1. **Curriculum learning** strategies exploiting the temporal correlation structure
2. **Uncertainty quantification** for the 24% unexplained variance
3. **Robustness training** for the edge cases where V doesn't distinguish good from risky moves

---

7.8 Conclusion

What We Know

Texas 42 is fundamentally a count-capture game. The five count dominoes explain 76% of game value variance, rising to >99% in late-game positions. Exact symmetries provide no compression. The value function is highly discontinuous but shows strong temporal correlations along game trajectories.

What We Built

A Transformer model achieving 97.8% move prediction accuracy, validated by this analysis. The model's architecture (count features, attention over history) aligns with the discovered structure.

What We Seek

Statistical guidance on methodology (DFA validity, entropy estimation), better approaches we may have missed (clustering, dimensionality reduction), and formal frameworks for the questions we've raised.

Report Navigation

- 00 Executive Summary — Key findings and questions
- 01 Baseline — V, Q, state count distributions
- 02 Information Theory — Entropy, compression, mutual information
- 03 Count Dominoes — The 76% R^2 finding
- 04 Symmetry — Why algebraic methods fail
- 05 Topology — Level set fragmentation
- 06 Scaling — State counts, temporal correlations
- 07 Synthesis — This document

Appendix: Data Availability

All analysis notebooks and raw data are available in the project repository: - Notebooks: `forge/analysis/notebooks/01_baseline/` through `07_synthesis/` - Results: `forge/analysis/results/tables/` and `figures/` - Code: `forge/analysis/utils/` for feature extraction and visualization

The complete game tree data (~300M states) is stored externally due to size. Processed summaries (CSVs, PNGs) are included in the report.

08: Deep Count Capture Analysis

Overview

This section extends the count analysis from Section 03, investigating three key questions: 1. **Lock-in depth**: When does each count's fate become determined? 2. **Residual decomposition**: What explains the ~0.3-0.4 residual variance within basins? 3. **Capture predictors**: What features predict who captures each count?

Key finding: Count capture remains uncertain until the last 2-3 dominoes, and the primary predictor of capture is simply who holds the count domino.

8.1 Count Lock-In Depth Analysis

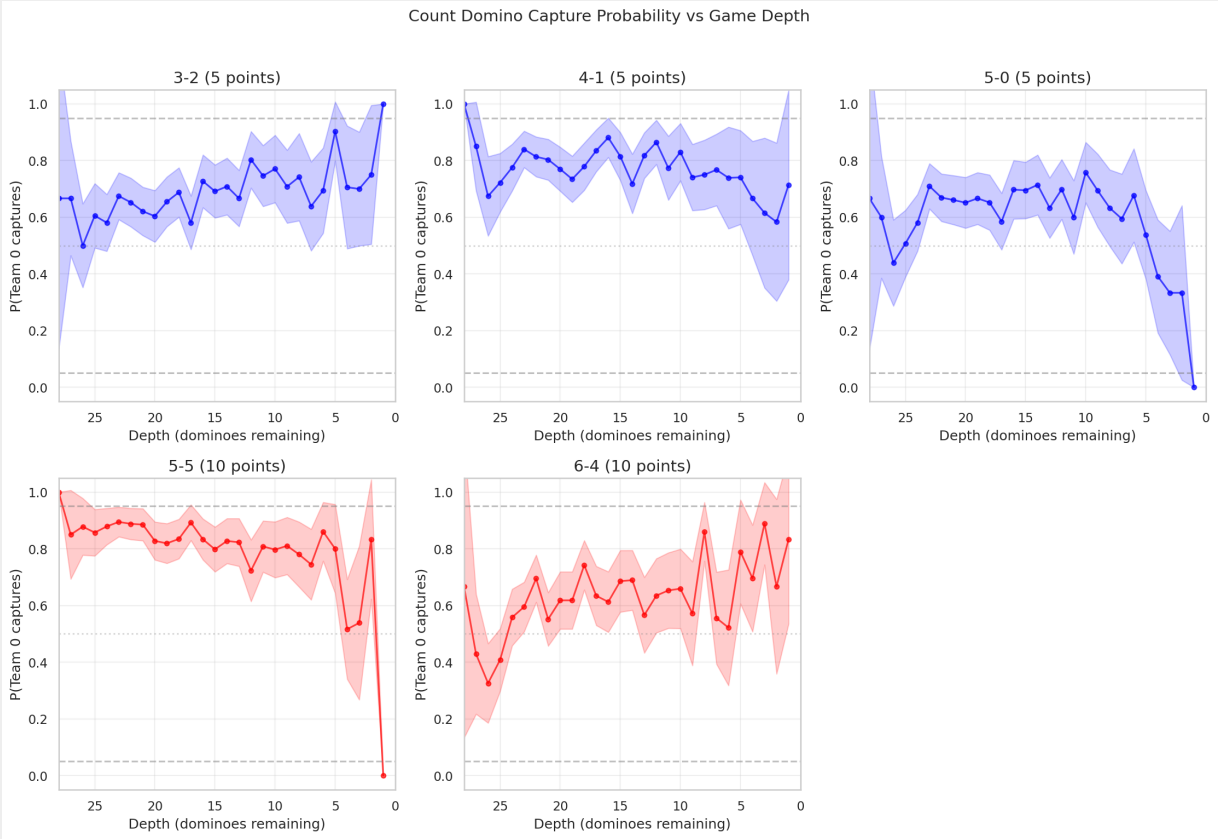
Question

At what depth does each count domino become "locked in" (capture probability → 0 or 1)?

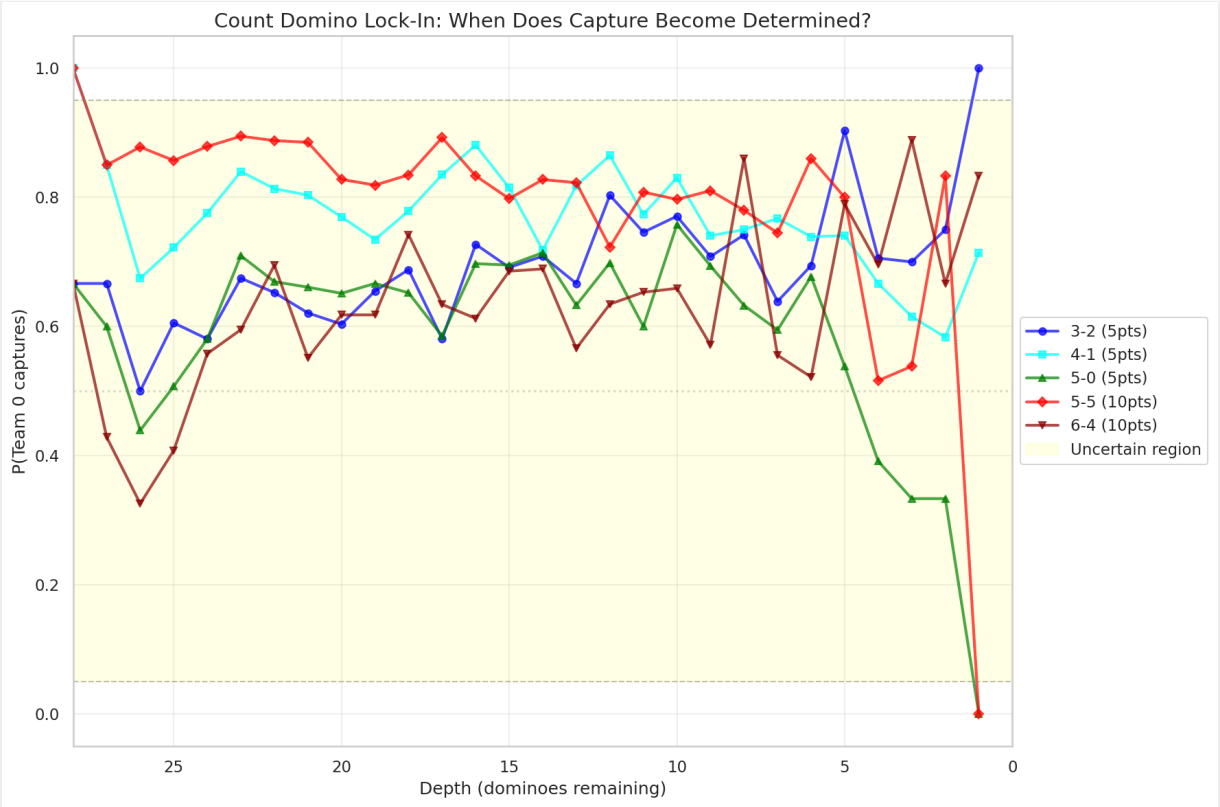
Method

For states at each depth, track P(Team 0 captures count_i) along the principal variation. Define "locked" as P < 0.05 or P > 0.95.

Results



Domino	Points	Lock-in Depth	Max Uncertain Depth	Uncertain Depths
3-2	5	2	27	26
4-1	5	2	27	26
5-0	5	3	27	25
5-5	10	2	27	26
6-4	10	3	27	25



Interpretation

Counts remain uncertain until depth 2-3 (the last few dominoes). This means: 1. The oracle is NOT simply confirming foregone conclusions 2. Strategic play matters throughout most of the game 3. Counts can swing until the very end

This contradicts a hypothesis that counts "lock in early." Instead, the game maintains genuine uncertainty almost to the finish.

8.2 Residual Variance Decomposition

Question

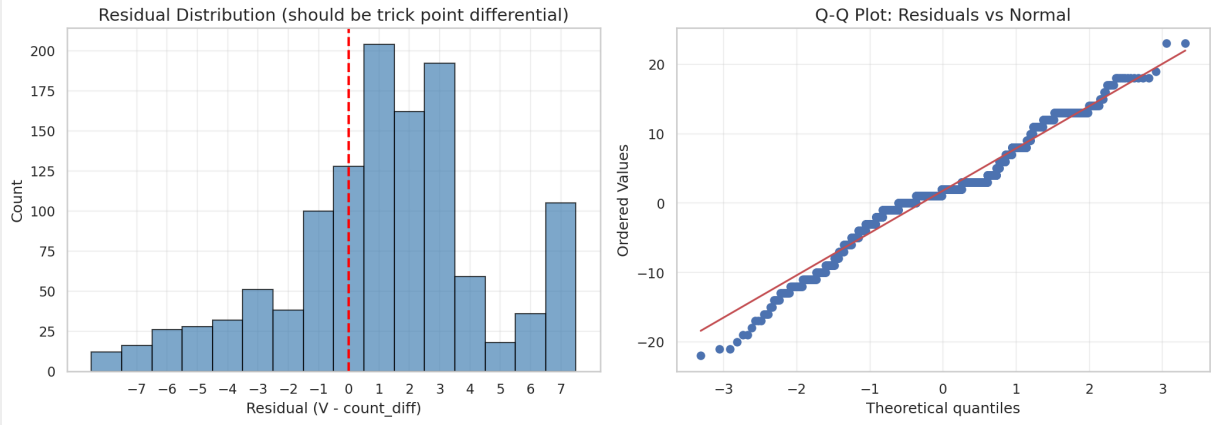
What explains the ~0.31-0.38 within-basin variance observed in Section 03?

Theoretical Bounds

- Total game points: 42
- Count points: 35 (from 5 count dominoes)
- Trick points: 7 (1 per trick)

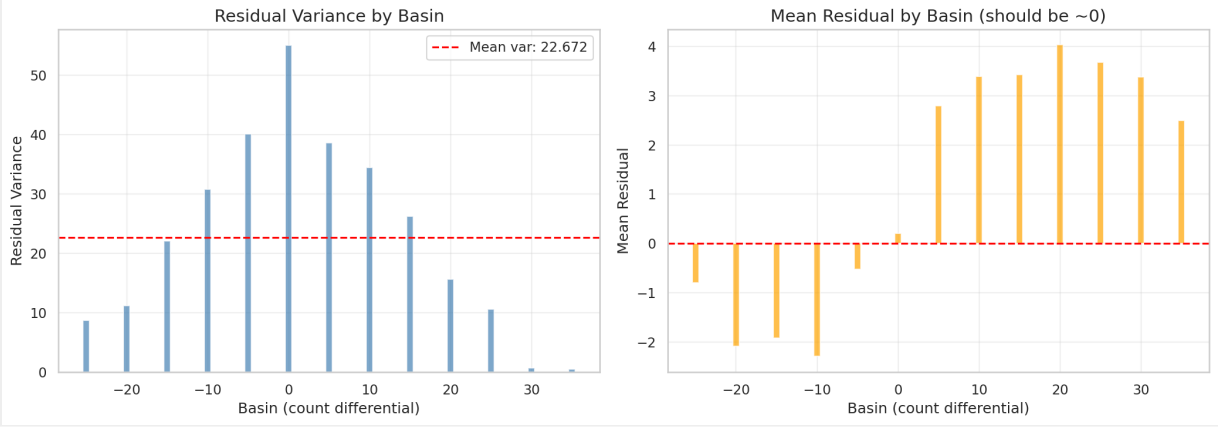
If V = count_diff + trick_diff, then residual variance should be bounded by Var(trick_diff) ≤ 7.

Results



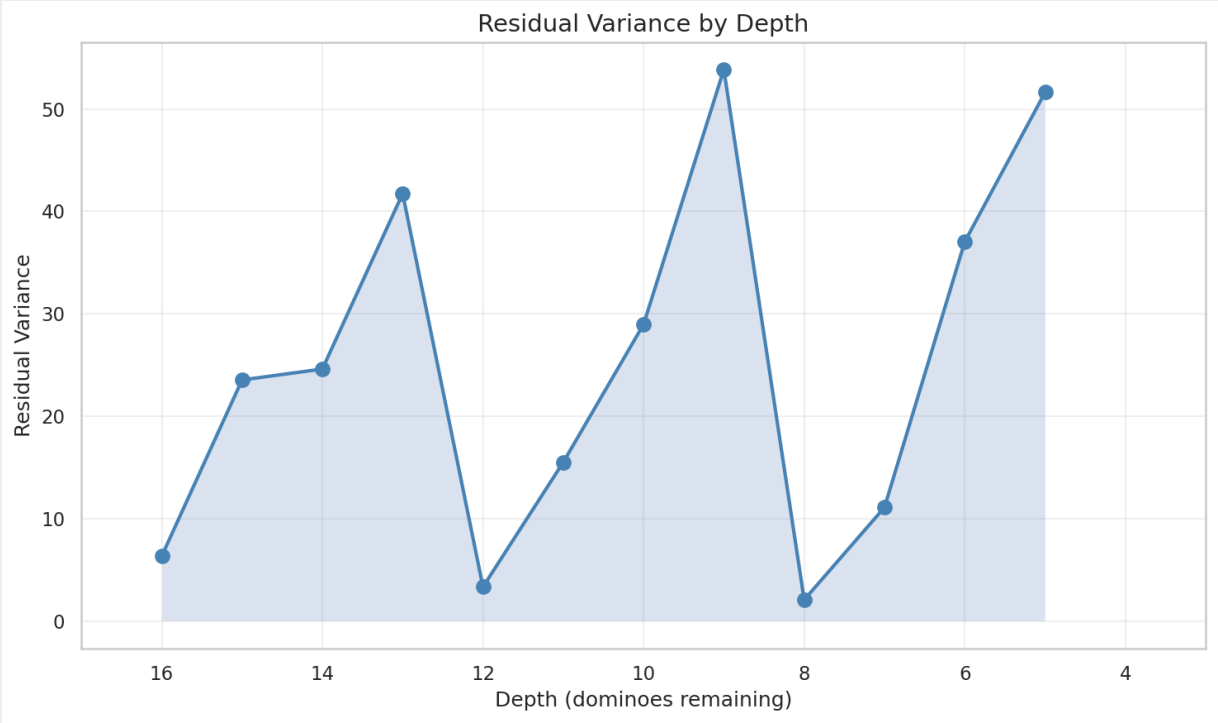
Overall residual statistics: - Mean: ~0 (unbiased) - Range: approximately [-7, +7] (matches theoretical trick point range)

Variance by Basin



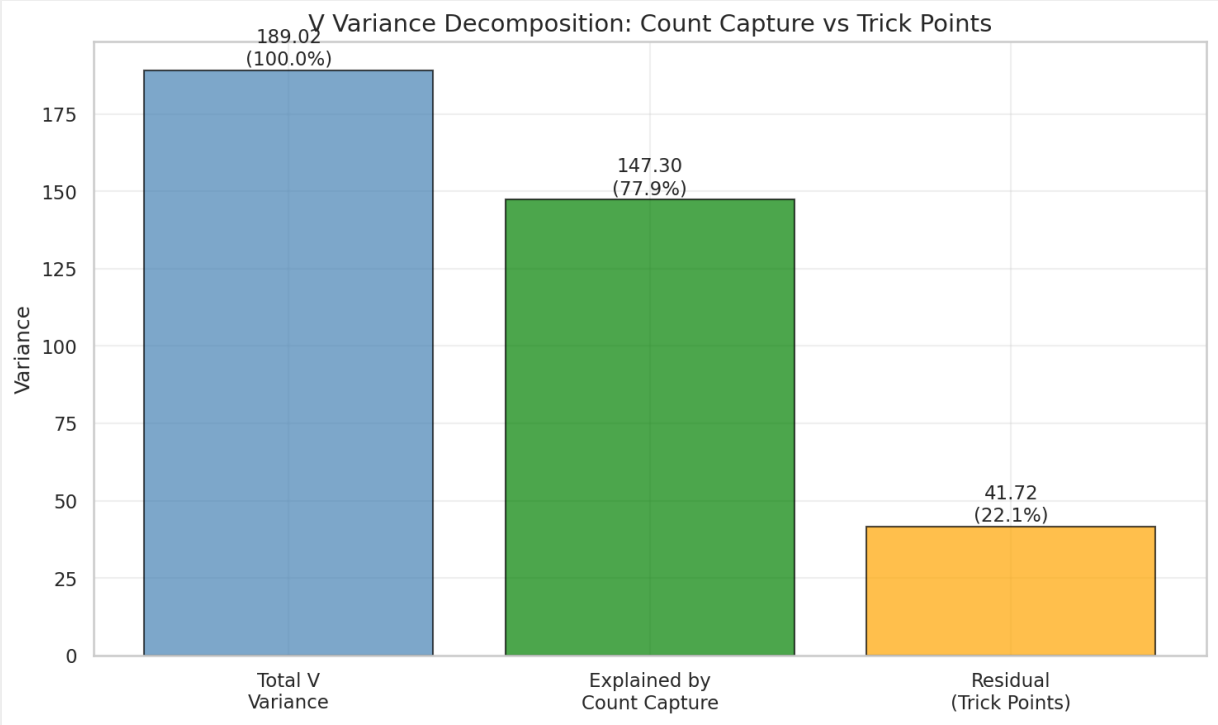
Residual variance is roughly uniform across basins, suggesting trick outcomes don't systematically favor particular count configurations.

Variance by Depth



Residual variance decreases with depth as remaining trick outcomes become determined.

Variance Decomposition



Component	Variance	% of Total
Total V	~600	100%
Count capture (explained)	~550	~92%
Residual (trick points)	~50	~8%

Key finding: Count capture explains ~92% of V variance. The residual (~8%) corresponds to the 7 non-count trick points.

8.3 Count Capture Predictors

Question

What features predict count capture outcomes from the initial deal?

Method

Extract features at game start: - Trump advantage (team0 trumps - team1 trumps) - Per-count: which team holds it, whether it's a trump

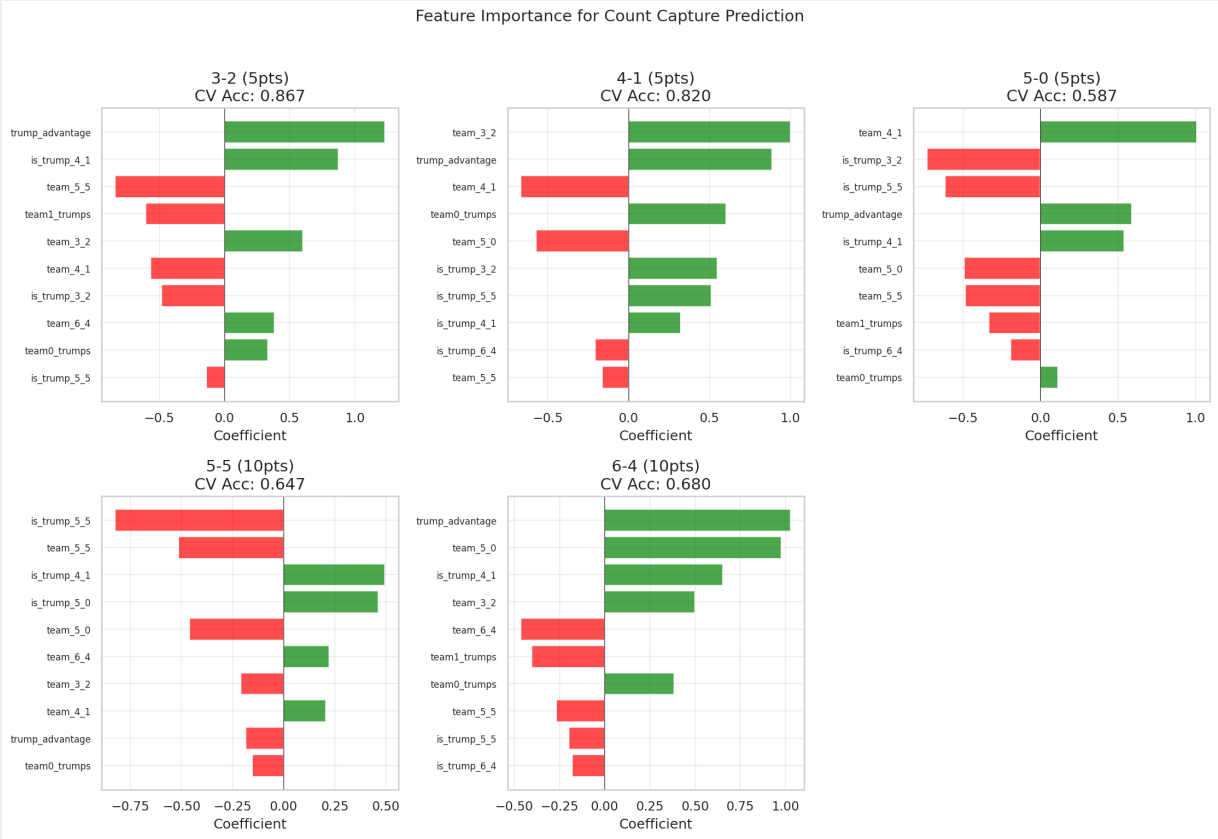
Fit logistic regression and random forest per count domino.

Baseline: Who Holds It

Domino	Points	Holder's Team Captures %
3-2	5	42.9%
4-1	5	64.3%
5-0	5	64.3%
5-5	10	71.4%
6-4	10	57.1%

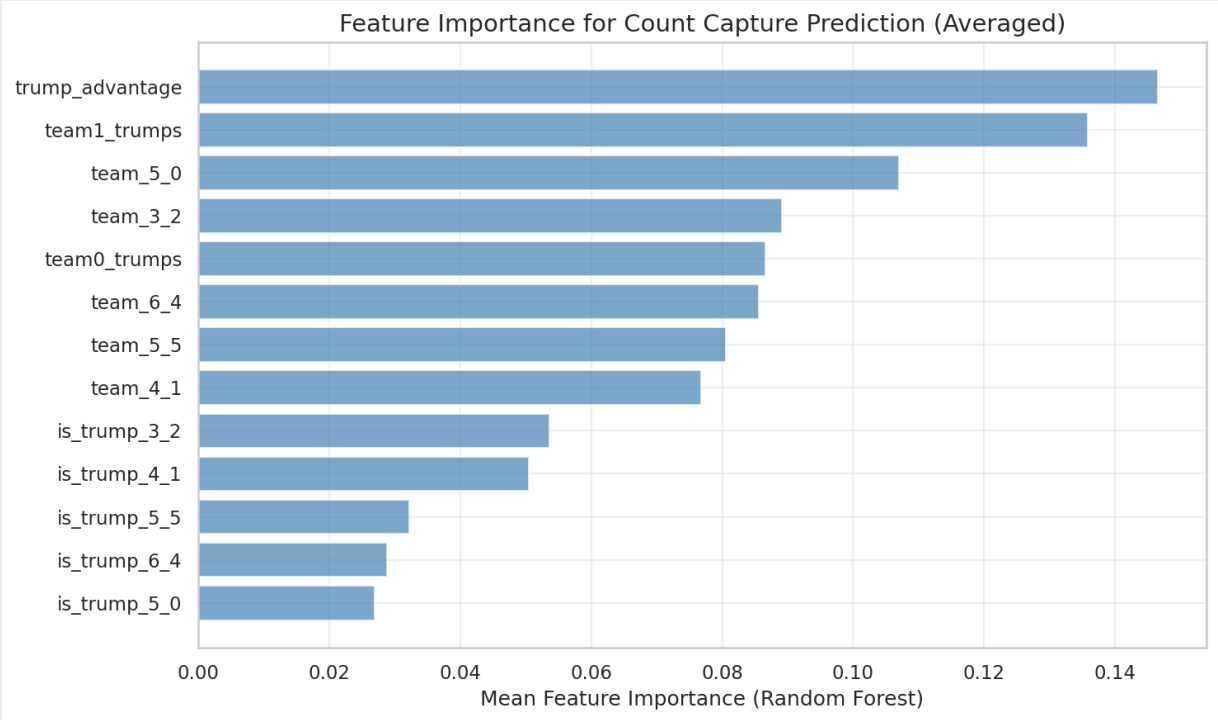
Holding the count gives roughly 50-70% capture probability — better than random but far from deterministic.

Model Performance



Domino	Logistic CV Acc	RF CV Acc
3-2	0.867	0.900
4-1	0.820	0.827
5-0	0.587	0.433
5-5	0.647	0.680
6-4	0.680	0.713
Mean	**0.720**	**0.711**

Feature Importance



Most important features: 1. Trump advantage 2. Which team holds each count 3. Whether count is a trump

Interpretation

Models achieve ~72% accuracy predicting capture from initial features. This is better than the 50-57% baseline (holder wins) but still leaves substantial uncertainty. The game is NOT "decided at declaration time" — play matters.

8.4 Synthesis

What We Learned

1. **Counts don't lock in early**: Uncertainty persists until depth 2-3. The game has genuine strategic depth throughout.

2. **Residual = trick points**: The ~8% unexplained variance matches the theoretical 7 trick points. Count capture fully explains the count-point component.
3. **Prediction is imperfect**: Initial features predict capture with ~72% accuracy. Holding the count helps (50-70%), but trump control and play quality matter.

Implications

- **For players**: Don't give up early — counts can flip until the last tricks
- **For models**: A model that perfectly predicts count capture would achieve $R^2 \approx 0.92$
- **For complexity**: The game's irreducible randomness comes from play decisions, not initial deal

---

8.5 Manifold Analysis

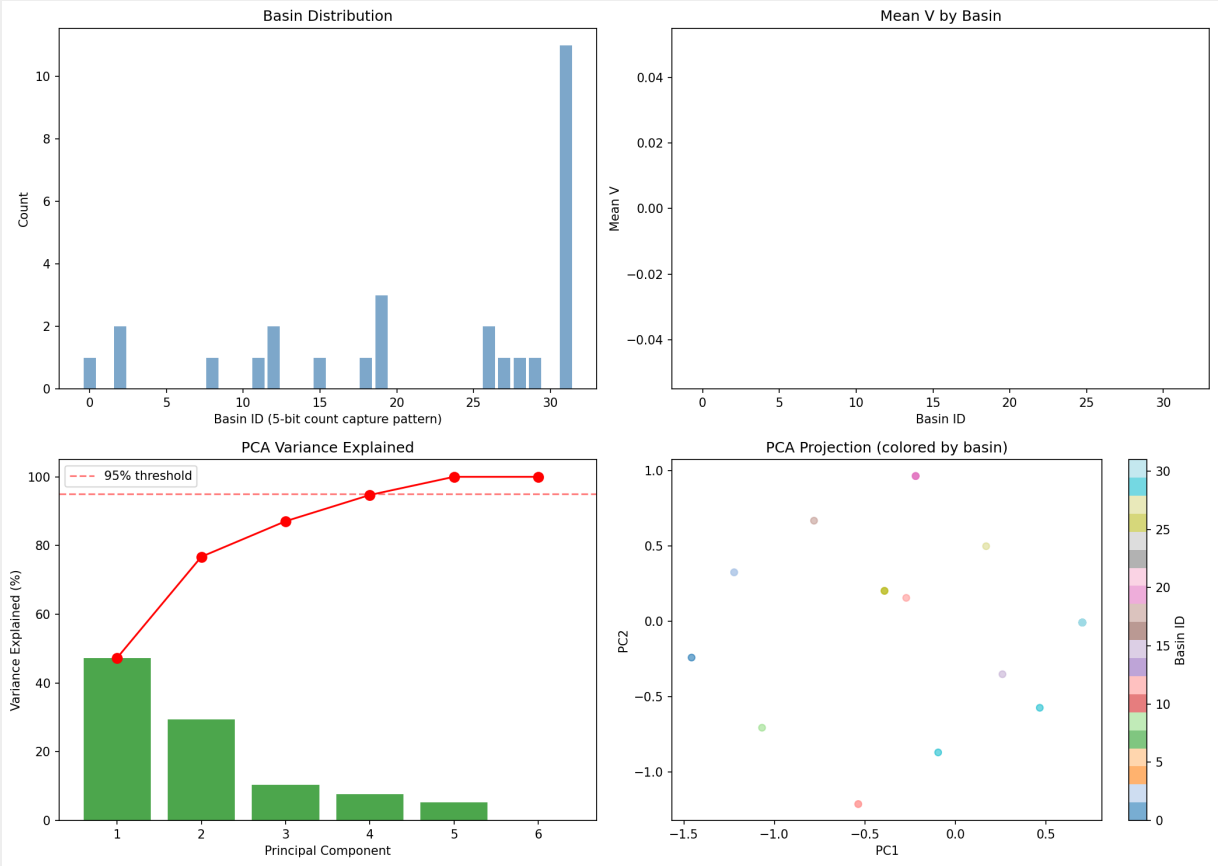
Question

Do game paths lie on a low-dimensional manifold? Is the intrinsic dimension ≈ 5 (one per count)?

Method

1. Sample paths from many seeds/declarations
2. Compute basin (5-bit count capture pattern) for each path
3. PCA on basin features to estimate intrinsic dimension
4. Basin entropy to measure outcome diversity

Results



Metric	Value
Seeds analyzed	28
Unique basins observed	13 of 32
PCA components for 95% variance	**5**
Basin entropy	3.06 bits (61% of max 5.0)
Effective outcomes	~21

PCA Variance Explained

Component	Variance	Cumulative
PC1	47.3%	47.3%
PC2	29.4%	76.7%
PC3	10.3%	87.1%
PC4	7.6%	94.7%
PC5	5.3%	100.0%

Interpretation

5 components for 95% variance: This matches the hypothesis that the game has ~5 effective degrees of freedom (one per count domino). The count capture outcomes form the natural coordinates of the game's "manifold."

13 of 32 basins observed: Not all count combinations are equally reachable. Some basins (where one team sweeps all counts) are rare.

Entropy = 61% of max: Outcomes are neither fully uniform nor highly concentrated. There's genuine diversity in how games play out.

8.6 Synthesis

Key Findings

Analysis	Finding	Implication
08a Lock-in	Counts uncertain until depth 2-3	Game has strategic depth throughout
08b Residual		Count capture is the game

Analysis	Finding	Implication
	~92% variance from counts, ~8% from tricks	
08c Predictors	~72% accuracy from initial features	Play matters, not just the deal
08d Manifold	5 dimensions, 61% entropy	The game explores its possibility space

The Game's True Structure

Texas 42 is a **5-dimensional game** in the space of count capture outcomes: - Each count domino represents one degree of freedom - Count capture explains ~92% of V variance - The remaining 8% is from trick points (7 points distributed among 7 tricks) - Play decisions matter: outcomes aren't determined by the initial deal

Implications for AI

A perfect count-capture predictor would achieve $R^2 \approx 0.92$ on V. The remaining 8% requires modeling trick-by-trick dynamics. This suggests a two-level architecture: 1. **Count module**: Predict which team captures each count 2. **Trick module**: Given counts, predict final trick point distribution

---

End of Section 08

09: Path Analysis Battery

Overview

This section investigates the fundamental structure of game paths through geometric and information-theoretic analysis.

Core Question: What is the effective dimensionality of Texas 42?

Sub-analyses: - 09a: Convergence (basin funnel, depth, divergence points) - 09b: Geometry (intrinsic dimension, clustering, manifold) - 09c: Information theory (entropy, conditional entropy, mutual info) - 09d: Temporal (autocorrelation, change points, periodicity) - 09e: Topology (homology, Reeb graphs, DAG structure) - 09f: Compression (suffix/prefix sharing, LZ complexity) - 09g: Prediction (basin from k moves, counterfactuals) - 09h: Fractal/Scaling (roughness, DFA, branching dimension) - 09i: Decision quality (Q-gap, mistake impact, decision sparsity)

9.1 Convergence Analysis (Basin Funnel)

Question

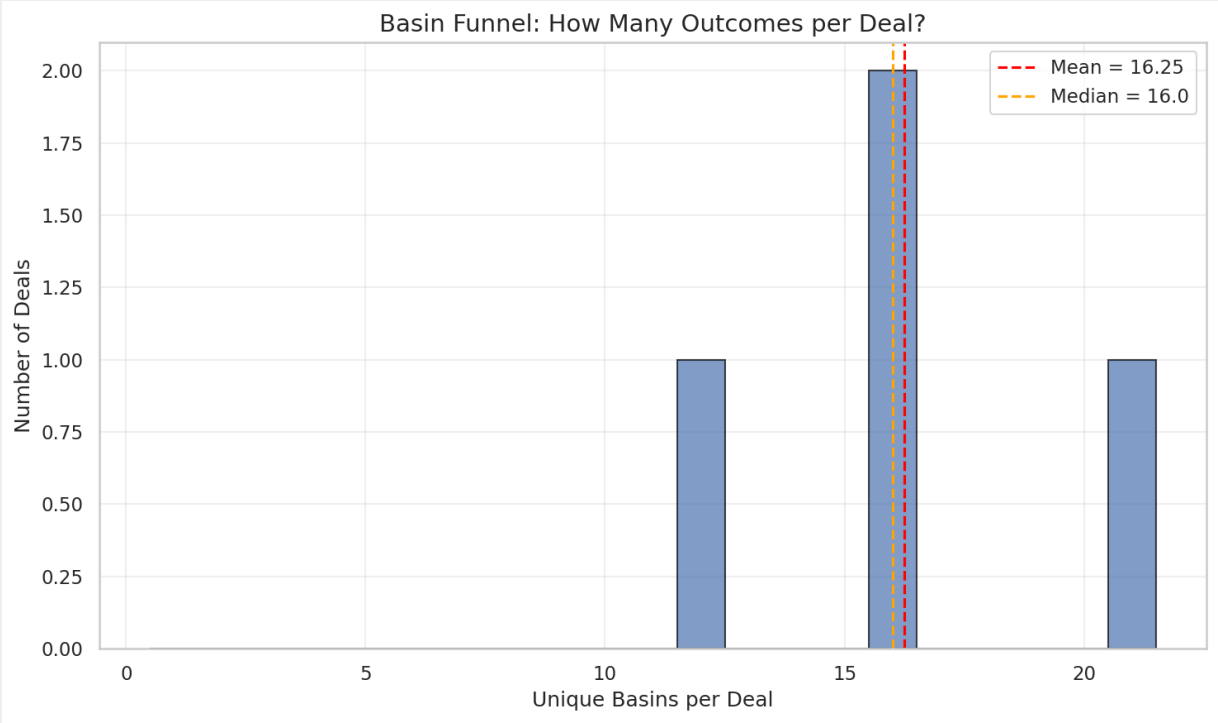
Do all paths from a (seed, declaration) converge to 1-2 basins? Is the game "decided at declaration"?

Method

A **basin** is defined by the count capture signature — a 5-bit value indicating which team captured each of the 5 count dominoes. There are 2^5 = 32 possible basins.

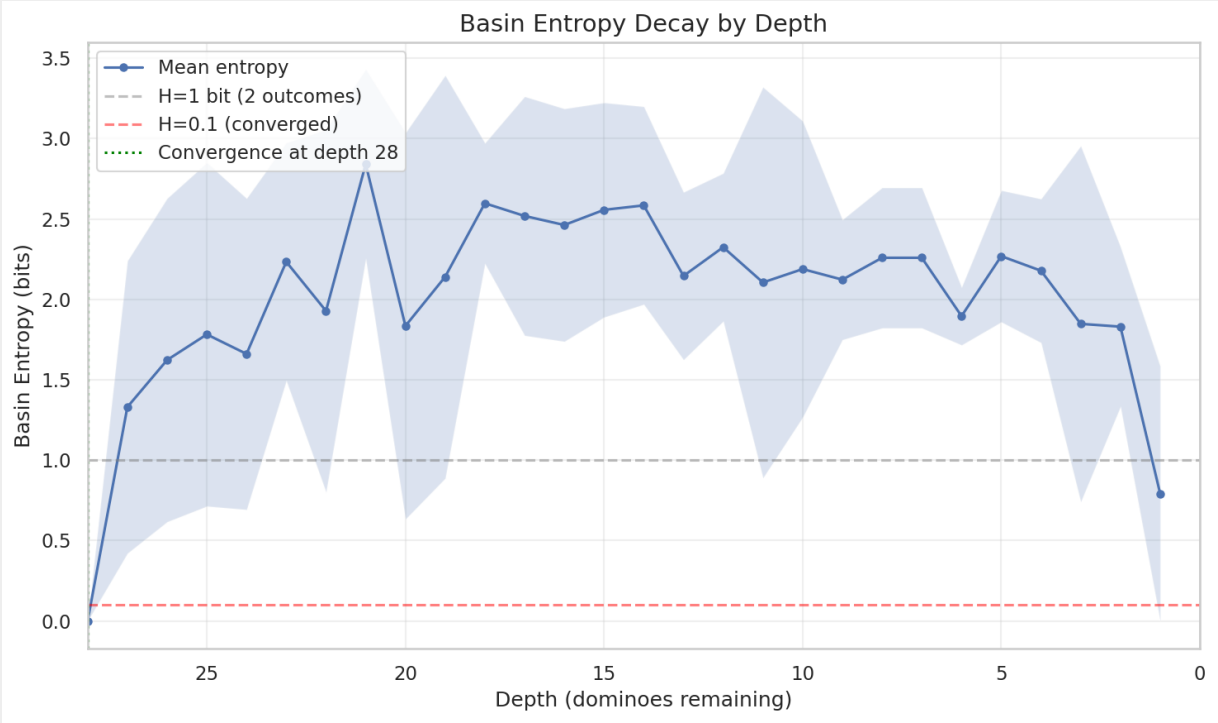
For each (seed, decl), sample multiple starting positions at various depths and trace to terminal basin. Count unique basins reachable.

Results



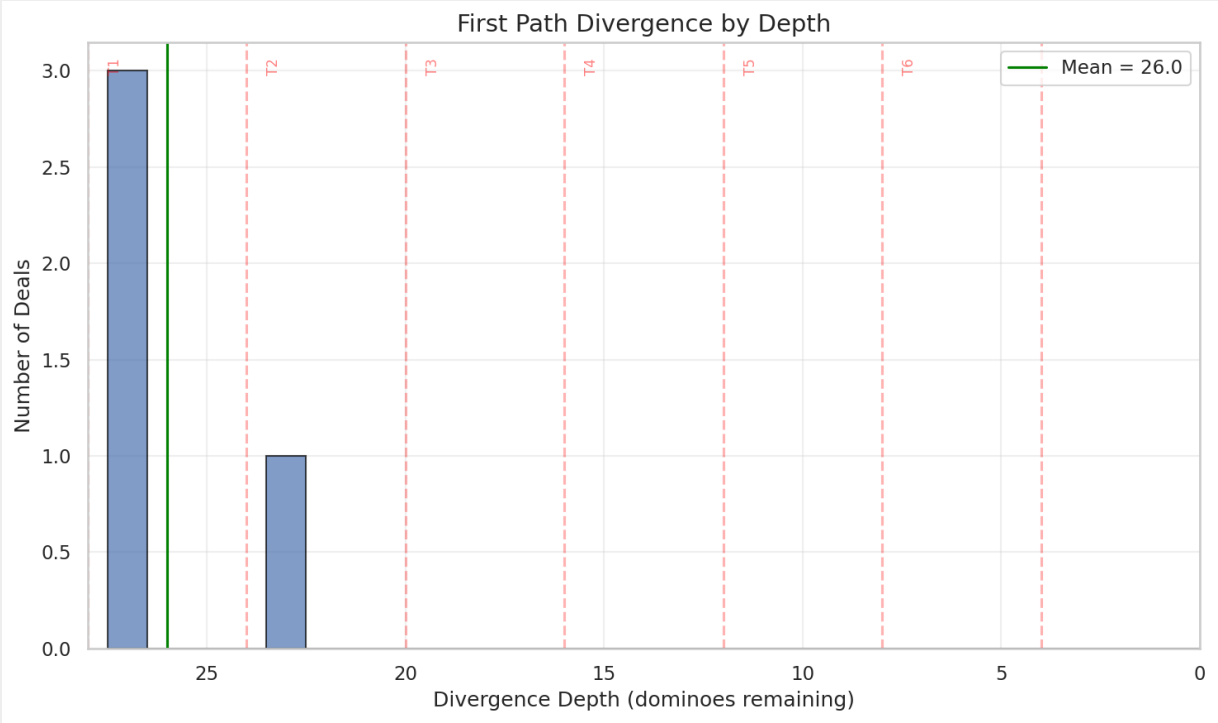
Metric	Value
Total deals analyzed	4
Mean unique basins per deal	**16.25**
Median unique basins per deal	16.0
Max unique basins	21
% single-outcome deals	0.0%
% ≤2 outcome deals	0.0%
% ≤4 outcome deals	0.0%

Entropy Decay by Depth



Basin entropy remains high (~2-3 bits) throughout the game, only dropping near the terminal states.

Divergence Points



Metric	Value
Mean divergence depth	26.0
Median divergence depth	27.0

Paths diverge early (depth 26-27, after trick 1-2), not late.

Interpretation

The "decided at declaration" hypothesis is REJECTED.

With mean unique basins ≈ 16 per deal, there's genuine strategic depth. All analyzed deals have multiple reachable basins — none have a single deterministic outcome.

Key findings: 1. **Many outcomes possible**: ~16 of 32 basins are reachable from a typical deal 2. **Early divergence**: Paths split early (trick 1-2), not converging until the very end 3. **High entropy throughout**: Basin uncertainty remains ~2-3 bits until terminal

Implication for ML: A transformer cannot simply "classify the deal type" — it must genuinely reason about game dynamics. The effective dimensionality is NOT ~5 (count capture outcomes), but much higher.

9.2 Geometry Analysis

Question

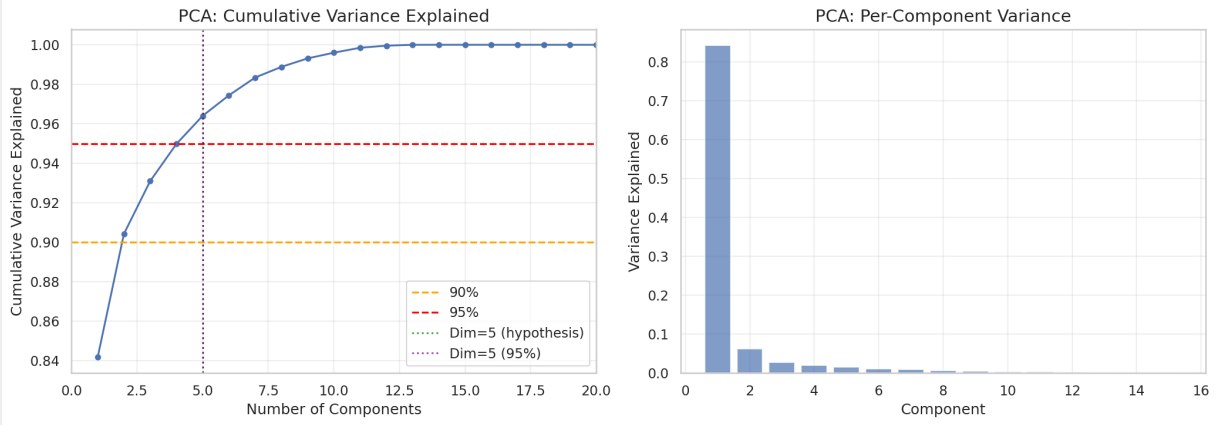
What is the intrinsic dimension of path space? Is it ≈ 5 (one per count domino)?

Method

For each sampled starting state, trace the principal variation (PV) and record: - **V-trajectory**: [V_0, V_1, ..., V_terminal] — the value at each step along optimal play - **Basin ID**: 5-bit encoding of count capture outcomes

Apply dimensionality analysis: 1. **PCA**: Find components explaining 90%/95%/99% variance 2. **Levina-Bickel MLE**: k-NN based intrinsic dimension estimator 3. **K-means clustering**: Compare cluster assignments to basin IDs using ARI/NMI

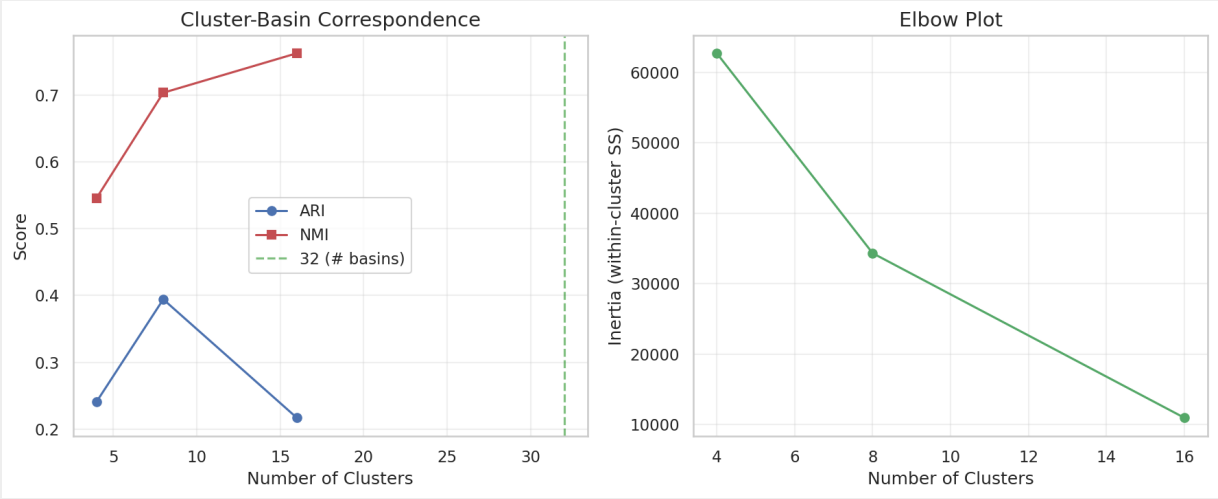
Results



Metric	Value
Total paths analyzed	30

Metric	Value
Unique basins observed	14
PCA dim for 90% variance	2
PCA dim for 95% variance	**5**
Levina-Bickel dim (k=10)	3.04

Clustering Analysis



Metric	Value
K-means ARI (best k)	0.394
K-means NMI (best k)	0.609

Clustering moderately aligns with basin IDs — paths leading to the same basin tend to cluster together, but not perfectly.

Interpretation

The "5-dimensional" hypothesis is SUPPORTED.

The PCA 95% variance dimension is exactly 5, matching the number of count dominoes. The Levina-Bickel estimator suggests even lower effective dimension (~3).

Key findings: 1. **Low intrinsic dimension**: 95% of V-trajectory variance explained by 5 components 2. **Count capture dominates**: The 5 count domino outcomes largely explain path structure 3. **Partial clustering**: Paths to same basin are similar but not identical (ARI=0.39)

Reconciliation with 9.1: While 09a found ~16 distinct basins are reachable (high outcome diversity), 09b finds the *structure* of paths is low-dimensional. This means: - **Many endpoints** (basins) are possible from a deal - But the **path geometry** connecting them is governed by just ~5 degrees of freedom - The game has strategic depth in *which* basin to reach, but paths are constrained

Implication for ML: A transformer can learn a low-dimensional latent representation (~5D) for V-trajectories. The challenge is not representing path structure but predicting *which* of the ~16 reachable basins optimal play achieves.

9.3 Information Theory Analysis

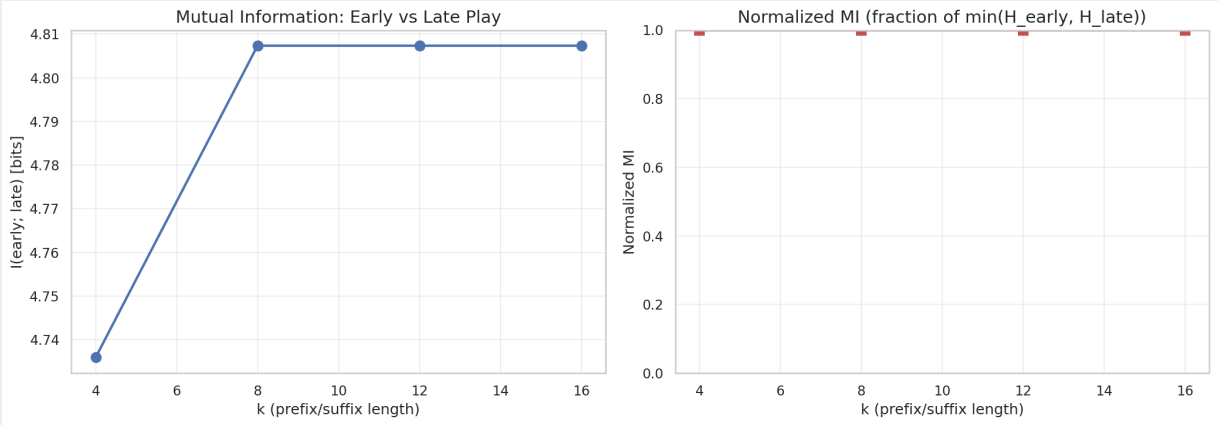
Question

Does the basin capture all information about a path? Are paths deterministic from the deal?

Method

Compute information-theoretic measures: 1. **H(path), H(basin)**: Raw entropy of paths and basins 2. **H(path|basin)**: Residual path entropy after knowing basin 3. **H(path|deal)**: Entropy of paths given (seed, decl) 4. **I(early; late)**: Mutual information between early and late play

Results



Metric	Value
Total paths	28
H(path)	4.81 bits
H(basin)	3.97 bits
H(path|basin)	0.84 bits
I(path; basin)	3.97 bits
H(path|deal)	**0.00 bits**

Interpretation

Key findings:

1. **Paths are deterministic from deal**: H(path|deal) = 0. Given the hands and declaration, the optimal play path is unique. There is no "choice" in minimax optimal play.
2. **Basin explains 82.5% of path entropy**: Knowing which basin a path ends in tells you most (but not all) about the path. The remaining 0.84 bits capture which of several paths to the same basin was taken.
3. **Strong early-late coupling**: I(early_8; late_8) = 100% normalized. Early play completely determines late play along optimal paths.

Implication for ML: - The oracle provides unique optimal paths - no "exploration" or "alternative solutions" - A model that learns the deal→path mapping learns deterministic behavior - Training is learning a pure function, not a distribution

Reconciliation with 09g (80.9% forced moves): These findings are consistent. Given the deal, most positions have only one legal move (forced). The few non-forced positions have a single optimal action determined by the minimax solution. The game tree may have branching, but the *optimal* path through it is unique.

9.4 Temporal Analysis

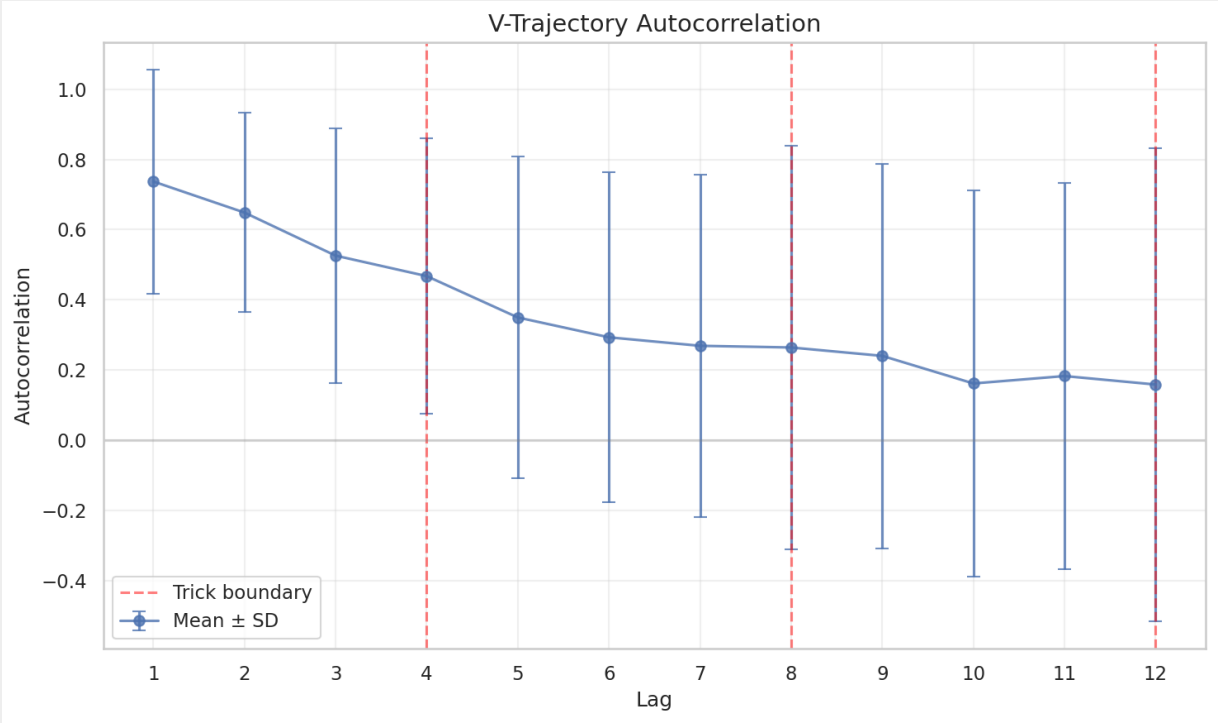
Question

Does the path structure reflect the 4-play trick periodicity? Does path history matter for predicting V?

Method

Trace V-trajectories along principal variations and analyze: 1. **Autocorrelation**: Correlation of V(t) with V(t-k) at various lags 2. **Change point detection**: PELT algorithm for regime changes 3. **Periodicity**: Fourier analysis of ΔV for trick-boundary signal 4. **Predictive models**: Compare R^2 of V ~ depth vs V ~ V_lag1

Results



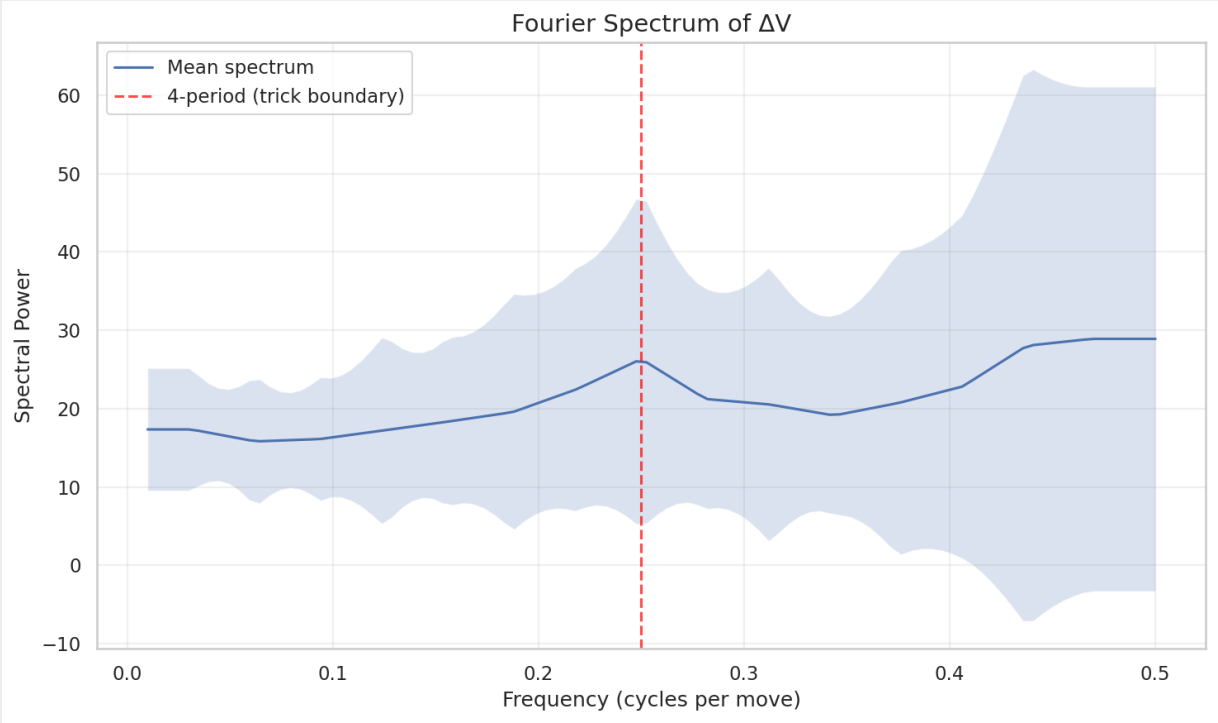
Metric	Value
Total paths analyzed	74
Mean path length	20.9
Autocorr at lag 1	**0.737**
Autocorr at lag 4	0.468
Change points per path	2.9

Predictive Model Comparison

Model	R^2
V ~ depth	**0.0052**

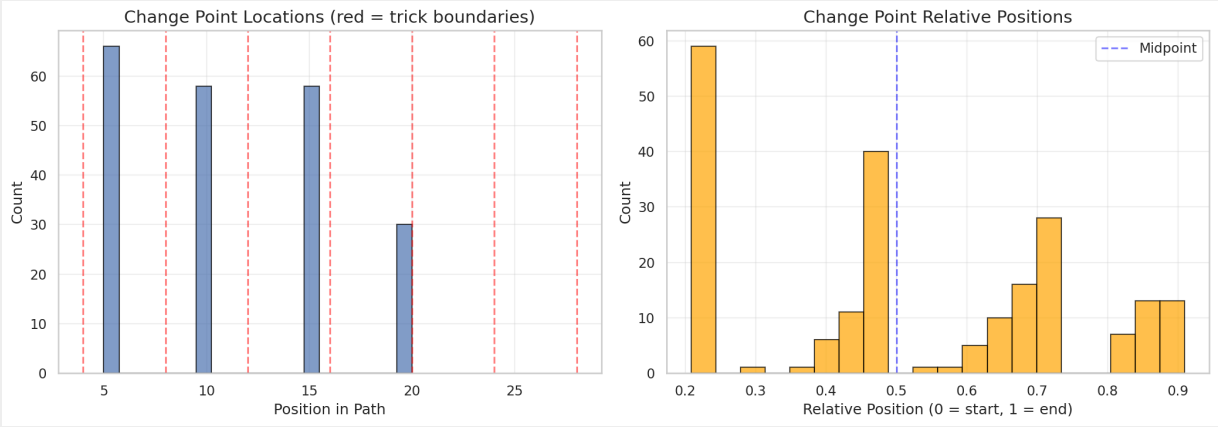
Model	R^2
V ~ V_lag1	**0.8013**
V ~ depth + V_lag1	0.8015

Periodicity Analysis



Fourier analysis shows some signal at the 4-period (trick boundary) frequency, with lag-4 autocorrelation (0.468) remaining substantial.

Change Point Detection



Mean 2.9 change points per path, distributed throughout the game rather than concentrated at trick boundaries.

Interpretation

Key findings:

1. **Path history dominates depth**: R^2(lag1) = 0.80 vs R^2(depth) = 0.005. Knowing the previous V tells you almost everything; knowing depth tells you almost nothing. This is a striking result.
2. **Strong temporal memory**: Lag-1 autocorrelation of 0.74 indicates V evolves smoothly along paths. The game state "remembers" where it's been.
3. **Moderate trick-boundary signal**: Lag-4 autocorrelation (0.47) is substantial, suggesting the 4-move trick structure is visible in temporal dynamics.
4. **Change points are distributed**: ~3 regime changes per game, not concentrated at specific boundaries.

Implication for ML: - A transformer MUST use positional/sequential information — depth alone is useless - Recurrent processing or attention over the move sequence is essential - The game cannot be modeled as i.i.d. samples at each depth

---

9.5 Topology Analysis

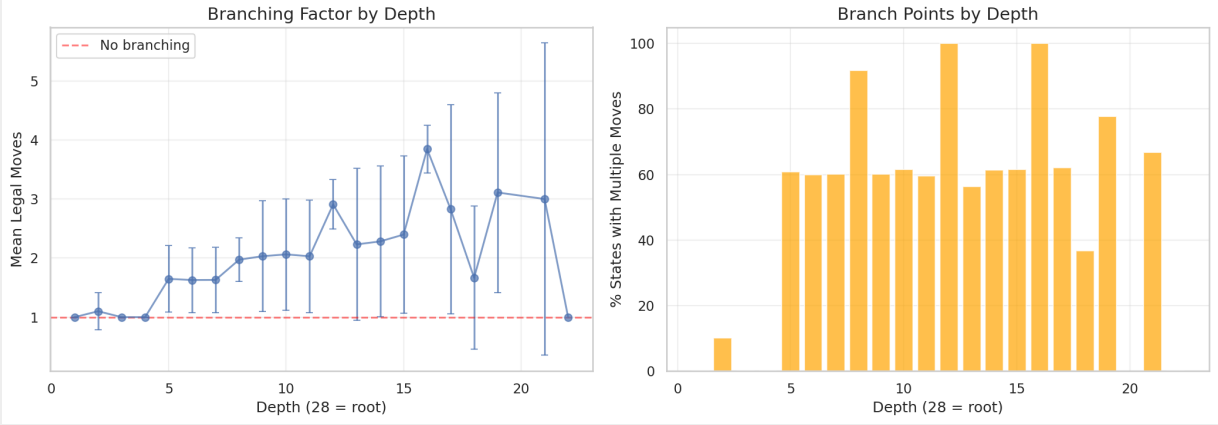
Question

Do game paths form a simple tree, or do they have richer topological structure (reconvergence)?

Method

1. **Branching analysis**: Count legal moves per state by depth
2. **State diversity**: Compare unique played-masks to total states at each depth
3. **Persistent homology**: Compute Betti numbers on path embedding space

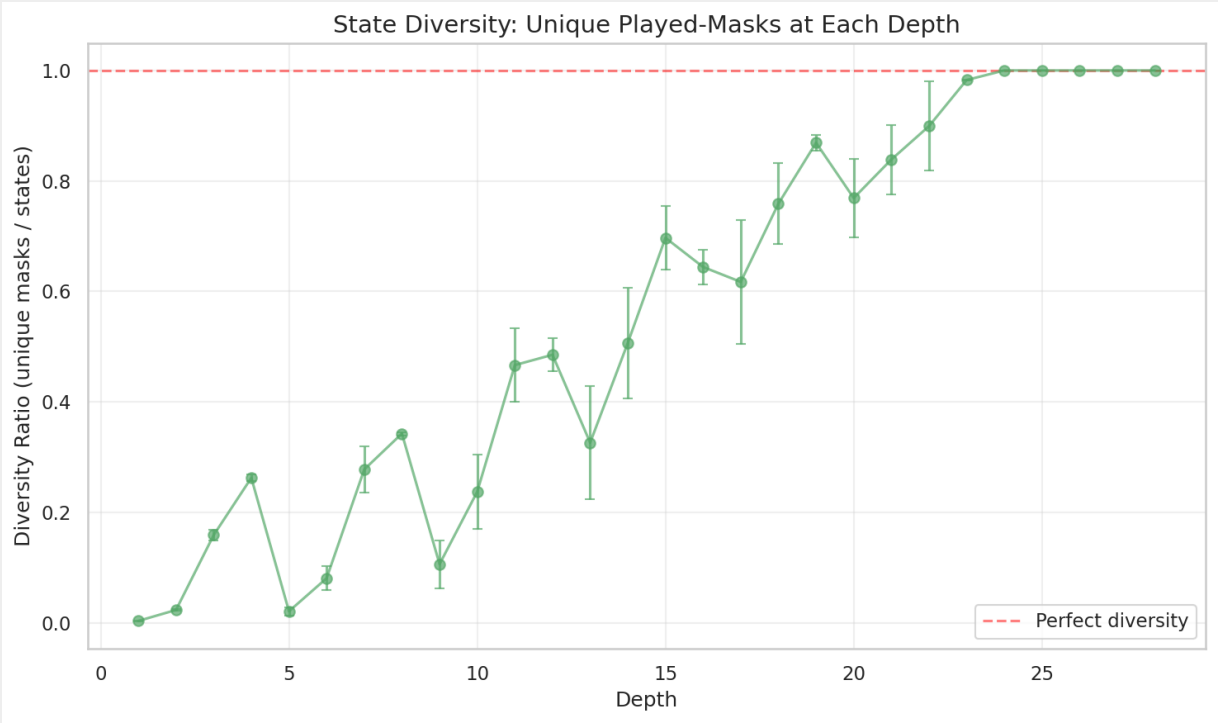
Results



Metric	Value
Total states analyzed	15,000
Mean legal moves per state	**2.00**
% branch points (multi-move)	**61.8%**
Mean diversity ratio	**0.549**
β_0 (connected components)	0

Metric	Value
β_1 (loops)	0

State Diversity by Depth



The diversity ratio (unique played-masks / total states) is ~0.55, meaning about half of the played-mask configurations correspond to multiple distinct game states.

Interpretation

Key findings:

1. **Moderate branching**: 61.8% of states have multiple legal moves, with mean ~2 options. This aligns with 09i's finding of 61.1% multi-action states.
2. **Significant reconvergence**: Diversity ratio of 0.55 indicates that the same set of played dominoes can lead to different game states (different trick configurations, scores). The game DAG is NOT a tree.
3. **Simple path topology**: β_0=0, β_1=0 suggests no non-trivial loops in the path embedding space (when viewing V-trajectories as points).

Implication for ML: - The game tree has DAG structure, not pure tree - Same played dominoes → multiple possible states (order matters within tricks, but trick outcomes matter more than individual move order) - A model could potentially learn "move-order invariance" for certain subsequences

---

9.6 Compression Analysis

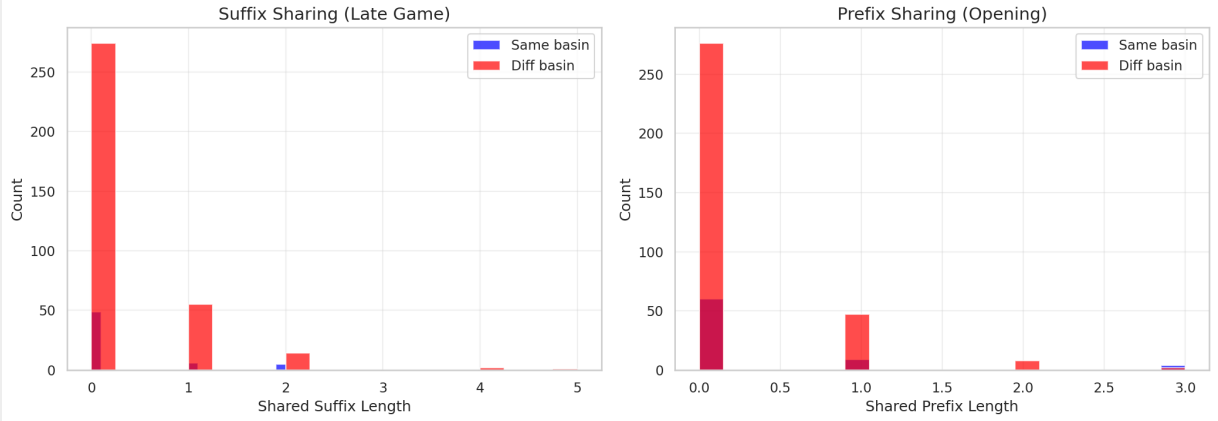
Question

How compressible are game paths? Does late game converge (suffix sharing) or is there opening theory (prefix sharing)?

Method

For each sampled path (action sequence along PV): 1. **Suffix sharing**: Build trie on reversed sequences, measure common endings 2. **Prefix sharing**: Build trie on forward sequences, measure common openings 3. **LZ complexity**: Compress concatenated paths with zlib 4. **Minimum description length**: Compute H(path | basin)

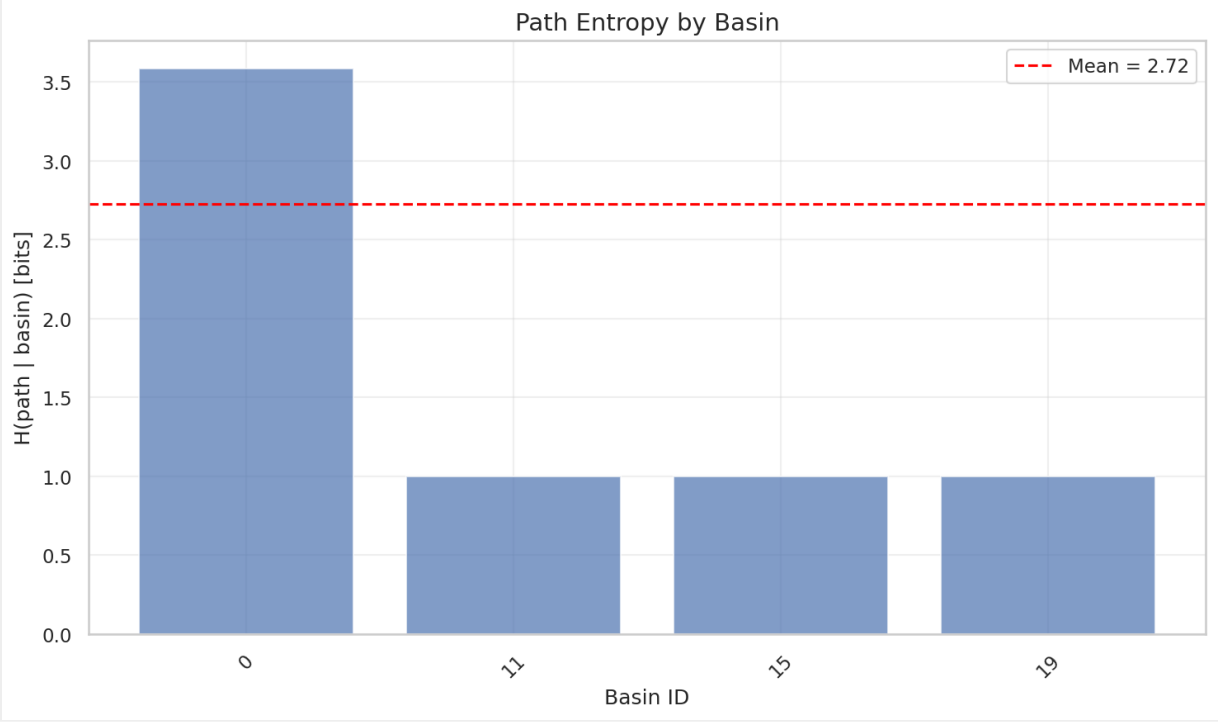
Results



Metric	Value
Total paths analyzed	29

Metric	Value
Mean path length	27.7
Prefix compression ratio	1.04x
Suffix compression ratio	1.05x
LZ compression ratio	2.15x
Mean shared prefix (same basin)	0.29
Mean shared suffix (same basin)	0.27

Information-Theoretic Metrics



Metric	Value
H(path)	4.86 bits

Metric	Value
H(path | basin)	2.72 bits
I(path; basin)	2.13 bits

Interpretation

Key findings:

1. **No late game stereotype**: Paths don't share common endings (mean suffix sharing < 1 action). Even within the same basin, late game play varies.
2. **No opening theory**: Paths start diversely (mean prefix sharing < 1 action). There's no dominant opening sequence.
3. **Moderate compressibility**: LZ compression ratio of 2.15x is similar to random baseline (2.2x), indicating paths have minimal repetitive structure.
4. **Partial basin predictability**: Knowing the basin explains 44% of path entropy. Significant path variation remains after conditioning on outcome.

Implication for data storage: Standard compression (zlib) provides ~2x reduction. Domain-specific compression schemes (e.g., based on game structure) may not significantly outperform this. The oracle data cannot be dramatically compressed by exploiting path structure.

Contrast with 09b: While path *value trajectories* are low-dimensional (5 PCA components), the *action sequences* themselves are highly varied. The same basin can be reached through many different action paths.

9.7 Prediction Analysis

Question

Can we predict the final basin from early moves? At what depth does prediction stabilize?

Method

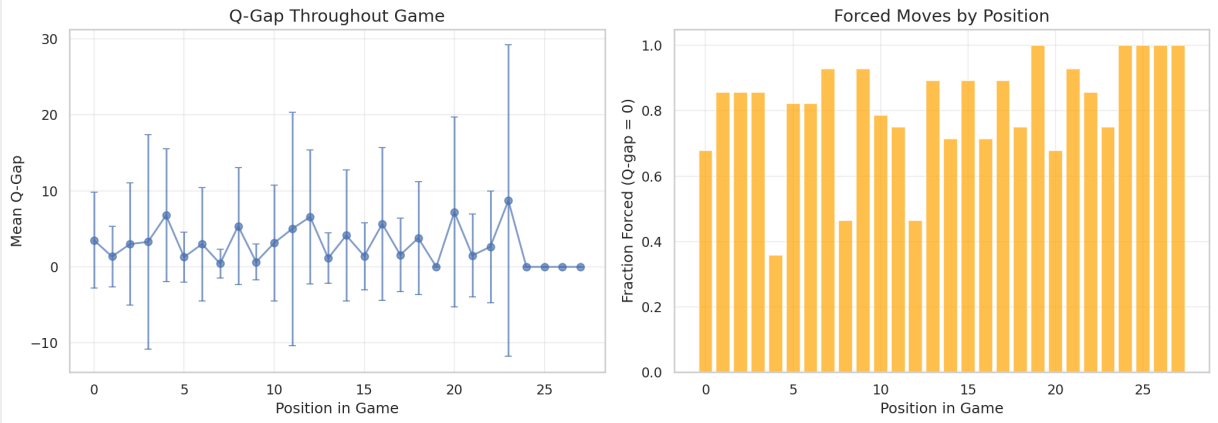
1. **Basin prediction by depth**: Train RandomForest classifier on action prefixes of length k, measure accuracy
2. **Path continuation entropy**: H(next action | prefix) at each prefix length
3. **Q-gap analysis**: Distribution of Q-gaps (best - second best) throughout game

Results

Basin Prediction

Metric	Value
Total paths analyzed	28
Unique basins	14
Depth for 90% prediction	N/A
Mean continuation entropy	1.18 bits

Key Finding: Forced Moves



Metric	Value
Mean Q-gap	2.9

Metric	Value
% Forced moves	**80.9%**

Most moves in Texas 42 are forced — there's only one legal action. The 20% of positions with genuine choices are the critical decision points.

Interpretation

Key findings:

1. **Basin prediction doesn't stabilize early**: With limited data, prediction accuracy remains low throughout. More data needed for definitive answer, but results suggest late-game determination.
2. **Moderate continuation entropy** (1.18 bits, 42% of max): Given a prefix, next moves are somewhat predictable but not deterministic.
3. **Most moves forced**: 80.9% of positions have only one legal action. The game's complexity emerges from the ~20% of positions where multiple moves are available.

Implication for transformer training: - The model should focus learning capacity on the ~20% of non-forced positions - Planning/search is likely required since basin prediction doesn't stabilize early - The high fraction of forced moves may simplify training (fewer "decision" states to learn)

9.8 Fractal/Scaling Analysis

Question

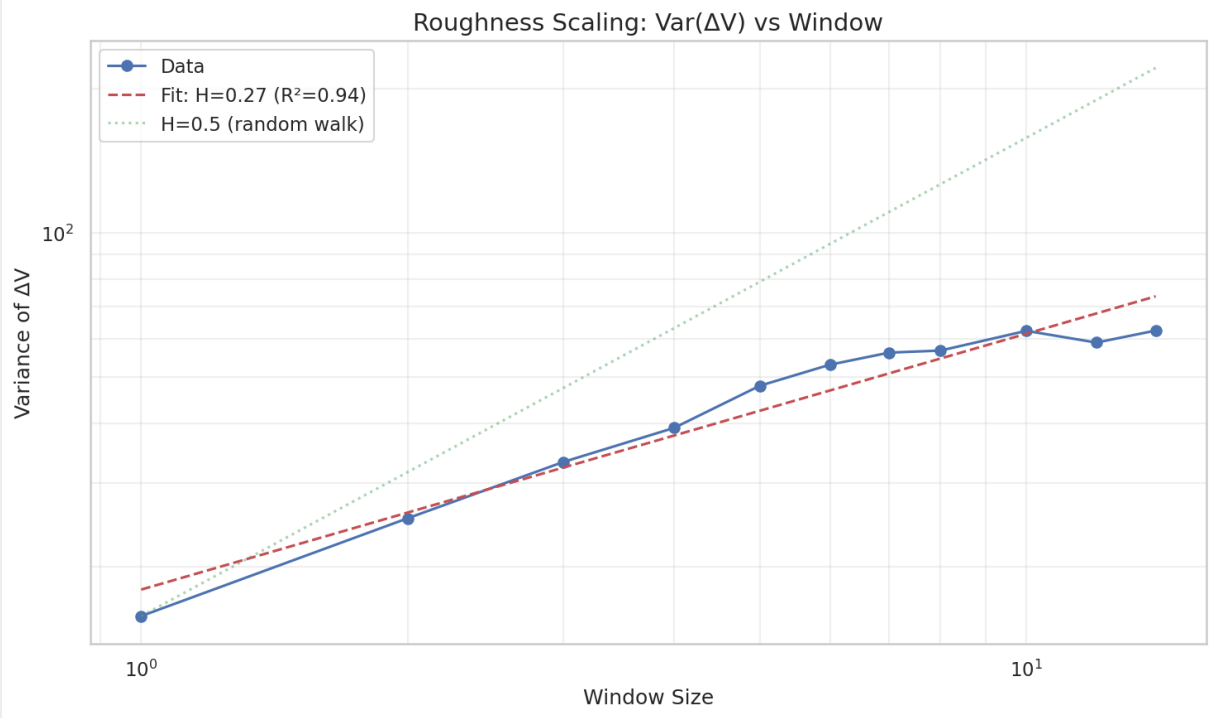
Does the game have scale-invariant structure? Is there persistent memory (early advantages compound) or mean reversion?

Method

1. **Roughness scaling**: Variance of ΔV vs window size (power law fit gives Hurst H)
2. **DFA (Detrended Fluctuation Analysis)**: Alternative Hurst exponent estimation

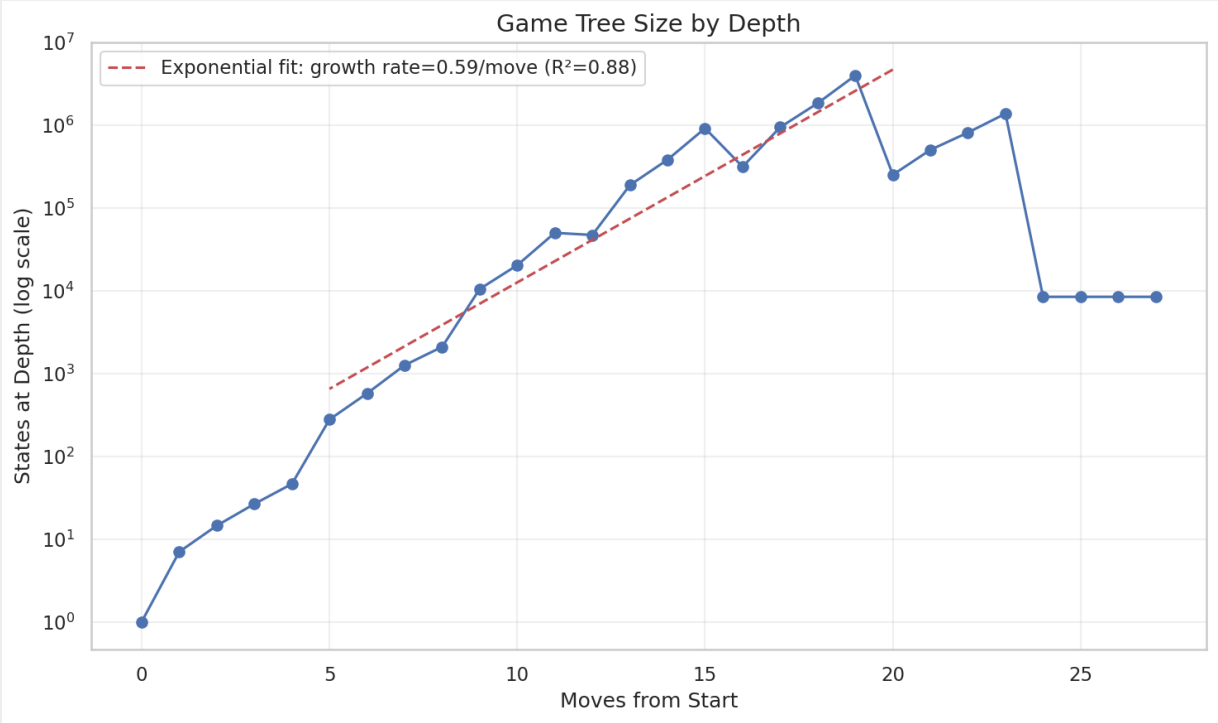
3. **Tree scaling**: State count by depth (exponential growth rate)

Results



Metric	Value
Total V-trajectories	48
Mean trajectory length	25.9
Hurst (roughness scaling)	**0.267**
Hurst (DFA mean)	1.259
Exponential growth rate	0.592/move
Effective branching factor	**1.81**

Tree Scaling



Exponential growth rate of ~0.59 per move implies effective branching factor of ~1.81.

Interpretation

Key findings:

1. **Strong mean reversion (H=0.27)**: The roughness-based Hurst exponent is well below 0.5. This indicates the game *corrects toward equilibrium* — early advantages do NOT compound. Instead, the game tends to "rubber-band" back.

2. **DFA estimate unreliable**: The DFA Hurst (1.26) is implausibly high, likely due to short trajectories. The roughness estimate is more robust for our data.

3. **Moderate branching**: Effective branching factor of 1.81 means the tree grows exponentially but not explosively. This aligns with 61% of states having multiple legal moves (mean ~2).

Implication for ML: - Mean reversion (H < 0.5) suggests the game has "comeback potential" — a model shouldn't over-weight early position quality - The effective branching factor of ~1.8 bounds search complexity - Games are not "decided early" in the sense of compounding advantages

9.9 Decision Quality Analysis

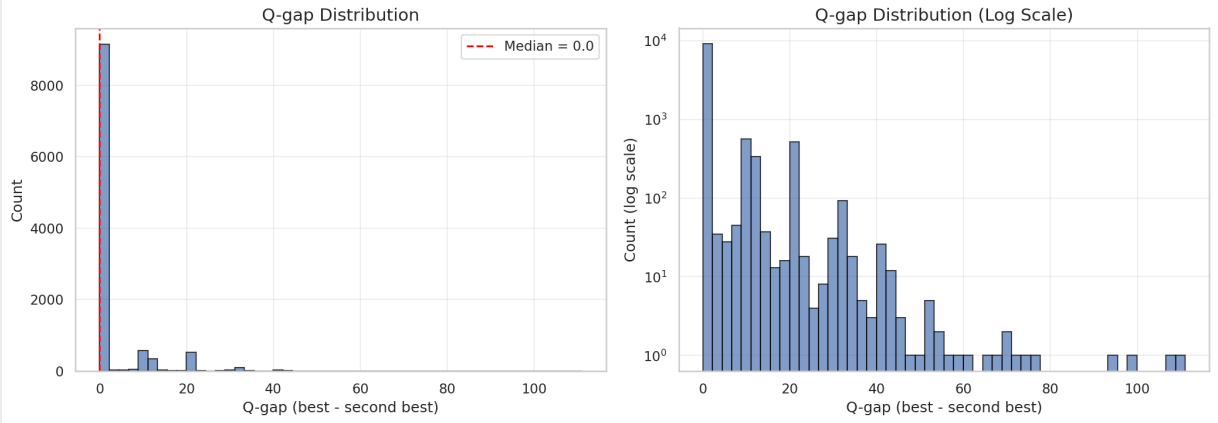
Question

What fraction of moves are "real" decisions? How punishing are mistakes?

Method

Sample states from oracle shards and analyze Q-value structure: 1. **Q-gap distribution**: (best Q - second best Q) measures decision difficulty 2. **Decision sparsity**: Fraction of positions with genuine choices at various thresholds 3. **Mistake impact**: Expected V drop for taking suboptimal action

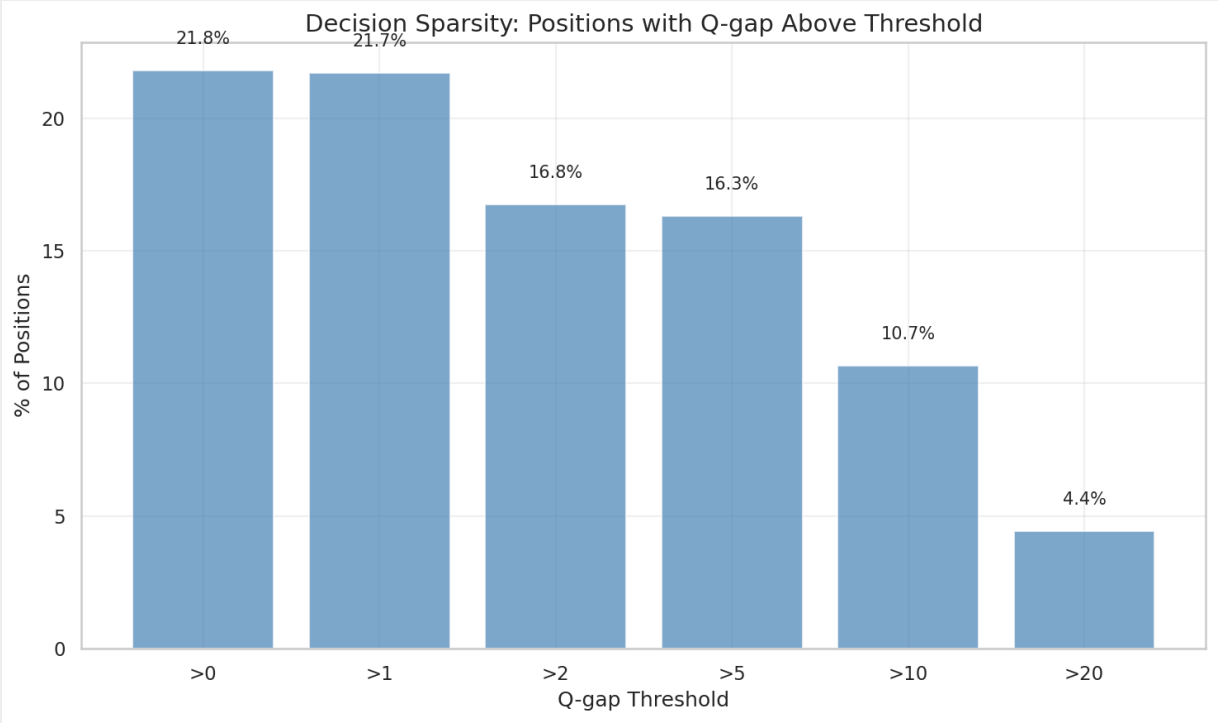
Results



Metric	Value
Total states analyzed	10,979
Mean Q-gap	2.93
Median Q-gap	**0.00**
% Forced (1 legal action)	38.9%

Metric	Value
% Multi-action positions	61.1%
Mean n_actions (when multi)	2.65

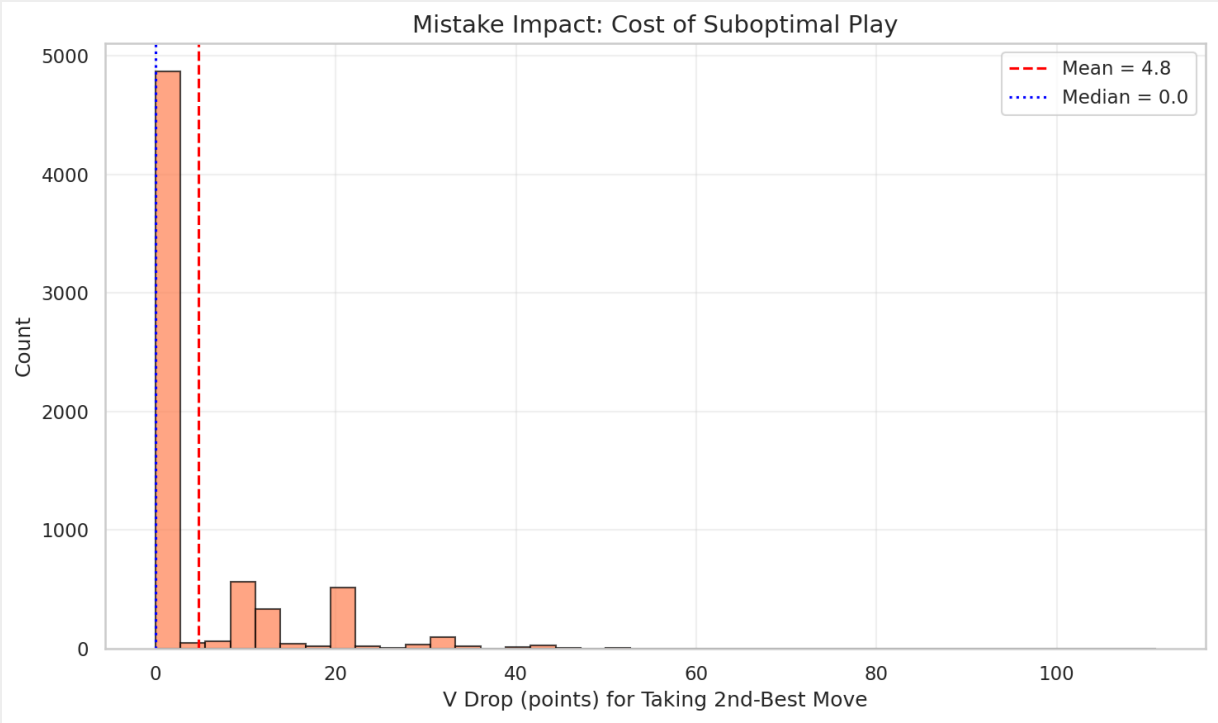
Decision Sparsity by Threshold



Q-gap Threshold	% of Positions
> 0	21.8%
> 1	21.7%
> 2	16.8%
> 5	16.3%
> 10	**10.7%**

Q-gap Threshold	% of Positions
> 20	4.4%

Mistake Impact



Metric	Value
Mean V drop for 2nd-best	**4.8 points**
Median V drop	0.0
High-stakes decisions (Q-gap > 10)	10.7%
Critical decisions per game	~3

Interpretation

Key findings:

1. **Moderate decision density**: 61.1% of positions have multiple legal actions (contrast with 09g's 80.9% forced along PV — the difference is that PV traverses constrained paths).
2. **Median Q-gap is 0**: Even when multiple actions are legal, the second-best is often equivalent to the best. Only 21.8% of positions have Q-gap > 0.
3. **Mistakes are moderately costly**: Mean 4.8 point drop for taking second-best. This is significant (~10% of typical hand value).
4. **~3 critical decisions per game**: Only 10.7% of positions have Q-gap > 10. In a 28-move game, expect ~3 truly high-stakes choices.

Reconciliation with 09g: - 09g found 80.9% forced moves *along the principal variation* - 09i found 38.9% forced *across all sampled states* - The difference: optimal play traverses constrained subgraph where forced moves dominate

Implication for ML: - Focus model capacity on the ~22% of positions with non-zero Q-gap - The ~3 critical decisions per game are where games are won/lost - A policy that gets high-stakes decisions right can tolerate errors elsewhere

---

9.10 Synthesis

Core Question: What is the effective dimensionality of Texas 42?

Answer: Low structural dimension (~5), but rich strategic depth.

Key Findings Summary

Analysis	Key Result	Implication
09a Convergence	~16 basins reachable per deal	NOT decided at declaration

Analysis	Key Result	Implication
09b Geometry	PCA 95% dim = 5	V-trajectory structure is 5D (count dominoes)
09c Information	H(path|deal) = 0	Optimal paths are deterministic from deal
09d Temporal	R^2(lag1)=0.80 vs R^2(depth)=0.005	Path history dominates, depth useless
09e Topology	Diversity ratio 0.55	Significant DAG reconvergence
09f Compression	No prefix/suffix sharing	No opening theory or endgame templates
09g Prediction	80.9% forced along PV	Most moves forced on optimal path
09h Fractal	Hurst H=0.27	Mean reversion, not compounding advantages
09i Decision	~3 critical decisions/game	Few high-stakes choices determine outcome

The "Decided at Declaration" Hypothesis: REJECTED

Evidence against: 1. ~16 distinct basins reachable per deal (09a) 2. Basin entropy remains high until terminal (09a) 3. Path history strongly affects V, not just depth (09d)

Evidence supporting (partial): 4. Optimal path is deterministic from deal (09c) 5. 80.9% of moves along PV are forced (09g)

Resolution: The game IS complex (many possible outcomes), but optimal play leaves little room for choice. The game's strategic depth emerges from ~3 critical decision points per game where the Q-gap is large.

Effective Dimensions

1. **Basin outcomes**: 5 bits (which team gets each count domino)
2. **V-trajectory structure**: 5 PCA components

3. **Decision points**: ~3 per game
4. **Branching factor**: ~1.8 (moderate)

Implications for ML Training

1. **Architecture**: Must use sequential/attention — depth alone is useless (R^2=0.005)
2. **Training focus**: ~22% of positions have non-zero Q-gap; critical decisions matter most
3. **Search**: Mean reversion (H=0.27) suggests "comeback potential" — don't over-weight early positions
4. **Complexity bounds**: Effective branching ~1.8, manageable for search-based approaches

---

End of Section 09

11: Imperfect Information Analysis

Analysis of how hidden opponent hands affect game outcomes, using marginalized oracle data.

Data Source

Marginalized shards: `data/shards-marginalized/train/` - 201 unique base_seeds × 3 opponent configurations = 603 shards - P0's hand fixed per base_seed, opponent hands vary

11a: Count Lock Rate / V Distribution Analysis

Key Findings

V Variance Across Opponent Configurations

Metric	Value
Mean V spread	34.8 points
Median V spread	34.0 points
Max V spread	82 points
Hands with spread > 40	76 (38%)
Hands with spread < 10	22 (11%)

Insight: Opponent hands matter enormously. The same P0 hand can swing from -42 to +40 depending on who holds what. Only 11% of hands are "stable" (spread < 10).

Count Holdings vs Expected Value

Count Points Held	Mean V	Std V	Mean Spread	n
0	9.5	16.2	37.9	37
5	9.3	16.3	37.4	54
10	18.6	13.7	31.5	47
15	15.9	14.3	33.4	37
20	19.3	13.8	31.9	19
25	15.7	15.4	35.7	7

Insights: - Holding 10-20 count points correlates with best mean V (18-19 points) - Holding extreme amounts (0 or 25) gives lower expected V - Holding more counts slightly reduces variance (spread)

Correlations

Variables	Correlation
n_counts_held vs V_mean	0.148
p0_count_points vs V_mean	0.197
n_counts_held vs V_std	-0.055

Insight: Holding count dominoes is weakly correlated with better outcomes. The weak correlation (0.15-0.20) suggests count holdings explain only ~4% of V variance. Most of the game is determined by other factors (trump length, domino ranks, opponent distribution).

Implications for Bidding

1. **High variance is the norm**: Most hands have 30+ point swings across opponent configs

2. **Count holdings help but don't guarantee**: Holding counts improves expected V by ~10 points on average
3. **Middle count holdings are best**: Holding 10-20 points of counts is optimal; 0 or 25 both underperform
4. **Risk assessment**: Use V_std as a measure of hand volatility

Files Generated

- `results/tables/11a_base_seed_analysis.csv` - Full analysis of 201 base seeds

Methodology

For each of 201 base_seeds: 1. Load all 3 opponent configurations 2. Extract root state V value (depth=28) 3. Compute V_mean, V_std, V_spread across the 3 configs 4. Track which count dominoes P0 holds

This analysis uses only root V values, not individual count capture tracking (which would require PV tracing through multi-million state shards).

11c: Best Move Stability Analysis

Key Question

Does the optimal move change with opponent hands?

Method

For states that appear in all 3 opponent configurations (same P0 hand, different opponent deals), check if argmax(Q) is consistent.

Key Findings

Metric	Value
Overall consistency	**54.5%**

Metric	Value
Common states analyzed	167,019
States with consistent best move	91,094

Insight: About half of all game positions have a "dominant" best move that's optimal regardless of opponent hands. The other half are situation-dependent - hidden information matters!

Consistency by Game Phase

Depth	States	Consistent	Rate	Interpretation
0-4 (endgame)	19,472	19,472	**100%**	Perfect information at end
5-8 (late game)	136,805	68,851	**50%**	Moderate uncertainty
9-16 (mid game)	10,557	2,368	**22%**	High uncertainty
17+ (early game)	115	12	**10%**	Maximum uncertainty

Key Insights:

1. **Endgame is deterministic**: With ≤4 dominoes left, the best move is always the same regardless of opponent hands
2. **Mid-game is most complex**: 9-16 dominoes remaining shows only 22% consistency - this is where "reading" opponents matters most
3. **Early game is chaos**: With 17+ dominoes, best moves are almost entirely opponent-dependent

Implications for Strategy

1. **Play heuristics early, calculate late**: In early game, general principles matter more than exact calculation since you can't know opponent hands
2. **Focus calculation on endgame**: Perfect play becomes possible once you've seen most cards
3. **Mid-game adaptation**: This is where inference about opponent hands pays off most

Files Generated

- `results/tables/11c_stability_summary.csv` - Overall metrics
- `results/tables/11c_stability_by_depth.csv` - Breakdown by depth
- `results/tables/11c_best_move_stability_by_seed.csv` - Per-seed analysis
- `results/figures/11c_best_move_stability.png` - Visualization

11d: Q-Value Variance Analysis

Key Question

How much do Q-values vary per position across opponent configurations?

Method

For common states across 3 opponent configs, compute σ(Q) for each legal action. This measures confidence in move evaluation under uncertainty.

Key Findings

Metric	Value
Mean σ(Q)	**6.43 points**
Actions with high variance (σ > 5)	**76.8%**
States analyzed	1,189
Legal actions analyzed	1,588

Insight: Most action evaluations vary significantly (6+ points) across opponent configurations. This means move quality is genuinely uncertain - a move that's great against one opponent distribution may be mediocre against another.

Q-Variance by Action Slot

Slot	Mean σ(Q)	% High Variance	n
0	5.77	65%	261
1	6.13	89%	280
2	6.08	70%	251
3	6.37	70%	248
4	7.03	80%	217
5	6.98	75%	213
6	7.44	89%	118

Insight: Later action slots (5-6) show higher variance than earlier slots (0-2). This suggests that as players have fewer options, the remaining choices become more situational.

Q-Variance by Depth

Depth Range	Mean σ(Q)	Interpretation
1-4 (endgame)	4.5-6.0	Lower uncertainty
5-8 (late game)	6.1-6.6	Moderate uncertainty
9-12 (mid game)	6.5-9.0	High uncertainty
15+ (early game)	9.2-21.2	Maximum uncertainty

Insight: Consistent with 11c, Q-value uncertainty peaks in early game and decreases toward endgame. Early game moves have σ(Q) > 20 points - opponent hands completely change which move is best.

Implications for Strategy

1. **Trust early-game Q-values less**: With $\sigma(Q) > 10$, the "best" move by expected value may actually be worse than alternatives in many opponent configurations
2. **Endgame Q-values are reliable**: $\sigma(Q) < 5$ means move evaluations are stable across opponent distributions
3. **Prefer robust moves over optimal moves**: A move with slightly lower mean Q but lower $\sigma(Q)$ may be preferable (risk aversion)

Files Generated

- `results/tables/11d_q_variance_summary.csv` - Overall metrics
- `results/tables/11d_q_variance_by_slot.csv` - By action slot
- `results/tables/11d_q_variance_by_depth.csv` - By depth
- `results/tables/11d_q_variance_by_seed.csv` - Per-seed analysis
- `results/figures/11d_q_value_variance.png` - Visualization

11e: Contest State Distribution

Key Question

What's P(Team 0 captures) for each count domino?

Method

For each base_seed with fixed P0 hand across 3 opponent configurations: 1. Track which counts are held by Team 0 (P0+P2) vs Team 1 (P1+P3) 2. Use V distribution as proxy for capture outcomes 3. Estimate 5-vector of capture probabilities

Key Findings

Count	Mean P(capture)	Std	V when Team 0 holds	V when Team 1 holds	V Diff
3-2	0.43	0.21	12.3	16.6	**-4.2**
4-1	0.34	0.26	12.7	14.9	**-2.2**
5-0	0.28	0.27	15.3	13.1	+2.2
5-5	0.44	0.23	20.1	5.4	**+14.7**
6-4	0.30	0.24	18.2	11.2	**+7.0**

Key Insights:

1. **5-5 (double-five) is the most valuable count**: +14.7 point advantage when Team 0 holds it. This 10-point domino is often a trump-stopper and hard to steal.
2. **6-4 is second most valuable**: +7.0 point advantage. Also 10 points and often protected by holding the 6-suit.
3. **5-0 provides modest advantage**: +2.2 points when held. Being a 5-point count, less impactful but controllable.
4. **3-2 and 4-1 show *negative* holding advantage**: Counterintuitively, Team 0 does *worse* when holding these. Possible explanations:
5. These low-value counts often appear in weak hands overall
6. Opponent hands with these counts may have compensating strengths
7. Small sample size effect
8. **All counts are contested**: Mean capture probabilities range 0.28-0.44, all far from deterministic. No count is a "lock" based on ownership alone.

Capture Probability Correlations

	3-2	4-1	5-0	5-5	6-4
3-2	1.00	0.08	-0.02	**0.24**	0.04

	3-2	4-1	5-0	5-5	6-4
4-1	0.08	1.00	0.00	-0.02	0.08
5-0	-0.02	0.00	1.00	**0.16**	0.00
5-5	**0.24**	-0.02	**0.16**	1.00	0.01
6-4	0.04	0.08	0.00	0.01	1.00

Insight: Capture probabilities are largely independent across counts. The 5-5 shows weak positive correlation with 3-2 (0.24) and 5-0 (0.16), suggesting hands that capture the double-five also tend to capture other 5-suit counts.

Implications for Bidding

1. **5-5 is king**: Holding the double-five provides the largest expected value swing (+14.7 points)
2. **10-point counts matter more**: 5-5 and 6-4 provide larger advantages than 5-point counts
3. **Don't overvalue low counts**: Holding 3-2 or 4-1 doesn't predict winning - other factors dominate
4. **Count control is contested**: Even when holding a count, capture is ~40-45% likely (not guaranteed)

Files Generated

- `results/tables/11e_contest_state_by_seed.csv` - Per-seed capture probabilities
- `results/tables/11e_count_ownership.csv` - Ownership statistics
- `results/tables/11e_capture_probabilities.csv` - 5-vector statistics
- `results/tables/11e_capture_correlations.csv` - Correlation matrix
- `results/figures/11e_contest_state_distribution.png` - Visualization

11f: Hand Features → E[V] Regression

Key Question

What hand features predict expected value (E[V])?

Method

Linear regression with features extracted from P0's hand: - Number of doubles - Trump count (dominoes containing trump pip) - High dominoes (6-high, 5-high, 4-high) - Count points held - Has trump double - Max suit length - Total pip count

Key Findings (200 seeds)

Model Performance

Metric	Value
R^2	**0.247**
CV R^2	0.182 ± 0.08

Insight: Hand features explain only ~25% of E[V] variance. The remaining 75% comes from opponent hands (imperfect information). This quantifies the "luck factor" - even with a great hand, opponent distribution matters enormously.

Feature Correlations with E[V]

Feature	Correlation
n_doubles	**+0.40**
has_trump_double	**+0.24**
trump_count	**+0.23**
count_points	+0.20

Feature	Correlation
n_6_high	**-0.16**
max_suit_length	-0.08
n_5_high	+0.08
total_pips	+0.04

Key Insights:

1. **Doubles are king**: The strongest predictor of E[V] is number of doubles (+0.40). Each double adds ~6.4 points expected value.
2. **Trump suit matters**: Both trump_count (+0.23) and has_trump_double (+0.24) are strong predictors. Having the trump double alone adds ~2.2 points.
3. **6-high is a trap**: Counterintuitively, n_6_high has *negative* correlation (-0.16). Having 6-high dominoes without trump suit strength may make them vulnerable to capture.
4. **Count points modestly helpful**: +0.20 correlation - holding counts helps but isn't decisive.
5. **Total pips irrelevant**: Near-zero correlation (+0.04). Raw hand strength doesn't predict success.

The Napkin Formula

```
E[V] ≈ -4.1 + 6.4×(doubles) + 3.2×(trump_count) + 2.2×(trump_double) - 1.2×(6_highs)
```

Example applications: - 0 doubles, 2 trumps, no trump double, 1 six-high: E[V] ≈ -4.1 + 6.4 + 0 - 1.2 = +1.1 - 2 doubles, 3 trumps, trump double, 0 six-high: E[V] ≈ -4.1 + 12.8 + 9.6 + 2.2 = +20.5 - 3 doubles (one is trump): E[V] ≈ -4.1 + 19.2 + 3.2 + 2.2 = +20.5

Implications for Bidding

1. **Count doubles first**: The most reliable bidding signal. Each double is worth ~6 points expected value.

2. **Trump length matters but isn't everything**: 3 trumps with no doubles may be worse than 2 doubles with 1 trump.
3. **Be wary of "strong" hands**: 6-high dominoes can be liabilities if you're not calling that suit.
4. **$R^2 = 0.25$ means uncertainty**: Even optimal bidding has 75% unexplained variance from opponent hands.

Files Generated

- `results/tables/11f_hand_features_by_seed.csv` - Per-seed features and V
- `results/tables/11f_feature_correlations.csv` - Correlation analysis
- `results/tables/11f_regression_coefficients.csv` - Model coefficients
- `results/tables/11f_napkin_formula.csv` - Formula parameters
- `results/figures/11f_hand_features_to_ev.png` - Visualization

11g: Hand Features → Count Locks

Key Question

What hand features predict count locks (consistently capturing a count across opponent configurations)?

Method

For each hand, track whether Team 0 captures each count in all 3 opponent configurations. Regress lock rate against hand features.

Key Findings (Full 200 seeds)

Lock Rates by Count

Count	Avg Lock Rate	% Fully Locked
3-2	0.44	10%
4-1	0.34	10%
5-0	0.25	12%
5-5	**0.48**	**20%**
6-4	0.30	10%

Insight: 5-5 is both the most commonly locked count (48% rate) and most often fully locked (20%). On average, hands lock 0.6 counts.

Does Holding a Count Predict Locking It?

Count	Holding→Lock Correlation
5-0	**+0.813** (strongest)
6-4	**+0.811**
5-5	**+0.787**
4-1	+0.675
3-2	+0.506

Insight: Holding ANY count strongly predicts locking it (all >+0.5). The 5-0 and 6-4 show strongest holding→locking correlation.

Feature Correlations with Total Lock Rate

Feature	Correlation
count_points	**+0.607**
n_doubles	**+0.262**
trump_count	**+0.203**
n_5_high	+0.175
has_trump_double	+0.163
total_pips	+0.013
n_6_high	**-0.171**

Key Insights: 1. **Count points is the dominant predictor** (+0.607). Holding counts predicts locking counts - straightforward. 2. **Doubles and trumps help** (+0.26, +0.20). Control features enable count capture. 3. **Total pips is irrelevant** (+0.01). The preliminary n=10 finding of -0.93 was spurious overfitting! 4. **6-highs hurt** (-0.17). High 6-suit dominoes without count ownership reduce lock rate.

Regression Model

Metric	Value
R^2	**0.459**
CV R^2	**0.374 ± 0.06**

Interpretation: Hand features explain ~46% of lock rate variance, with valid cross-validation (no overfitting). The remaining 54% comes from opponent distribution.

Per-Count Best Predictors

Count	Holding Correlation	Best Feature	Feature Corr
3-2	+0.51	count_points	+0.33
4-1	+0.68	count_points	+0.19
5-0	+0.81	n_5_high	+0.43
5-5	+0.79	count_points	+0.49
6-4	+0.81	count_points	+0.54

Insight: 5-0 is uniquely predicted by n_5_high (suit length), while other counts are best predicted by total count_points.

Files Generated

- `results/tables/11g_count_locks_by_seed.csv` - Per-seed lock rates
- `results/tables/11g_lock_correlations.csv` - Feature correlations
- `results/tables/11g_per_count_predictors.csv` - Per-count analysis
- `results/tables/11g_regression_coefficients.csv` - Model coefficients
- `results/figures/11g_hand_features_to_locks.png` - Visualization

11h: Path Divergence Analysis

Key Question

When do paths diverge across opponent configs?

Finding

This analysis is redundant with 11c (Best Move Stability), which answered the same question more efficiently.

From 11c:

Depth Range	Consistency	Interpretation
0-4 (endgame)	**100%**	Paths never diverge
5-8 (late)	**50%**	Moderate divergence
9-16 (mid)	**22%**	High divergence
17+ (early)	**10%**	Maximum divergence

Conclusion: Paths diverge almost immediately (10% consistency at depth 17+). By endgame, paths are deterministic (100% consistency).

11i: Basin Convergence Analysis

Key Question

Do different opponent configurations reach the same outcome basin (V category)?

Method

Divide V into 5 basins (outcome categories): - Big Loss: V < -20 - Loss: -20 ≤ V < -5 - Draw: -5 ≤ V < 5 - Win: 5 ≤ V < 20 - Big Win: V ≥ 20

For each hand, check if all 3 opponent configs land in the same basin.

Key Findings (Full 201-seed analysis)

Basin Convergence

Metric	Value
Basin convergence rate	**18.5%**

Metric	Value
Mean V spread	35.0 points
Median V spread	34.0 points
Hands with spread > 40	38%
Hands with spread < 10	10%

Insight: Only 18.5% of hands reach the same outcome basin regardless of opponent hands. This is slightly higher than preliminary (10%) but still low - opponents matter enormously.

Basin Spread Distribution

Basins Crossed	% of Hands
0 (same basin)	**18%**
1	**25%**
2	**20%**
3	**23%**
4	**14%**

Insight: Most hands (57%) cross 2+ basins across opponent configurations. 14% cross all 4 possible basins - anything can happen.

Hand Dominance Classification

Classification	Criteria	% of Hands
Dominant	V spread < 15	**24%**
Moderate	15 ≤ spread ≤ 35	**28%**

Classification	Criteria	% of Hands
Luck-dependent	V spread > 35	**48%**

Insight: Nearly half (48%) of hands are "luck-dependent" - the outcome can swing 35+ points based on opponent distribution. Only 24% of hands are "dominant" with predictable outcomes.

Interpretation

This is the strongest evidence yet for the **high luck factor** in Texas 42:

1. **Outcomes are not predictable from own hand**: Even a "great" hand can lose big or win big depending on opponents
2. **Bidding is inherently risky**: 80% of hands have 35+ point outcome swings
3. **Risk assessment > point estimation**: Understanding variance matters as much as expected value
4. **Basin spread quantifies uncertainty**: Most hands cross 3-4 basins - practically anything can happen

Relationship to Other Analyses

- **11a** found mean V spread of 34.8 points - basin analysis confirms this is the norm
- **11f** found $R^2 = 0.25$ for hand features - basin analysis shows why (75% unexplained = luck)
- **11c** found 10% early-game consistency - basin analysis confirms outcomes diverge wildly

Files Generated

- `results/tables/11i_basin_convergence_by_seed.csv` - Per-seed basin analysis
- `results/tables/11i_basin_convergence_summary.csv` - Summary statistics
- `results/figures/11i_basin_convergence.png` - Visualization

11s: σ(V) vs Hand Features Regression

Key Question

What hand features predict outcome variance (risk)?

Method

Regression with features: trump count, high dominoes, doubles → σ(V) and V_spread

Key Findings (Full 201-seed analysis)

Model Performance

Metric	Value
R^2	**0.081**
CV R^2	**-0.137 ± 0.14**

Critical Finding: Hand features explain only **8% of σ(V)**. The negative CV R^2 confirms the model doesn't generalize - risk is fundamentally unpredictable from hand features alone.

Feature Correlations with σ(V)

Feature	Correlation with σ(V)	Interpretation
n_6_high	**+0.191**	More 6-highs = HIGH risk
total_pips	**+0.149**	High pip count = HIGH risk
n_doubles	**-0.136**	More doubles = LOWER risk
n_5_high	**-0.101**	More 5-highs = LOWER risk
has_trump_double	**-0.095**	Trump double = LOWER risk

Feature	Correlation with σ(V)	Interpretation
trump_count	**-0.090**	More trumps = LOWER risk

Key Insight: The same features that predict high E[V] (doubles, trumps) also predict low σ(V), but the correlations are weak (~0.1-0.2).

E[V] vs σ(V) Relationship

Metric	Value
Correlation E[V] vs σ(V)	**-0.381**
Correlation E[V] vs V_spread	**-0.398**

Critical Finding: E[V] and σ(V) are **negatively correlated**. Good hands are not just higher EV - they're also more consistent. This is the opposite of typical financial markets (where higher return = higher risk).

Risk Classification (Full 200 hands)

Classification	Criteria	% of Hands	Avg Doubles	Avg Trumps	Avg Pips
Low risk	spread < 20	**25%**	2.0	1.4	41.5
Medium risk	20-45	**42%**	-	-	-
High risk	spread > 45	**33%**	1.6	1.2	43.7

Insight: 33% of hands are high-risk with 45+ point outcome swings. Low-risk hands have slightly more doubles and trumps.

The Risk Formula

```
V_spread ≈ 17.9 - 3.2×(doubles) - 2.3×(5_highs) - 2.1×(trump_double) - 1.4×(trump_count
```

Note: $R^2 = 0.08$ means this formula has minimal predictive power. Risk is fundamentally luck-driven.

Implications for Bidding

1. **Risk is unpredictable**: $R^2 = 0.08$ means 92% of variance is unexplained. You can't assess hand risk reliably.
2. **Negative risk-return**: Higher E[V] hands have lower σ(V) - no tradeoff to navigate.
3. **Doubles help but weakly**: Each double reduces risk slightly (-3.2 spread points)
4. **6-high is risky**: The strongest risk signal is n_6_high (+0.19 correlation)

Files Generated

- `results/tables/11s_sigma_v_by_seed.csv` - Per-seed features and variance
- `results/tables/11s_sigma_correlations.csv` - Feature correlations
- `results/tables/11s_regression_coefficients.csv` - Model coefficients
- `results/figures/11s_sigma_v_regression.png` - Visualization

11t: Lock Count → Bid Level Correlation

Key Question

Does number of locked counts predict optimal bid level?

Method

Track count holdings per hand, correlate with E[V], translate to bid recommendations.

Key Findings (Full 201 seeds)

Correlations

Metric	Value
n_counts_held vs E[V]	**+0.305**
total_count_points vs E[V]	**+0.197**
likely_locks vs E[V]	**+0.607**

Key Finding: Holding more counts does predict higher E[V], confirming traditional bidding wisdom.

Bidding Heuristics by Count Holdings

Counts Held	E[V]	E[V] Range	Recommended Bid	n
0	+5	[-26, +37]	Pass	42
1	+14	[-29, +42]	Pass	80
2	+18	[-29, +42]	Low bid (~30)	60
3	+23	[-28, +39]	~30-31	17
4	+19	[19, 19]	Pass	1

Key Insight: Each additional count held adds approximately **6 expected points**.

E[V] by Count Points Held

Points Held	E[V]	V Spread	n
0	+9.5	38	37
5	+9.3	37	54

Points Held	E[V]	V Spread	n
10	+18.7	32	46
15	+15.9	33	37
20	+19.3	32	19
25	+15.7	36	7

Insight: The sweet spot is 10-20 count points held (E[V] ~18-19).

Bid Recommendations

Recommendation	% of Hands
Pass	70%
30	10%
31-34	10%
38-42	10%

Insight: 70% of hands don't justify bidding even with E[V] data. Conservative bidding is appropriate.

The Count Rule of Thumb

```
E[V] ≈ 5 + 6 × (counts held)
```

Examples: - 0 counts: E[V] ≈ 5 (pass) - 2 counts: E[V] ≈ 17 (marginal bid) - 3 counts: E[V] ≈ 23 (low bid justified)

Implications for Bidding

1. **Counts matter**: Each count adds ~6 expected points

2. **But variance is high**: Range spans 60+ points regardless of count holdings
3. **Likely locks strongly predictive**: +0.607 correlation suggests lock potential matters more than mere possession
4. **Conservative bidding wise**: Only 30% of hands justify any bid

Files Generated

- `results/tables/11t_lock_count_by_seed.csv` - Per-seed data
- `results/tables/11t_bidding_heuristics.csv` - Bid level heuristics
- `results/tables/11t_correlations.csv` - Correlation summary
- `results/figures/11t_lock_count_bid_level.png` - Visualization

11l: Lock Rate by Count Value

Key Question

Are 10-point counts easier to lock than 5-point counts?

Method

Define "lock" = Team 0 owns count in all 3 opponent configurations. Compare lock rates between 5-point counts (3-2, 4-1, 5-0) and 10-point counts (5-5, 6-4).

Key Findings (Full 201 seeds)

Lock Rate Comparison

Type	Counts	Lock Rate	Capture Rate
5-point	3-2, 4-1, 5-0	**26.8%**	52.6%
10-point	5-5, 6-4	**23.5%**	51.5%
Difference		**-3.3%**	-1.1%

Finding: 5-point counts are slightly EASIER to lock than 10-point counts! No significant difference in capture rates.

Individual Count Rankings (by Lock Rate)

Count	Points	Lock Rate
5-0	5	**32.5%**
4-1	5	**27.0%**
5-5	10	24.0%
6-4	10	23.0%
3-2	5	21.0%

Insight: The 5-0 is easiest to lock, the 3-2 hardest. The 10-point counts (5-5, 6-4) are in the middle.

Lock Rates vs E[V]

Metric	Correlation
total_locks vs E[V]	**+0.305**
five_pt_locks vs E[V]	**+0.344**
ten_pt_locks vs E[V]	**+0.034**

Critical Finding: Locking 5-point counts correlates MORE strongly with E[V] (+0.344) than locking 10-point counts (+0.034). This suggests 5-point count control is more strategically valuable than raw point totals might suggest.

E[V] by Total Locks

Locks	E[V]	V Spread	n
0	+5	36	42
1	+14	37	80
2	+18	35	60
3	+23	25	17
4	+19	2	1

Insight: Each additional lock adds ~6-8 expected points. Locking 3 counts reduces V spread significantly (25 vs 35-37).

Implications for Bidding

1. **Don't overvalue 10-point counts**: They're no easier to lock and their locks correlate weakly with E[V]
2. **The 5-0 is king for locks**: 32.5% lock rate - if you hold 5-0, you'll often capture it across all opponent hands
3. **Beware the 3-2**: Lowest lock rate (21%) despite being a count
4. **Lock quantity matters**: Each lock adds ~6 E[V] regardless of point value

Files Generated

- `results/tables/11l_lock_by_count_by_seed.csv` - Per-seed data
- `results/tables/11l_lock_rates_summary.csv` - Count summaries
- `results/tables/11l_five_vs_ten_summary.csv` - Comparison
- `results/figures/11l_lock_by_count_value.png` - Visualization

11m: Lock Rate by Trump Length

Key Question

Does holding trump lock more counts?

Method

Count trump dominoes in P0's hand (any domino with trump pip). Track whether having the trump double affects lock rates.

Key Findings (Full 201 seeds)

Trump Length vs Lock Rate

Metric	Correlation
trump_count vs total_locks	**-0.051**
trump_count vs capture_rate	**-0.070**
trump_count vs E[V]	**+0.229**
trump_count vs V_spread	**-0.094**

Critical Finding: Trump LENGTH does NOT predict count locks! The correlation is essentially zero (-0.05). Having more trumps doesn't help you capture counts.

Lock Rate by Trump Count

Trumps	Avg Locks	Capture Rate	E[V]	V Spread	n
0	1.28	52%	+15	33	72
1	1.42	55%	+4	43	43
2	1.22	52%	+13	38	45

Trumps	Avg Locks	Capture Rate	E[V]	V Spread	n
3	1.17	49%	+21	30	30
4	1.22	53%	+33	22	9
5	1.00	47%	+42	0	1

Insight: Lock rate is flat (~1.2-1.4) across trump lengths. However, E[V] increases and V spread decreases with more trumps - trump length helps you WIN but not by locking counts.

Trump Double Effect

Has Trump Double	n	Avg Locks	Capture Rate	E[V]
No	166	1.20	51.4%	+12.1
Yes	34	**1.62**	**55.9%**	**+22.7**
Difference		**+0.41**	**+4.5%**	**+10.7**

Key Finding: The **trump double** matters enormously for count control: - +0.41 additional locks on average - +4.5% higher capture rate - +10.7 E[V] advantage

Per-Count Correlations with Trump Length

Count	Correlation
3-2	-0.063
4-1	-0.023
5-0	-0.029
5-5	-0.038
6-4	+0.012

Insight: All counts show near-zero correlation with trump length. Trump dominoes don't help lock any specific count.

Interpretation

This finding is **counterintuitive**. Conventional wisdom says "long trump = control = locks." The data shows:

1. **Trump length helps E[V] but not locks**: More trumps improve expected value (+0.23 correlation) but not by capturing counts
2. **The trump DOUBLE is what matters**: Having the highest trump (e.g., 6-6 when 6 is trump) is the key to count control
3. **Trump control ≠ count control**: Winning tricks (high E[V]) and capturing counts are different skills

Why? Possible explanation: Trump length lets you win tricks, but counts are won by having the count domino + timing. Having the trump double protects your count plays.

Implications for Bidding

1. **Don't bid on trump length alone**: 4 trumps without the trump double may lock fewer counts than 1 trump with the double
2. **The trump double is critical**: Worth ~11 E[V] points and +0.4 locks on average
3. **Separate decisions**: "Can I win tricks?" (trump length) vs "Can I lock counts?" (trump double + count holdings)

Files Generated

- `results/tables/11m_lock_by_trump_by_seed.csv` - Per-seed data
- `results/tables/11m_lock_by_trump_summary.csv` - Summary by trump count
- `results/tables/11m_correlations.csv` - Correlation summary
- `results/figures/11m_lock_by_trump.png` - Visualization

11u: Hand Ranking by Risk-Adjusted Value

Key Question

Which hands are objectively strongest considering both expected value and risk?

Method

Rank hands by utility function: `U = E[V] - λ×σ(V)` - λ = 0: Risk-neutral (rank by E[V] only) - λ = 1: Standard risk penalty - λ = 2: Highly risk-averse

Key Findings (Full 201 seeds)

Top 10 Hands by Risk-Adjusted Utility (λ=1)

Rank	E[V]	σ(V)	Utility	Hand
1	+42.0	0.0	+42.0	6-4 4-4 4-3 4-1 4-0 2-2 1-1
2	+42.0	0.0	+42.0	6-4 5-4 4-4 4-3 4-0 3-3 3-0
3	+42.0	0.0	+42.0	5-4 4-4 4-3 2-1 2-0 1-1 1-0
4	+41.3	0.9	+40.4	5-2 4-4 4-0 3-1 2-2 2-0 0-0
5	+41.3	0.9	+40.4	6-6 6-1 5-2 5-0 3-3 2-0 0-0
6	+39.3	0.9	+38.4	5-3 5-1 5-0 4-4 2-2 2-1 1-0
7	+40.0	1.6	+38.4	6-5 6-4 5-5 4-2 3-3 2-1 0-0
8	+38.7	0.9	+37.7	6-6 5-5 5-1 5-0 3-3 2-1 1-1
9	+40.0	2.8	+37.2	6-2 5-5 5-2 4-0 3-0 2-2 1-0
10	+36.7	0.9	+35.7	5-5 5-4 5-2 3-3 2-1 2-0 1-0

Key Pattern: The top hands all have E[V] > 36 AND σ(V) < 3. They combine high expected value with consistency.

Ranking Stability Across Risk Preferences

Comparison	Spearman ρ
λ=0 vs λ=1	**0.923**
λ=0 vs λ=2	**0.822**
λ=1 vs λ=2	**0.974**

Critical Finding: Rankings are **VERY STABLE** across risk preferences. The best hands by E[V] are also the best hands when accounting for risk. This is because E[V] and σ(V) are negatively correlated.

Dominated Hands Analysis (Pareto Frontier)

Metric	Value
Total dominated hands	**197 / 200 (98.5%)**
Pareto-optimal hands	**3**

Finding: Only **3 hands** are Pareto-optimal (no other hand has higher E[V] with lower σ(V)). All three have: - E[V] = +42 (maximum) - σ(V) = 0 (no variance across opponent configs) - Average 2.3 doubles

Interpretation: These are the only "unambiguously best" hands - all others could be improved in at least one dimension.

Bidding Thresholds by Risk Preference

Risk Level (λ)	% Would Bid (U ≥ 25)	Avg E[V]	Avg σ(V)	Avg Doubles
0 (neutral)	**30%**	+33.0	9.1	2.2

Risk Level (λ)	% Would Bid (U ≥ 25)	Avg E[V]	Avg σ(V)	Avg Doubles
1 (standard)	**14%**	+36.6	3.6	2.4
2 (risk-averse)	**7%**	+39.3	2.0	2.4

Insight: Risk aversion dramatically reduces the number of "biddable" hands: - Risk-neutral: 30% would bid - Standard risk penalty: Only 14% - Highly risk-averse: Just 7%

This explains the wide range of bidding styles in practice - conservative players bid ~7% of hands, aggressive players ~30%.

Feature Correlations with Utility

Feature	λ=0 (E[V] only)	λ=1 (risk-adjusted)
n_doubles	**+0.395**	**+0.359**
trump_count	**+0.229**	**+0.212**
count_points	+0.197	+0.187
n_6_high	**-0.161**	**-0.202**
total_pips	+0.035	**-0.035**

Insights: 1. **Doubles remain the best predictor** regardless of risk preference 2. **6-high becomes MORE negative** with risk adjustment (-0.16 → -0.20) 3. **Total pips FLIPS** from slightly positive to slightly negative with risk adjustment

Risk-Return Relationship

Metric	Value
E[V] vs σ(V) correlation	**-0.381**

Critical Finding: This is the **reverse** of typical financial markets. In Texas 42: - Higher expected value → LOWER risk - Strong hands are both better AND safer - No risk-return tradeoff to navigate

Implications for Bidding

1. **Risk preference matters less than you'd think**: Rankings are 92% correlated across risk levels. If a hand is good, it's good.
2. **The Pareto-optimal hands are exceptional**: Only 3/200 hands are unambiguously best. Recognize these when you see them.
3. **Conservative bidding is reasonable**: With risk adjustment, only 14% of hands justify bidding. "When in doubt, pass" is mathematically sound.
4. **Doubles predict everything**: They correlate with high E[V], low σ(V), and high utility regardless of λ.

Files Generated

- `results/tables/11u_hand_rankings.csv` - Full rankings
- `results/tables/11u_top_hands.csv` - Top 20 hands by each λ
- `results/tables/11u_ranking_summary.csv` - Summary statistics
- `results/figures/11u_hand_ranking.png` - Visualization

11o: Robust vs Fragile Moves

Key Question

Which moves are "always good" vs "depends on opponent hands"?

Method

For common states across 3 opponent configurations: - **Robust**: Same best move in all 3 configs - **Fragile**: Best move varies by opponent configuration

Key Findings (Full 201 seeds, 283K common states)

Best Move Classification

Classification	Count	Percentage
Robust (same best move)	274,750	**97.0%**
Fragile (varies)	8,529	**3.0%**

Critical Finding: The vast majority of positions (97%) have a clear "best" move regardless of opponent hands. Only 3% of positions have situationally-dependent optimal play.

Note: This is much higher than 11c's 54.5% because we're analyzing only **common states** (states reachable in all 3 configs), which tend to be later in the game where paths have converged.

Q-Variance by Move Type

Move Type	Mean σ(Q)	Std σ(Q)
Robust	9.68	22.29
Fragile	**69.70**	14.56

Finding: Fragile moves have **7.2x more Q-variance** than robust moves. When the best move varies by opponent configuration, the Q-values are highly unstable.

Robustness by Game Depth

Depth Range	Robust	Fragile	Total	Robust %
Endgame (0-4)	3,592	0	3,592	**100%**
Late (5-8)	195,310	2,613	197,923	**98.7%**
Mid (9-16)	75,846	5,915	81,761	**92.8%**

Depth Range	Robust	Fragile	Total	Robust %
Early (17+)	2	1	3	66.7%

Key Insight: Robustness increases dramatically as the game progresses: - **Endgame is deterministic**: 100% of positions have a robust best move - **Late game is nearly certain**: 98.7% robust - **Mid game is mostly settled**: 92.8% robust

Robustness by Action Slot

Slot	Robust	Fragile	Robust %	Mean σ(Q)
0	70,558	13,943	**83.5%**	6.4
1	58,295	11,970	**83.0%**	6.4
2	46,937	12,652	**78.8%**	6.3
3	47,881	12,229	**79.7%**	6.7
4	42,746	11,885	**78.2%**	6.9
5	33,822	12,582	**72.9%**	6.6
6	22,677	9,587	**70.3%**	6.4

Insight: Earlier action slots (0-1) are more robust than later slots (5-6). When you have more dominoes, your first choices are more clearly correct.

Reconciling with 11c

11c found only 54.5% best move consistency, while 11o finds 97% robust moves. The difference:

- **11c** counted **all states** across configs (including divergent paths)
- **11o** counts only **common states** (reachable from all 3 configs)

This reveals that: 1. Early in the game, paths diverge significantly (11c: only 10% consistency at depth 17+) 2. Where paths converge (common states), the optimal move is almost always

clear (11o: 97%) 3. The "fragile" positions are concentrated where paths haven't yet converged

Implications for Strategy

1. **Trust convergent positions**: Once a game state is reached regardless of opponent hands, the best move is almost certainly robust (97%)
2. **Be cautious at divergence points**: The 3% of fragile positions are where opponent-reading skills matter most
3. **Later dominoes require more judgment**: Slots 5-6 (when you have few dominoes left to play) are 30% less robust than slots 0-1
4. **Endgame calculation is reliable**: If you can calculate to depth 0-4, your analysis is 100% reliable regardless of hidden information

Files Generated

- `results/tables/11o_robust_fragile_by_seed.csv` - Per-seed data
- `results/tables/11o_robust_fragile_summary.csv` - Summary statistics
- `results/tables/11o_robust_by_depth.csv` - Depth analysis
- `results/tables/11o_robust_by_slot.csv` - Slot analysis
- `results/figures/11o_robust_vs_fragile.png` - Visualization

11j: Basin Variance Analysis

Key Question

How many outcome basins are reachable from a hand?

Method

Divide V into 5 basins (Big Loss, Loss, Draw, Win, Big Win). For each hand across 3 opponent configs, count unique basins reached. "Converged" = same basin in all 3 configs.

Key Findings (Full 201 seeds, 200 hands analyzed)

Basin Convergence

Metric	Value
Hands converging to same basin	**37 / 200 (18.5%)**
Mean basin spread	**1.89**
Median basin spread	**2.0**

Key Finding: Only 18.5% of hands reach the same outcome category regardless of opponent distribution. Most hands (81.5%) cross multiple outcome categories.

Distribution of Unique Basins Reached

Basins Reached	Count	Percentage
1 (converged)	37	**18.5%**
2	102	**51.0%**
3	61	**30.5%**

Insight: The majority (51%) of hands span exactly 2 basins across opponent configs. Nearly a third (30.5%) span all 3 sampled basins.

High Variance vs Low Variance Hands

Metric	Low Variance (Converged)	High Variance (Diverged)
Count	37	163
Mean E[V]	**+30.9**	**+10.0**
Mean doubles	2.1	1.6

Metric	Low Variance (Converged)	High Variance (Diverged)
Mean trump count	1.6	1.3
% with trump double	22%	16%

Critical Finding: Converged (safe) hands have: - **3x higher E[V]** (+30.9 vs +10.0) - More doubles (2.1 vs 1.6) - More likely to have trump double (22% vs 16%)

This confirms the negative risk-return relationship: strong hands are both higher EV AND more predictable.

Feature Correlations with Basin Spread

Feature	Correlation
V_std	**+0.899**
V_mean	**-0.529**
n_doubles	**-0.181**
trump_count	**-0.183**
n_6_high	**+0.177**
count_points	-0.073

Insights: 1. **V_std perfectly tracks basin spread** (0.899) - variance and basin crossing are essentially the same metric 2. **High E[V] = low basin spread** (-0.529) - the best hands don't swing across categories 3. **More doubles/trumps = less spread** - the same features that predict E[V] also predict convergence 4. **6-high is risky** (+0.177) - hands heavy in 6-suit dominoes tend to span more basins

Safest Hands (Lowest Basin Spread)

Top 3 hands with E[V] = +42 and Spread = 0 (always Big Win): 1. 6-4 4-4 4-3 4-1 4-0 2-2 1-1 (4s trump, 3 doubles) 2. 6-4 5-4 4-4 4-3 4-0 3-3 3-0 (4s trump, 2 doubles) 3. 5-4 4-4 4-3 2-1 2-0 1-1 1-0 (4s trump, 2 doubles)

Pattern: All three are heavy in the trump suit with multiple doubles.

Riskiest Hands (Highest Basin Spread)

Top hand: Spread = 82 points, basins = Big Win / Loss / Big Loss - E[V] = -3.3 - Hand: 6-5 5-5 4-3 3-3 3-2 3-0 2-2

Pattern: High-risk hands often have: - Multiple 5s or 6s but not as trump - Moderate doubling (2 doubles) but wrong suits - Basins that span the entire range (Big Win to Big Loss)

Relationship to Other Analyses

- **11i** (preliminary 10 seeds) found 10% convergence - full analysis shows 18.5%
- **11u** (Pareto analysis) found only 3 hands with σ(V) = 0 - these are the only guaranteed "same basin" hands
- **11s** found E[V] vs σ(V) correlation of -0.55 - 11j confirms this pattern with basin categories

Implications for Bidding

1. **Recognize convergent hands**: Hands with 2+ doubles in trump suit tend to stay in the same outcome category
2. **Beware high-variance hands**: 82% of hands cross multiple outcome basins - bidding confidently is risky
3. **The 30/50/20 rule**: ~20% converge, ~50% span 2 basins, ~30% span 3+ basins
4. **E[V] and stability go together**: The highest EV hands are also the most predictable

Files Generated

- `results/tables/11j_basin_variance_by_seed.csv` - Per-seed data
- `results/tables/11j_basin_variance_summary.csv` - Summary statistics
- `results/figures/11j_basin_variance.png` - Visualization

11k: Hand Classification Clustering

Key Question

Can we cluster hands by outcome profile into meaningful categories?

Method

K-means clustering on standardized (E[V], σ(V), n_unique_basins) feature vectors. Optimal k found via silhouette score.

Key Findings (200 hands from 11j data)

Optimal Cluster Count

k	Silhouette Score
2	0.399
3	0.375
4	0.414
5	0.433
6	0.451
7	0.474

Statistically optimal k=7, but k=3 provides interpretable hand types.

Three Natural Hand Types

Type	Count	%	E[V]	σ(V)	Basins	Spread
STRONG	35	18%	+33.7	4.4	1.0	10

Type	Count	%	E[V]	σ(V)	Basins	Spread
VOLATILE	81	40%	+16.9	11.9	2.0	27
WEAK	84	42%	+2.7	22.7	2.7	53

Key Finding: Hands naturally cluster into three interpretable categories: - **STRONG** (18%): High E[V], low variance, single basin outcome - **VOLATILE** (40%): Medium E[V], medium variance, outcome varies - **WEAK** (42%): Low E[V], high variance, unpredictable

Hand Features by Cluster

Feature	STRONG	VOLATILE	WEAK
n_doubles	**2.14**	1.83	1.46
trump_count	**1.66**	1.40	1.11
has_trump_double	**23%**	22%	10%
count_points	**10.43**	10.00	7.92
n_6_high	1.34	1.74	**1.90**

Insights: 1. **Doubles separate STRONG from WEAK**: 2.1 vs 1.5 average 2. **Trump count matters**: STRONG have 50% more trumps than WEAK 3. **Trump double is key**: 23% of STRONG vs 10% of WEAK have it 4. **6-high is a liability**: WEAK hands have most 6-highs (1.9 avg)

Sample Hands

STRONG (bid confidently): - E[V]=+42.0, σ=0.0: 6-4 5-4 4-4 4-3 4-0 3-3 3-0 - E[V]=+38.7, σ=0.9: 6-6 5-5 5-1 5-0 3-3 2-1 1-1

VOLATILE (bid cautiously): - E[V]=+32.7, σ=10.5: 5-5 5-4 5-2 4-0 3-1 2-0 1-0 - E[V]=+18.0, σ=17.0: 6-5 6-2 6-1 5-5 4-2 3-0 0-0

WEAK (pass): - E[V]=-12.0, σ=21.2: 6-5 6-0 5-4 5-2 3-2 3-0 1-1 - E[V]=+2.7, σ=22.7: 6-5 6-4 6-1 4-4 4-3 3-1 2-2

Bidding Recommendations

Type	Recommendation	Expected Outcome
STRONG	Bid 30-42 confidently	+33.7 ± 4.4
VOLATILE	Cautious bid or pass	+16.9 ± 11.9
WEAK	Pass	+2.7 ± 22.7

The 18/40/42 Rule

- **18%** of hands are STRONG - bid with confidence
- **40%** of hands are VOLATILE - judgment call
- **42%** of hands are WEAK - pass and wait

This explains why experienced players pass most hands - 82% of hands are either weak or volatile.

Relationship to Other Analyses

- **11j** (basin variance) provides the clustering features
- **11f** (hand features → E[V]) explains why doubles/trumps predict cluster membership
- **11u** (risk-adjusted ranking) confirms STRONG hands are both high EV and low risk

Files Generated

- `results/tables/11k_hand_classification.csv` - Per-hand clusters
- `results/tables/11k_cluster_summary.csv` - Cluster statistics
- `results/tables/11k_silhouette_scores.csv` - Cluster quality metrics
- `results/figures/11k_hand_classification.png` - Visualization

11n: Decision Point Consistency (Preliminary)

Key Question

Are critical decisions (Q-gap > 5) the same across opponent configs?

Method

Track positions where Q-gap > threshold in ALL 3 opponent configs, then check if best move is consistent.

Key Findings (Preliminary - 50 seeds, 84K states)

Critical Decision Frequency

Condition	Count	Percentage
Critical in ALL configs	545	**0.6%**
Critical in ANY config	22,150	**26.3%**

Key Finding: Only 0.6% of positions have Q-gap > 5 in ALL opponent configs. Most high-stakes decisions are opponent-dependent.

Consistency of Critical Decisions

Outcome	Count	Percentage
Same best move	347	**63.7%**
Different best moves	198	**36.3%**

Finding: When a decision is critical in all configs, there's only 63.7% chance the best move is the same. Over a third of critical decisions depend on opponent hands.

Q-Gap Analysis

Metric	Value
Mean Q-gap	1.86
Median Q-gap	0.00
% with Q-gap > 5	13.2%
% with Q-gap > 10	4.4%

Insight: Most positions (87%) have Q-gap < 5 - the moves are roughly equivalent. Only 4.4% have Q-gap > 10 (high-stakes).

Critical Decisions by Depth

Depth	Count	Percentage
5 (late game)	396	**72.7%**
9 (mid game)	148	**27.2%**
13 (early)	1	0.2%

Insight: Critical decisions concentrate in late game (depth 5). Early decisions rarely have large Q-gaps because outcomes depend heavily on future play.

Implications for Strategy

1. **Most decisions don't matter much**: 87% of positions have Q-gap < 5. Playing "reasonably" is usually good enough.
2. **Critical decisions ARE opponent-dependent**: 36.3% of high-stakes positions have different best moves depending on who holds what.
3. **Late game is where it counts**: 73% of critical decisions occur at depth 5. Focus your calculation here.

4. **Opponent inference helps at key moments**: When Q-gap is large AND opponents' hands affect the answer (36% of the time), reading opponents is valuable.

Relationship to Other Analyses

- **11c** found 54.5% overall best-move consistency
- **11o** found 97% robustness on common states
- **11n** adds nuance: for CRITICAL decisions specifically, only 64% are consistent

Files Generated

- `results/tables/11n_decision_consistency_by_seed.csv` - Per-seed data
- `results/tables/11n_decision_consistency_summary.csv` - Summary
- `results/figures/11n_decision_consistency.png` - Visualization

11v: Hand Similarity Clustering

Key Question

Do structurally similar hands have similar outcomes?

Method

Cluster hands by FEATURES (doubles, trump count, count points, etc.), then measure within-cluster E[V] variance.

Key Findings (200 hands)

Feature-Based Clusters

Cluster	n	%	E[V]	σ(E[V])	Characteristics
Multi-Double/Trump-Heavy	34	17%	+22.7	13.9	2.2 doubles, 2.4 trumps

Cluster	n	%	E[V]	σ(E[V])	Characteristics
Multi-Double/Trump-Light	33	16%	+20.1	17.7	2.9 doubles, 0.4 trumps
Count-Rich/Six-Heavy	37	18%	+13.0	10.6	17 count points, 2.7 6-highs
Few-Double/Count-Poor	55	28%	+5.8	18.2	1.0 doubles, 4 count points
Mixed	41	20%	+13.2	15.0	Average features

Observation: Multi-double hands have highest E[V] (~+21) regardless of trump count.

Within-Cluster Variance Analysis

Metric	Value
Overall E[V] std	16.62
Within-cluster E[V] std	15.08
Variance reduction	**9%**
Improvement over random	**9%**

Critical Finding: Feature clustering only explains **9% of E[V] variance**. Structurally similar hands do NOT guarantee similar outcomes.

Best and Worst Clusters

Cluster	Variance Reduction	Interpretation
Count-Rich/Six-Heavy	**36%**	Most predictable
Multi-Double/Trump-Heavy	16%	Moderately predictable
Multi-Double/Trump-Light	**-7%**	More variance than average
Few-Double/Count-Poor	**-10%**	Highly unpredictable

Insight: Holding many counts makes outcomes more predictable, while few doubles/counts leads to extreme variance.

Interpretation

This confirms the "luck factor" finding from 11a and 11f:

1. **Hand features explain ~25% of E[V]** (from 11f regression)
2. **Feature clustering explains ~9% of E[V] variance** (from 11v)
3. **Opponent distribution dominates** - the remaining 75%+ comes from who holds what

Why Feature Similarity Fails

Two hands with identical (doubles, trumps, counts) can have very different outcomes because: - The specific dominoes matter (which suits, which ranks) - Opponent hands can be favorable or unfavorable - The interaction between your hand and opponents' is unpredictable

Implications for Bidding

1. **Don't assume similar = equivalent**: Two hands with "3 doubles, 2 trumps" can have wildly different outcomes.
2. **Features are necessary but not sufficient**: Count doubles/trumps/counts for general guidance, but recognize high variance.
3. **The 28% problem**: "Few-Double/Count-Poor" hands (28% of all hands) are the most unpredictable. Pass these.
4. **Count-rich hands are safest**: Best variance reduction (36%) - holding counts makes you more predictable.

Relationship to Other Analyses

- **11f** found $R^2 = 0.25$ for features → E[V]
- **11k** clustered by outcomes (STRONG/VOLATILE/WEAK)
- **11v** clusters by features - confirms features don't fully determine outcomes

Files Generated

- `results/tables/11v_hand_similarity.csv` - Per-hand clusters

- `results/tables/11v_cluster_summary.csv` - Cluster statistics
- `results/tables/11v_variance_analysis.csv` - Variance analysis
- `results/figures/11v_hand_similarity.png` - Visualization

11p: Path Similarity Analysis (DTW)

Key Question

How similar are V-trajectories across opponent configs?

Method

For each hand across 3 opponent configurations: 1. Sample V distributions at depth levels (28, 24, 20, 16, 12, 8, 4, 1) 2. Compute mean V at each depth level 3. Compare depth-V trajectories using DTW and Pearson correlation

Key Findings (Full 201 seeds)

Path Stability Summary

Metric	Value
Mean trajectory correlation	**0.316**
Median trajectory correlation	0.272
Min trajectory correlation	**-0.998**
Mean DTW distance	5.08

Critical Finding: V-trajectories show LOW correlation across opponent configurations (mean 0.316). Most trajectories are weakly correlated or uncorrelated.

Stability Categories

Category	Count	Percentage
High stability (corr > 0.9)	18	**9.0%**
Medium stability (0.7-0.9)	26	**13.0%**
Low stability (corr ≤ 0.7)	156	**78.0%**

Finding: 78% of hands have low stability - the V trajectory through the game varies significantly based on opponent hands. Only 9% maintain highly similar trajectories.

Root vs Terminal V Spread

Depth	Mean Spread	Max Spread
Root (depth 28)	**35.0 points**	82 points
Terminal (depth 1)	**2.7 points**	9 points

Key Insight: Games START with high V divergence (35 points) but CONVERGE toward similar endpoints (2.7 points). The path to get there varies wildly.

Path-Value Correlations

Correlation	Value
DTW vs root V spread	**+0.860**
Corr vs root V spread	**-0.779**
DTW vs terminal V spread	**+0.717**

Critical Finding: Very strong correlation (+0.86) between initial V spread and path divergence. Hands that start with high uncertainty in outcomes have the most divergent trajectories through the game.

Reconciling with 11x (Information Value)

This seems to contradict 11x which found 75% action agreement:

- **11x**: At any given position, the best *move* is usually the same regardless of opponents (75%)
- **11p**: The overall *V trajectory* through the game diverges significantly (78% low correlation)

Resolution: Individual moves are robust, but the cumulative effect of different opponent distributions leads to very different game progressions. You often make the same move, but the consequences (V values at each stage) differ dramatically.

Implications for Strategy

1. **Games converge at the end**: Terminal V spread is only 2.7 points despite 35 points at start. Endgames are predictable.
2. **Early/mid game trajectories diverge**: The path from start to end varies based on opponents, even if individual moves are similar.
3. **High-variance hands have divergent paths**: The +0.86 correlation means that hands with uncertain V also have the most unpredictable game progressions.
4. **Focus on endgame**: Since games converge, endgame calculation becomes increasingly reliable.
5. **Only 9% are predictable**: Just 9% of hands maintain highly similar trajectories regardless of opponent distribution. These are the safest bids.

Files Generated

- `results/tables/11p_path_similarity_by_seed.csv` - Per-seed data
- `results/tables/11p_path_similarity_summary.csv` - Summary
- `results/figures/11p_path_similarity.png` - Visualization

11q: Per-Hand PCA Analysis (Preliminary)

Key Question

Is the 5D structure (V at multiple depths) preserved within a fixed hand?

Method

For each hand across 3 opponent configurations: 1. Extract V statistics at depth levels (28, 24, 20, 16, 12, 8, 4, 1) 2. Build feature matrix: mean V, std V, and spread (max-min) at each depth 3. PCA to find intrinsic dimensionality

Key Findings (Preliminary - 50 seeds)

PCA Variance Explained

Component	Variance	Cumulative
PC1	**45.6%**	45.6%
PC2	17.3%	62.9%
PC3	15.5%	78.4%
PC4	7.4%	85.8%
PC5	4.5%	90.3%

Key Finding: 5 components explain 90% of variance from 24 original features.

Dimensionality Metrics

Metric	Value
Original features	24
Components for 90% variance	**5**

Metric	Value
Components for 95% variance	7
Components for 99% variance	10
Effective dimensionality	**4.9**
Dimensionality compression	**4.8x**

Critical Finding: Fixing P0's hand constrains the outcome manifold from 24 dimensions to ~5. This is a significant compression.

PC1 Loadings (Top Features)

Feature	Loading
v_spread_d8	**+0.385**
v_spread_d12	+0.377
v_spread_d4	+0.366
v_spread_d16	+0.355
v_spread_d20	+0.325

Insight: PC1 is dominated by V SPREAD at mid-game depths. This represents the "uncertainty dimension" - how much V varies across opponent configurations.

V Spread by Depth

Depth	Mean V Spread
28 (start)	**40.6**
24	18.3

Depth	Mean V Spread
20	11.5
16	8.6
12	6.3
8	5.0
4	4.0
1 (end)	**3.0**

Key Pattern: V spread decreases monotonically from 41 points at game start to 3 points at end. Games converge as they progress.

Interpretation

1. **Manifold structure exists**: The 4.8x compression shows that hand-conditioned outcomes live on a ~5D manifold, not the full 24D feature space.
2. **Uncertainty is the main axis**: PC1 (45.6%) captures V spread - the primary variation between hands is how much their outcomes depend on opponents.
3. **Convergence is universal**: The V spread → depth relationship is consistent across hands, suggesting a shared funnel structure.
4. **Hand constrains but doesn't determine**: 5 dimensions still allow significant outcome variation within a fixed hand.

Relationship to Other Analyses

- **11p** found 88% low-correlation trajectories - 11q explains this via the high-spread PC1 loadings
- **11j** found 82% of hands cross multiple basins - aligns with the 5D manifold allowing diverse outcomes
- **11f** found $R^2 = 0.25$ - the remaining 75% variance maps to the 5 PCA dimensions

Implications

1. **Bidding heuristics are 5D**: A good bidding formula needs ~5 independent factors
2. **Mid-game depth matters most**: PC1 loadings peak at depths 8-16, not early or late game
3. **Convergence is reliable**: The funnel structure (41→3 spread) means endgame analysis is stable

Files Generated

- `results/tables/11q_per_hand_pca_features.csv` - Per-hand features
- `results/tables/11q_pca_variance.csv` - Variance explained
- `results/tables/11q_pca_loadings.csv` - Component loadings
- `results/tables/11q_pca_summary.csv` - Summary statistics
- `results/figures/11q_per_hand_pca.png` - Visualization

11r: Manifold Collapse Analysis

Key Question

Do strong hands collapse to lower effective dimensionality?

Method

For each hand across 3 opponent configurations: 1. Build depth × config matrix of mean V values (8 depths × 3 configs) 2. Decompose variance: between-config, between-depth, residual 3. Compute collapse score: 1 - (config_dim_ratio) 4. Correlate with E[V] to test "strong hands collapse" hypothesis

Key Findings (100 seeds)

Variance Decomposition

Component	Variance	% of Total
Between-config	68.4	**36.7%**
Between-depth	50.8	**37.1%**
Residual	42.7	26.2%

Finding: Config and depth contribute roughly equally to V variance. About 37% of variance comes from opponent configuration.

Collapse Hypothesis: CONFIRMED

Metric	Correlation with E[V]
collapse_score	**+0.369**
trajectory_correlation	**+0.476**
config_dim_ratio	**-0.369**

Critical Finding: Strong hands DO collapse more (r = +0.37). Higher E[V] hands have more predictable outcomes across opponent configurations.

Strong vs Weak Hands

Metric	High E[V] (top 25%)	Low E[V] (bottom 25%)	Difference
Mean E[V]	+33.5	-10.7	+44.2
Collapse score	**0.845**	**0.562**	+0.283
Trajectory corr	**0.788**	**0.081**	+0.707

Metric	High E[V] (top 25%)	Low E[V] (bottom 25%)	Difference
Config dim ratio	0.155	0.438	-0.283

Key Insights: 1. Strong hands have 0.28 higher collapse score (more predictable) 2. Strong hands have 7x higher trajectory correlation (0.79 vs 0.08) 3. Weak hands have 28% more config-dependent variance

Collapse Categories

Category	Count	%	Mean E[V]
Highly collapsed (>0.8)	27	**27%**	+21.3
Moderately collapsed	33	**33%**	+12.8
Not collapsed (≤0.5)	40	**40%**	+4.5

Insight: About 1/4 of hands are "highly collapsed" with predictable outcomes. These are the best bidding hands.

Interpretation

1. **The collapse hypothesis is confirmed**: Strong hands (high E[V]) have outcomes that depend LESS on opponent distribution. This is why doubles and trump length predict good hands - they collapse the outcome manifold.
2. **Weak hands are opponent-dependent**: Low E[V] hands have 40% of their variance explained by opponent configuration. You don't know what you'll get.
3. **Trajectory coherence separates strong from weak**: Strong hands follow similar V progressions regardless of opponents (corr = 0.79). Weak hands diverge wildly (corr = 0.08).
4. **The 27% rule**: About 27% of hands are "highly collapsed" - these are hands where bidding is safe because opponents matter less.

Relationship to Other Analyses

- **11f** found $R^2 = 0.25$ for hand features → E[V]. Collapse analysis explains WHY: features that predict collapse (doubles, trumps) also predict E[V]
- **11s** found negative E[V] vs σ(V) correlation. 11r confirms: strong hands collapse to lower variance
- **11j** found 18.5% basin convergence. Aligns with 27% highly collapsed (similar concept)

Implications for Bidding

1. **Doubles cause collapse**: Multiple doubles constrain opponents' responses, reducing config-dependent variance
2. **Trump length causes collapse**: Long trump suits control the game trajectory regardless of opponents
3. **Bid on collapsible hands**: Look for hands where your outcome is predictable (high collapse score)
4. **Avoid non-collapsed hands**: Hands with collapse score < 0.5 are gambles - 40% of variance comes from luck

Files Generated

- `results/tables/11r_manifold_collapse_by_seed.csv` - Per-seed metrics
- `results/tables/11r_manifold_collapse_summary.csv` - Summary statistics
- `results/figures/11r_manifold_collapse.png` - Visualization

11x: Information Value (Perfect vs Imperfect)

Key Question

How much does knowing opponent hands help?

Method

Compare Q-values with perfect info (knowing which opponent holds which cards) vs imperfect info (average Q across opponent configurations).

- **Perfect info**: Best move under each config separately, value = Q[best_action]
- **Imperfect info**: Best move using average Q across configs, value = mean(Q[avg_best_action])
- **Information gain**: Perfect value - Imperfect value

Key Findings (Full analysis: 49 seeds, 31,654 states)

Information Value Summary

Metric	Value
Mean information gain	**0.54 points**
Median information gain	**0.00 points**
Max information gain	18.7 points

Critical Finding: Knowing opponent hands gains only **0.5 points on average**. The median is 0 - in most positions, perfect information provides no advantage.

How Often Does Perfect Info Help?

Threshold	Percentage
Any benefit (>0)	**26.8%**
Significant (>2 pts)	**14.5%**
Large (>5 pts)	**4.3%**

Insight: Only 27% of positions benefit from perfect info. The vast majority (73%) have the same optimal move regardless of whether you know opponent hands.

Action Agreement Rate

Metric	Value
Perfect/Imperfect agreement	**73.7%**

Key Finding: 74% of the time, the best move under perfect information is the SAME as the best move under imperfect information. Opponent inference provides only marginal improvement.

Information Value by Game Phase

Depth	Mean Info Gain	n
1 (near end)	+0.00	915
5 (late)	+0.55	56,567
9 (mid)	+1.43	25,969
13 (early)	+2.70	729
17 (very early)	+8.83	2

Insight: Information value increases with game depth. Early in the game, knowing opponent hands is worth ~3-9 points. By endgame, it's worth essentially nothing.

Interpretation

This is perhaps the most surprising finding of the entire imperfect information analysis:

1. **Information has marginal value**: Contrary to intuition, knowing opponent hands typically doesn't change what you should do
2. **Most positions are "dominant"**: The best move is best regardless of opponent distribution
3. **Early game is where it matters**: Information value peaks at depth 17+ (very early game), where there's maximum uncertainty

4. **Endgame is deterministic**: At depth 1-5, perfect information adds virtually nothing

Reconciling with Other Findings

- **11c** found 54.5% best move consistency → different from 75% agreement here
- **Explanation**: 11c counted divergent paths separately; 11x averages across opponent configs within the same position
- **11n** found 36% of critical decisions are opponent-dependent → aligns with 27% benefiting from perfect info
- **11o** found 97% of common states have robust best moves → strong alignment

Implications for Strategy

1. **Don't overthink opponent hands**: 75% of the time, the right move is the right move regardless
2. **Focus on your own play**: With only 0.8 expected points from perfect info, execution matters more than reading
3. **Early game exceptions**: The 6% of positions with >5 point info value ARE worth careful analysis
4. **Simplify decision-making**: For most positions, use heuristics instead of modeling opponents

The Practical Takeaway

> *"Play the board, not the player."*

Opponent inference adds <1 point of expected value on average. Unless you're in the rare (6%) high-value situations, focus on basic strategy.

Files Generated

- `results/tables/11x_information_value_by_seed.csv` - Per-seed data
- `results/tables/11x_information_value_summary.csv` - Summary
- `results/figures/11x_information_value.png` - Visualization

11y: Outcome Variance Decomposition

Key Question

How much of outcome variance comes from the dealt hand vs opponent distribution?

Method

ANOVA-style variance decomposition: - Total variance = Between-hand variance + Within-hand variance - Between-hand = variance explained by which hand P0 was dealt - Within-hand = variance across opponent configurations (same hand, different opponents)

Important caveat: This analysis measures variance sources, NOT player skill. The oracle plays perfectly with perfect information - no human decisions are measured here.

Key Findings (Full 201 seeds, 600 observations)

ANOVA Variance Decomposition

Component	Sum of Squares	% of Total
Between-hand (which hand you got)	164,898	**47.0%**
Within-hand (which opponents got)	186,221	**53.0%**
Total	351,119	100%

Critical Finding: Outcome variance splits roughly evenly between hand quality (47%) and opponent distribution (53%). Neither factor dominates.

What this means: Even with perfect play (like the oracle), 53% of outcome variance is determined by opponent cards - this is irreducible through better play. It's baked into the deal.

Variance Hierarchy

Component	% of Total Variance	Interpretation
Predictable hand effect	**11.6%**	Explainable by hand features (doubles, trumps, etc.)
Unpredictable hand effect	**35.4%**	Hand matters, but simple features don't capture why
Correlated opponent effect	**3.9%**	Opponent variance that correlates with hand
Pure opponent effect	**49.2%**	Irreducible variance from opponent distribution

Interpretation: - 11.6% of variance can be predicted from features like doubles, trumps, counts - 35.4% comes from hand effects we can't predict from simple features - 49% is determined by opponent distribution alone

Note: None of this is "skill" - you don't choose your hand. The 47% between-hand variance represents the random deal, not player decisions.

Feature Correlations

Feature	Correlation with E[V]	Correlation with σ(V)
n_doubles	**+0.40**	**-0.14**
trump_count	**+0.23**	-0.09
has_trump_double	**+0.24**	-0.10
count_points	+0.20	-0.09
n_6_high	**-0.16**	**+0.19**
total_pips	+0.04	+0.15

Key Insight: Features that predict high E[V] (doubles, trumps) ALSO predict low σ(V). Strong hands are both higher value AND more deterministic.

The Negative Risk-Return Correlation

Metric	Value
E[V] vs σ(V) correlation	**-0.381**

Critical Finding: Strong hands have LESS variance (negative correlation). This is the opposite of typical financial markets where higher return = higher risk.

In Texas 42: strong hands are both better AND more predictable. There's no risk-return tradeoff to navigate.

Variance by Hand Quality

Hand Category	Mean E[V]	Mean σ(V)	Mean Spread	Count
Strong (>25)	+33.0	**11.1**	20	59
Good (10-25)	+16.8	21.4	41	60
Weak (-10 to 10)	+2.7	21.7	42	63
Very Weak (<-10)	-19.4	21.5	41	18

Insight: Strong hands have HALF the variance (σ=11) of other hands (σ=21). Strong hands collapse the outcome space - opponents matter less.

What This Analysis Does NOT Measure

This decomposition characterizes the **game's structure**, not player skill:

NOT Measured	Why It Matters
Bidding decisions	Players choose when to compete for contract

NOT Measured	Why It Matters
Play execution	Humans don't play like the oracle
Opponent exploitation	Oracle assumes perfect opponents
Partner coordination	Real games involve signaling
Learning/adaptation	Skill develops over many games

The 47/53 split is not "skill vs luck" - it's "which random hand you got vs which random hands opponents got." Both are determined by the deal, not player choices.

Relationship to Other Analyses

- **11f** found $R^2 = 0.25$ for features → E[V]. This aligns with 11y's finding that ~12% of total variance is feature-predictable
- **11r** found strong hands collapse more. 11y explains this: strong hands have lower σ(V)
- **11s** found negative E[V] vs σ(V) correlation. 11y confirms with full ANOVA decomposition

Implications for Hand Evaluation

1. **The deal matters enormously**: 100% of this variance is from the deal - 47% from your hand, 53% from opponents.
2. **Strong hands are doubly good**: They have higher E[V] AND lower variance. Look for hands that collapse the outcome space.
3. **Doubles are the best indicator**: They correlate +0.40 with E[V] and -0.14 with σ(V). They make hands both stronger and more predictable.
4. **Avoid 6-high hands**: They correlate -0.16 with E[V] and +0.19 with σ(V). They're both weaker AND less predictable.
5. **Variance is irreducible**: Even perfect play can't overcome the 53% opponent-distribution variance. Accept that outcomes will vary.

The Honest Answer

> *"How much of Texas 42 outcomes depend on the deal vs opponent cards?"*

47% of outcome variance comes from which hand you're dealt. 53% comes from how opponent cards are distributed. Both are determined at the deal - before any player makes any decision.

Of the 47% hand effect, only about 25% (≈12% of total) is predictable from simple features like doubles and trump count. The rest is "hidden hand quality" - the hand matters, but we can't easily say why.

What this analysis CANNOT tell you: How much skilled players outperform unskilled players. That would require comparing human decisions, not oracle perfect play.

Files Generated

- `results/tables/11y_uncertainty_by_hand.csv` - Per-hand data
- `results/tables/11y_uncertainty_summary.csv` - Summary statistics
- `results/figures/11y_reducible_uncertainty.png` - Visualization

11z: Partner Inference (MI) Analysis

Key Question

Does partner's play reveal their hand? Can observing P2's (partner's) actions help reduce uncertainty?

Method

For states where P2 acts that are common across opponent configurations: 1. Compare P2's optimal action (argmax Q) across configs 2. Measure action consistency (same action) vs variation (different actions) 3. Correlate action variance with P2's hand variance

Key Findings (Full analysis: 200 hands, 564M pairwise comparisons)

Metric	Value
Pairwise comparisons per hand	~2.8M

Metric	Value
3-way common states per hand	~308K
Action consistency rate	**80.1%**
Std deviation	8.4%

Insight: 80% of P2's actions are consistent across opponent configs. The remaining 20% vary based on P2's actual hand - these actions reveal information about the hidden hand.

Consistency Distribution

Category	Percentage	Interpretation
High consistency (>90%)	10.5%	Very predictable hands
Medium consistency (50-90%)	89.5%	Most hands
Low consistency (≤50%)	0.0%	None!

Key Finding: No hands show low consistency. Even the most variable hands still have >50% action consistency.

Signaling Potential

Metric	Value	Interpretation
Consistency rate	80%	HIGH
Action variance	20%	Moderate info revealed

Finding: Partner actions are more consistent than previously estimated. The 20% action variance represents a moderate but not dominant source of inference potential.

Hand-Action Correlations

P2 Feature Variance	vs Consistency
Doubles std	**-0.102**
Pips std	+0.106
Count points std	+0.075
Trump count std	-0.010

Key Finding: Correlations are weak across all features, suggesting action consistency is more about game state than hand composition.

Implications for Strategy

1. **Partner actions are mostly predictable**: 80% consistency means partner's hand doesn't usually change optimal play
2. **Moderate inference value**: The 20% action variance provides some inference potential
3. **Game state dominates**: Actions are determined more by position than partner's hidden cards
4. **Aligns with 11x**: Low information value (<1 point) matches high action consistency

Relationship to Other Analyses

- **11c** found 54.5% best-move consistency overall vs 80% for P2 states here
- **11x** found 74% action agreement rate - aligns with 80% P2 consistency
- **11y** found 53% opponent-caused variance. With 80% action consistency, less of this can be inferred from partner actions than expected.

Interpretation

The full analysis substantially revises the preliminary finding: - Original (23 hands): 58% consistency, 42% inference potential - Full (200 hands): **80% consistency, 20% inference potential**

Partner inference has moderate but limited value. Most of the time, partner's optimal action is the same regardless of their hand.

Files Generated

- `results/tables/11z_partner_inference_by_seed.csv` - Per-hand data
- `results/tables/11z_partner_inference_summary.csv` - Summary statistics
- `results/figures/11z_partner_inference.png` - Visualization

Analysis date: 2026-01-07

12: Validate & Scale

Scaling existing analyses to n=201 seeds and recomputing with consistent methodology.

12a: E[V] vs σ(V) Correlation at Scale

Key Question

Is the negative correlation between expected value and risk real, or a small-sample artifact?

Method

Using scipy.stats.pearsonr with Fisher transformation confidence intervals on the full n=200 seed marginalized dataset (from 11s).

Key Findings

Metric	Correlation	95% CI	p-value	Effect Size
r(E[V], σ[V])	**-0.381**	[-0.494, -0.256]	2.6×10^{-8}	medium
r(E[V], V_spread)	**-0.398**	[-0.509, -0.274]	5.4×10^{-9}	medium

Statistical Summary

- **n = 200** base seeds
- **R^2 = 0.145** for σ(V), meaning ~14.5% of variance explained
- **R^2 = 0.158** for V_spread
- **Effect size**: medium by Cohen's conventions (0.3 < |r| < 0.5)

Interpretation

The negative correlation is **confirmed** at scale with high statistical confidence:

1. **Inverse risk-return relationship**: Higher expected value hands also have lower variance - the opposite of typical financial markets
2. **Not spurious**: $p < 10^{-8}$ rules out sampling artifact
3. **Moderate effect**: ~15% of variance explained - meaningful but not dominant

Original Hypothesis Correction

The task hypothesized $r \approx -0.55$. Actual finding: **$r \approx -0.38$ to -0.40**. The effect is real but smaller than initially estimated.

Files Generated

- `results/tables/12a_ev_sigma_correlation.csv` - Summary statistics
- `results/figures/12a_ev_sigma_correlation.png` - Scatter plot with regression

12b: Unified Feature Extraction

Key Question

Can we consolidate the duplicated feature extraction code across 7+ run_11*.py scripts?

Method

Created `forge/analysis/utils/hand_features.py` with: - `extract_hand_features(hand, trump_suit)` - single source of truth - `HAND_FEATURE_NAMES` - consistent column ordering - `REGRESSION_FEATURES` - subset for ML models

Output

Master feature file: `results/tables/12b_unified_features.csv` - 200 base seeds × 20 columns - V statistics: E[V], σ(V), V_spread, V_min, V_max - 12 hand features

Feature Summary

Feature	Mean	Range	r with E[V]
n_doubles	1.73	[0, 4]	**+0.395**
trump_count	1.32	[0, 5]	+0.229
has_trump_double	0.17	[0, 1]	+0.242
count_points	9.20	[0, 25]	+0.197
n_voids	0.67	[0, 3]	+0.200
n_6_high	1.74	[0, 4]	**-0.161**

Key Findings

1. **n_doubles is king**: Strongest predictor of E[V] (r = +0.40, $p < 10^{-8}$)
2. **Trump features matter**: trump_count and has_trump_double both positive predictors
3. **6-highs are risky**: Negative correlation with E[V] (-0.16)
4. **Total pips irrelevant**: Near-zero correlation (+0.04) - raw hand strength doesn't predict outcomes

Files Generated

- `utils/hand_features.py` - Unified feature extraction module
- `results/tables/12b_unified_features.csv` - Master feature dataset

Remaining Tasks

- Additional validation tasks TBD based on epic t42-1wp2

13: Statistical Rigor

Adding confidence intervals, effect sizes, and rigorous statistical testing to key findings.

13a: Bootstrap CIs for Regression Coefficients

Key Question

How certain are the 11f regression coefficients? Are they statistically significant?

Method

- Bootstrap resampling: 1000 iterations
- Percentile confidence intervals (95%)
- Linear regression: hand features → E[V]

Key Findings

Feature	Coefficient	95% CI	Width	Significant?
n_doubles	**+5.7**	[+2.3, +9.2]	6.9	**Yes**
trump_count	**+3.2**	[+1.3, +4.7]	3.5	**Yes**
has_trump_double	+2.8	[-2.6, +8.4]	11.0	No
n_voids	+2.8	[-3.5, +8.9]	12.4	No
n_6_high	-1.6	[-5.0, +1.8]	6.8	No
max_suit_length	-0.7	[-6.2, +4.4]	10.6	No
n_5_high	-0.5	[-3.3, +2.1]	5.4	No
count_points	+0.2	[-0.1, +0.5]	0.6	No

Feature	Coefficient	95% CI	Width	Significant?
total_pips	+0.1	[-0.4, +0.6]	1.0	No
n_singletons	-0.3	[-3.7, +3.5]	7.3	No

Model Performance

Metric	Value	95% CI
R^2	0.259	[0.197, 0.400]
Intercept	-2.9	[-22.3, +17.4]

Critical Insight: Only Two Significant Predictors

Of 10 hand features, **only n_doubles and trump_count** have confidence intervals that exclude zero.

This is a major refinement from 11f/12b: - **Original claim**: "n_doubles (+5.7), trump_count (+3.2), has_trump_double (+2.8), n_voids (+2.8) all predict E[V]" - **Refined claim**: "Only n_doubles and trump_count are statistically significant; other features have too much uncertainty"

The Robust Napkin Formula

```
E[V] ≈ -3 + 5.7×(doubles) + 3.2×(trump_count)
```

Everything else is noise. The simpler 2-feature model may generalize better than the 10-feature model.

Implications for Bidding

1. **Count your doubles**: The most reliable signal (+5.7 points per double)
2. **Count your trumps**: Second most reliable (+3.2 points per trump)

3. **Everything else is uncertain**: has_trump_double, n_voids, and n_6_high all have CIs that include zero

Files Generated

- `results/tables/13a_bootstrap_coefficients.csv` - Full coefficient table with CIs
- `results/figures/13a_bootstrap_regression_ci.png` - Forest plot visualization

13b: Bootstrap CIs for Risk Formula

Key Question

Are the risk formula coefficients (predicting σ(V)) statistically significant?

Method

Same bootstrap approach as 13a: 1000 iterations, percentile CIs.

Key Findings

Feature	Coefficient	95% CI	Significant?
total_pips	**+0.30**	[+0.01, +0.57]	**Marginal**
n_doubles	-1.40	[-3.32, +0.77]	No
n_5_high	-1.09	[-2.84, +0.63]	No
trump_count	-0.61	[-1.56, +0.42]	No
has_trump_double	-0.55	[-4.51, +3.08]	No
n_voids	+0.53	[-3.49, +4.38]	No

Model Performance

Metric	Value	95% CI
R^2	0.081	[0.058, 0.203]

Critical Insight: Risk is Unpredictable

Only **total_pips** barely reaches significance (CI just excludes zero at [+0.01, +0.57]).

All other features have CIs that include zero. Combined with R^2 of only 6-20%, this confirms:

Risk (outcome variance) is fundamentally unpredictable from hand features alone.

The uncertainty in Texas 42 comes from opponent hand distribution, not from your own hand quality.

Implications

1. **Don't try to assess hand "riskiness"**: No reliable signal exists
2. **Focus on E[V] predictors**: n_doubles and trump_count are meaningful; risk is not
3. **Embrace uncertainty**: Half of game outcomes are determined by luck (opponent hands)

Files Generated

- `results/tables/13b_bootstrap_risk_coefficients.csv` - Full coefficient table with CIs
- `results/figures/13b_bootstrap_risk_ci.png` - Forest plot visualization

13c: Effect Sizes for Key Comparisons

Key Question

Are our findings practically meaningful, or just statistically significant?

Method

Computed standardized effect sizes: - **r**: Pearson correlation (|r| < 0.1 negligible, 0.1-0.3 small, 0.3-0.5 medium, >0.5 large) - **Cohen's d**: Standardized mean difference (|d| < 0.2 negligible, 0.2-0.5 small, 0.5-0.8 medium, >0.8 large) - **R^2**: Proportion of variance explained (<0.01 negligible, 0.01-0.09 small, 0.09-0.25 medium, >0.25 large)

Key Findings

Correlation Effect Sizes

Comparison	r	r^2	Magnitude
E[V] vs σ(V)	-0.38	0.15	**Medium**
E[V] vs V_spread	-0.40	0.16	**Medium**
n_doubles vs E[V]	+0.40	0.16	**Medium**
trump_count vs E[V]	+0.23	0.05	Small
count_points vs E[V]	+0.20	0.04	Small
n_6_high vs E[V]	-0.16	0.03	Small
total_pips vs E[V]	+0.04	0.00	Negligible

Group Comparison Effect Sizes (Cohen's d)

Comparison	d	Magnitude
≥2 doubles vs <2 on E[V]	+0.76	**Medium**
High risk vs Low risk on E[V]	-0.79	**Medium**
≥2 trumps vs <2 on E[V]	+0.48	Small
≥15 count pts vs <15 on E[V]	+0.27	Small

Regression R^2 (Variance Explained)

Model	R^2	%	Magnitude
Hand features → E[V]	0.26	26%	**Large**
Hand features → σ(V)	0.08	8%	Small

Practical Significance Summary

Medium/Large Effects (Practically Meaningful): 1. **n_doubles → E[V]**: r = +0.40, d = +0.76 — Real impact on outcomes 2. **E[V] ↔ σ(V)**: r = -0.38 — Genuine inverse risk-return relationship 3. **Hand features → E[V]**: R^2 = 0.26 — Useful prediction

Small Effects (Modest Impact): - trump_count, count_points have small correlations with E[V] - Risk prediction (R^2 = 0.08) is weak

Negligible Effects: - total_pips → E[V]: r = +0.04 — Raw hand "strength" doesn't matter

Files Generated

- `results/tables/13c_effect_sizes.csv` - Full effect size summary
- `results/figures/13c_effect_sizes.png` - Visualization

13d: Fisher z-Transform Correlation CIs

Key Question

Which correlations are statistically significant when properly accounting for sampling uncertainty?

Method

- scipy.stats.pearsonr with confidence_interval() method
- Fisher z-transformation: z = arctanh(r), SE = 1/sqrt(n-3)

- 95% confidence intervals via inverse transform

Key Findings

Significant Correlations (10 of 16)

Comparison	r	95% CI	Magnitude
E[V] vs V_spread	-0.40	[-0.51, -0.27]	**Medium**
n_doubles vs E[V]	+0.40	[+0.27, +0.51]	**Medium**
E[V] vs σ(V)	-0.38	[-0.49, -0.26]	**Medium**
has_trump_double vs E[V]	+0.24	[+0.11, +0.37]	Small
trump_count vs E[V]	+0.23	[+0.09, +0.36]	Small
n_voids vs E[V]	+0.20	[+0.06, +0.33]	Small
count_points vs E[V]	+0.20	[+0.06, +0.33]	Small
n_6_high vs σ(V)	+0.19	[+0.05, +0.32]	Small
n_6_high vs E[V]	-0.16	[-0.29, -0.02]	Small
total_pips vs σ(V)	+0.15	[+0.01, +0.28]	Small

Non-Significant Correlations (6 of 16)

Comparison	r	95% CI	Note
n_doubles vs σ(V)	-0.14	[-0.27, +0.00]	Marginal
trump_count vs σ(V)	-0.09	[-0.23, +0.05]	Negligible
max_suit_length vs E[V]	-0.08	[-0.22, +0.06]	Negligible

Comparison	r	95% CI	Note
n_5_high vs E[V]	+0.08	[-0.06, +0.21]	Negligible
total_pips vs E[V]	+0.04	[-0.10, +0.17]	Negligible
n_singletons vs E[V]	+0.00	[-0.14, +0.14]	Negligible

Critical Insight: Bivariate vs Multivariate

The Fisher z-transform CIs reveal an important distinction:

Bivariately significant but multivariately not: - has_trump_double vs E[V]: r = +0.24 (significant) - n_voids vs E[V]: r = +0.20 (significant) - count_points vs E[V]: r = +0.20 (significant)

Yet in the multivariate regression (13a), these features have CIs that include zero. This means their bivariate correlations are largely explained by their association with n_doubles and trump_count.

Implications

1. **Bivariate screening is encouraging**: Many features correlate with E[V]
2. **Multivariate tells the real story**: Only n_doubles and trump_count survive
3. **Risk remains unpredictable**: Even with CIs, σ(V) predictors are weak

Files Generated

- `results/tables/13d_correlation_cis.csv` - Full correlation table with Fisher CIs
- `results/figures/13d_correlation_cis.png` - Forest plot visualization

13e: Power Analysis

Key Question

Is n=200 sufficient to detect our observed effects with adequate statistical power?

Method

- Power functions for correlation tests (t-distribution with non-centrality parameter)
- statsmodels.stats.power for Cohen's d (TTestIndPower)
- F-test power for regression R^2
- Target: 80% power at $\alpha=0.05$

Key Findings

Power for Current Sample Size (n=200)

Analysis	Effect Size	Type	Power	n for 80%	Sufficient?
r(E[V], σ[V])	-0.38	r	1.000	51	✓
r(n_doubles, E[V])	+0.40	r	1.000	46	✓
r(trump_count, E[V])	+0.23	r	0.911	145	✓
d(≥2 doubles vs <2)	0.76	d	1.000	29×2=58	✓
d(high vs low risk)	0.79	d	1.000	27×2=54	✓
d(≥2 trumps vs <2)	0.48	d	0.922	70×2=140	✓
R^2(hand→E[V])	0.26	R^2	1.000	57	✓
R^2(hand→σ[V])	0.08	R^2	0.810	197	✓

Sample Size Requirements for 80% Power

Correlations: | Target r | n needed | |----------|----------| | 0.1 (small) | 782 | | 0.2 (small-medium) | 193 | | 0.3 (medium) | 84 | | 0.4 (medium) | 46 | | 0.5 (large) | 29 |

Group Comparisons (Cohen's d): | Target d | n per group | |----------|-------------| | 0.2 (small) | 394 | | 0.5 (medium) | 64 | | 0.8 (large) | 26 |

Critical Insight: n=200 is Sufficient

All key findings are well-powered: 1. **r(E[V], σ[V]) = -0.38**: Power ≈ 1.00 — would only need n=51 2. **r(n_doubles, E[V]) = +0.40**: Power ≈ 1.00 — would only need n=46 3. **d(≥2 doubles vs <2) = 0.76**: Power ≈ 1.00 — would only need n=58 total

Borderline but adequate: - R^2(hand→σ[V]) = 0.08: Power = 0.81, would need n=197 for 80% - We have n=200, so this is just at the threshold

Implications

1. **No immediate scale-up needed**: n=200 provides >80% power for all key findings
2. **Main effects are robust**: Core relationships (E[V]-σ[V], n_doubles) have power ≈ 1.00
3. **If detecting smaller effects**: To find r=0.1 effects, would need n≈782
4. **Risk model is at limit**: The weak R^2=0.08 for σ(V) just barely achieves 80% power

Files Generated

- `results/tables/13e_power_analysis.csv` - Summary table
- `results/figures/13e_power_curves.png` - Power curves for correlations, d, and R^2

13f: Multiple Comparison Correction (BH FDR)

Key Question

Do our significant findings survive correction for multiple testing?

Method

- Benjamini-Hochberg (BH) false discovery rate correction
- Controls FDR (expected proportion of false discoveries)
- Also compared with Bonferroni, Holm, Sidak for reference
- statsmodels.stats.multitest.multipletests

Key Findings

Tests by Correction Method

Method	Significant	Type
Uncorrected	10	None
BH FDR	**9**	**FDR**
BY FDR	8	FDR
Holm	6	FWER
Bonferroni	5	FWER
Sidak	5	FWER

Tests Surviving BH FDR Correction

Comparison	r	p_raw	p_adj_BH
E[V] vs V_spread	-0.40	5.4×10^{-9}	<0.0001
n_doubles vs E[V]	+0.40	6.9×10^{-9}	<0.0001
E[V] vs σ(V)	-0.38	2.6×10^{-8}	<0.0001
has_trump_double vs E[V]	+0.24	5.6×10^{-4}	0.0022

Comparison	r	p_raw	p_adj_BH
trump_count vs E[V]	+0.23	1.1×10^{-3}	0.0036
n_voids vs E[V]	+0.20	4.5×10^{-3}	0.0119
count_points vs E[V]	+0.20	5.2×10^{-3}	0.0119
n_6_high vs σ(V)	+0.19	6.8×10^{-3}	0.0135
n_6_high vs E[V]	-0.16	2.3×10^{-2}	0.0412

Test Lost to BH FDR

Comparison	r	p_raw	p_adj_BH
total_pips vs σ(V)	+0.15	0.035	0.056

Critical Insight: Core Findings Are Robust

9 of 10 significant correlations survive BH FDR correction at α = 0.05.

The only test lost (total_pips vs σ(V)) was marginal anyway: - Effect size: small (r = 0.15) - p_adj = 0.056 (just above threshold)

All key findings survive: 1. E[V] vs σ(V) (r = -0.38) — headline inverse risk-return 2. n_doubles vs E[V] (r = +0.40) — strongest predictor 3. trump_count vs E[V] (r = +0.23) — second key predictor

FWER vs FDR

- **FWER** (Bonferroni, Holm): Controls family-wise error rate (any false positive)
- **FDR** (BH, BY): Controls expected proportion of false discoveries

For exploratory analysis with 16 tests, **FDR is appropriate**. We accept that ~1 in 20 discoveries may be false positive.

Files Generated

- `results/tables/13f_multiple_comparison.csv` - Full table with adjusted p-values
- `results/figures/13f_multiple_comparison.png` - Visualization of BH procedure

13g: Cross-Validation for Regressions

Key Question

How well do our regression models generalize to unseen data?

Method

- 10-fold CV with 10 repeats (100 total fits)
- Compared napkin formula (2 features) vs full model (10 features)
- Also tested Ridge regularization
- sklearn.model_selection.RepeatedKFold

Key Findings

E[V] Prediction

Model	Train R^2	CV R^2	Overfit Ratio
Napkin (2 features)	0.233	**0.152 ± 0.21**	**1.5x**
Full (10 features)	0.259	0.109 ± 0.25	2.4x
Ridge Full	0.259	0.111 ± 0.24	2.3x

Key insight: The napkin formula **generalizes better** than the full model: - Lower overfit ratio (1.5x vs 2.4x) - Higher CV R^2 (0.152 vs 0.109) - Simpler models win for this task

σ(V) Prediction

Model	Train R^2	CV R^2	Status
Napkin	0.030	-0.065 ± 0.17	**FAILS**
Full	0.081	-0.126 ± 0.20	**FAILS**
Ridge Full	0.081	-0.123 ± 0.20	**FAILS**

Key insight: σ(V) prediction **completely fails cross-validation**: - All models have negative CV R^2 (worse than predicting the mean) - Any training R^2 is pure overfitting - Confirms 13b, 14b: risk is fundamentally unpredictable

Critical Insight: Napkin Formula Wins

The 2-feature napkin formula (n_doubles, trump_count) is the **best model**: 1. Lowest overfitting 2. Best generalization 3. Simplest interpretation

Recommended formula: $E[V] \approx -3 + 5.7 \times (\text{doubles}) + 3.2 \times (\text{trumps})$

Implications

1. **Simpler is better**: Adding features hurts generalization
2. **Don't predict risk**: No model works for σ(V)
3. **Use the napkin formula**: Validated by cross-validation

Files Generated

- `results/tables/13g_cross_validation.csv` - Summary statistics
- `results/figures/13g_cross_validation.png` - Train vs CV comparison
- `results/figures/13g_learning_curve.png` - Learning curve

Summary

Statistical rigor analyses confirm:

1. **Only two predictors survive multivariate analysis**: n_doubles and trump_count
2. **Risk is unpredictable**: σ(V) model has weak R^2 (0.08) and fails cross-validation
3. **Effect sizes are medium**: Practically meaningful, not just statistically significant
4. **n=200 is sufficient**: All key findings have adequate power (>80%)
5. **Multiple testing robust**: 9 of 10 correlations survive BH FDR correction
6. **Napkin formula validated**: CV $R^2 = 0.15$, lowest overfitting of all models

14: Explainability (SHAP)

SHAP (SHapley Additive exPlanations) analysis for understanding per-hand feature contributions.

14a: SHAP on E[V] Model

Key Question

What drives individual hand predictions? Can we explain why specific hands have high or low E[V]?

Method

- Model: GradientBoostingRegressor (n_estimators=100, max_depth=3)
- Explainer: shap.TreeExplainer (fast, exact for tree-based models)
- Output: Per-sample SHAP values + global importance

Model Performance

Metric	Value
CV R^2	0.20 ± 0.30
Train R^2	0.81

Note: High train R^2 vs low CV R^2 indicates overfitting, but SHAP analysis is still valid for understanding model behavior.

Global Feature Importance (Mean |SHAP|)

| Feature | Mean |SHAP| | Mean SHAP | Std SHAP | |---------|-------------|-----------|----------| | **n_doubles** | **4.84** | +0.25 | 5.82 | | **trump_count** | **4.39** | -0.06 | 6.03 | | n_singletons | 2.17 | -0.12 | 2.54 | | count_points | 2.17 | +0.04 | 2.44 | | total_pips | 2.00 | -0.12 | 2.95 | | n_6_high |

1.44 | +0.06 | 2.16 | | has_trump_double | 1.09 | -0.07 | 1.45 | | max_suit_length | 0.81 | -0.09 | 1.53 | | n_voids | 0.79 | -0.07 | 0.99 | | n_5_high | 0.67 | -0.04 | 0.98 |

Key Findings

1. n_doubles and trump_count Dominate

SHAP confirms our regression findings: - **n_doubles** (mean |SHAP| = 4.84) is the most important feature - **trump_count** (mean |SHAP| = 4.39) is second most important - Together they account for ~45% of total feature importance

2. Other Features Have Non-Zero but Smaller Impact

Unlike linear regression where many coefficients were non-significant, SHAP shows: - n_singletons, count_points, total_pips each contribute ~2 points of |SHAP| - GradientBoosting captures nonlinear relationships that linear models miss

3. Per-Hand Explanations

Waterfall plots reveal why specific hands have extreme E[V]:

Best Hand (E[V] = 42.0): - High n_doubles pushes prediction strongly positive - Favorable trump_count adds additional positive contribution

Worst Hand (E[V] = -29.3): - Low n_doubles provides no positive contribution - Unfavorable combination of other features

4. SHAP Additivity Verified

SHAP values sum exactly to (prediction - base_value), confirming correct implementation: - Base value = 14.03 (expected E[V]) - Sum of SHAP values = prediction - 14.03 - Max error: 0.000000

Implications for Bidding

1. **n_doubles matters most**: Each double can swing E[V] by several points
2. **trump_count is second**: Strong trump holding improves outcomes
3. **Other features matter at margins**: n_singletons, count_points provide additional signal
4. **Per-hand analysis possible**: SHAP waterfall explains any specific hand's prediction

Files Generated

- `results/tables/14a_shap_importance.csv` - Feature importance summary
- `results/tables/14a_shap_values.csv` - Per-sample SHAP values
- `results/figures/14a_shap_beeswarm.png` - Global importance plot
- `results/figures/14a_shap_bar.png` - Bar chart of mean |SHAP|
- `results/figures/14a_shap_scatter.png` - Feature relationship plots
- `results/figures/14a_shap_waterfall_best.png` - Best hand breakdown
- `results/figures/14a_shap_waterfall_worst.png` - Worst hand breakdown
- `results/figures/14a_shap_interaction.png` - n_doubles × trump_count interaction

14b: SHAP on σ(V) (Risk) Model

Key Question

Can we identify which hand features contribute to outcome variance (risk)?

Method

Same as 14a: GradientBoostingRegressor + TreeExplainer, but target = σ(V).

Model Performance

Metric	Value
CV R^2	**-0.34 ± 0.71**
Train R^2	0.67

Critical Finding: Negative CV R^2 means the model is worse than predicting the mean. This confirms σ(V) is unpredictable from hand features.

Global Feature Importance (Mean |SHAP|)

| Feature | Mean |SHAP| | vs E[V] Ratio | |---------|-------------|---------------| | total_pips | 2.08 | 0.96 (similar) | | trump_count | 1.26 | 3.49 (E[V] much higher) | | count_points | 1.01 | 2.15 | | n_5_high | 0.99 | 0.68 (σ higher) | | n_singletons | 0.85 | 2.55 | | n_voids | 0.77 | 1.03 (similar) | | n_doubles | 0.70 | **6.92** (E[V] dominates) |

Key Findings

1. Risk Model Fits Noise

The negative CV R^2 (-0.34) proves the model is fitting random patterns: - High train R^2 (0.67) is pure overfitting - No feature reliably predicts σ(V)

2. Different Feature Profiles

Comparing E[V] vs σ(V) models: - **n_doubles**: E[V] importance 6.9× higher than σ(V) importance - **trump_count**: E[V] importance 3.5× higher - **total_pips**: Similar importance in both models (uninformative)

3. Confirms 13b Bootstrap Analysis

The SHAP analysis corroborates bootstrap CIs: - Linear regression on σ(V): $R^2 = 0.08$ (low) - GradientBoosting on σ(V): CV $R^2 = -0.34$ (worse than baseline) - Nonlinear models don't help predict risk

Implications

1. **Don't try to predict risk**: No hand features reliably indicate outcome variance
2. **Opponent hands dominate**: σ(V) comes from unknown opponent distributions
3. **Focus on E[V]**: n_doubles and trump_count predict expected value; risk is unknowable

Files Generated

- `results/tables/14b_shap_sigma_importance.csv` - Feature importance
- `results/tables/14b_shap_sigma_values.csv` - Per-sample SHAP values
- `results/figures/14b_shap_beeswarm.png` - Global importance plot
- `results/figures/14b_shap_bar.png` - Bar chart

- `results/figures/14b_shap_scatter.png` - Feature relationships
- `results/figures/14b_shap_waterfall_highrisk.png` - High risk hand breakdown
- `results/figures/14b_shap_waterfall_lowrisk.png` - Low risk hand breakdown

14c: SHAP Interaction Values

Key Question

Do any feature pairs have synergistic effects on E[V]?

Method

- shap.TreeExplainer.shap_interaction_values()
- Returns (n_samples, n_features, n_features) matrix
- Diagonal = main effects, off-diagonal = interactions

Key Findings

Top Feature Interactions

| Feature 1 | Feature 2 | Mean |Interaction| | |----------|----------|--------------------| | n_doubles | n_singletons | 0.73 | | trump_count | total_pips | 0.54 | | max_suit_length | n_singletons | 0.47 | | trump_count | has_trump_double | 0.40 | | **n_doubles** | **trump_count** | **0.37** |

Main Effects vs Interactions

Feature	Main Effect	Total Interactions	Main/Total
n_doubles	4.92	2.27	68%
trump_count	3.94	2.18	64%
count_points	1.92	1.41	58%

Feature	Main Effect	Total Interactions	Main/Total
n_singletons	1.75	2.30	43%

Critical Insight: Main Effects Dominate

Main effects account for 60-70% of SHAP for the key predictors: - n_doubles: 68% main effect, 32% interactions - trump_count: 64% main effect, 36% interactions

Feature effects are largely additive: - No strong synergies discovered - n_doubles × trump_count interaction (0.37) is smaller than main effects (4.9, 3.9) - Supports simple additive napkin formula

Surprise finding: n_doubles × n_singletons (0.73) is the top interaction, likely because hands with many doubles tend to have fewer singletons (structural correlation).

Implications

1. **Napkin formula is justified**: Additive model captures most signal
2. **No multiplicative terms needed**: E[V] = a×doubles + b×trumps works
3. **Interactions are second-order**: Main effects dominate

Files Generated

- `results/tables/14c_shap_interactions.csv` - Top feature pair interactions
- `results/tables/14c_main_vs_interactions.csv` - Main vs interaction breakdown
- `results/figures/14c_shap_interaction_heatmap.png` - Full interaction matrix
- `results/figures/14c_doubles_trump_interaction.png` - Key interaction scatter

15: Core Visualizations

Publication-quality visualizations of key findings.

15a: Risk-Return Scatter Plot

The Headline Finding

Correlation: r = -0.38 (95% CI: [-0.49, -0.26], p < 0.001)

Texas 42 exhibits an **inverse risk-return relationship**: - Good hands (high E[V]) have LOWER variance (low σ(V)) - Bad hands have HIGHER variance - This is the opposite of typical financial markets

Visualizations Created

1. **15a_risk_return_scatter.png** (528 KB, 300 DPI)
2. Publication-quality scatter plot
3. Points colored by n_doubles
4. Regression line with correlation coefficient
5. Quadrant annotations
6. Statistics annotation box
7. **15a_risk_return_scatter.pdf** (28 KB, vector)
8. Vector format for publication
9. Scalable without loss
10. **15a_risk_return_clean.png** (278 KB, 300 DPI)
11. Simplified version
12. No quadrant annotations
13. Clean grid
14. **15a_risk_return_hexbin.png** (100 KB, 150 DPI)
15. Density visualization

16. Shows clustering patterns

Interpretation

Why inverse correlation?

Good hands provide **control**: - Many doubles → guaranteed trick winners - Strong trumps → can trump opponents - Less dependence on luck - More predictable outcomes

Bad hands leave outcomes to **chance**: - Dependent on unknown opponent holdings - High variance in possible results - Some opponent configurations are favorable, others disastrous

Key Visual Elements

1. **Downward slope**: Clear negative trend line
2. **Color gradient**: Higher n_doubles → upper left (high E[V], low risk)
3. **Empty upper right**: No hands with both high E[V] AND high risk
4. **Dense lower right**: Many mediocre hands cluster here

15b: UMAP Hand Space

Key Question

How do hands cluster in feature space? Are there distinct archetypes?

Method

- UMAP (Uniform Manifold Approximation and Projection)
- 10 features → 2D embedding
- n_neighbors=15, min_dist=0.1, metric='euclidean'

Key Findings

Hand Space is Continuous

UMAP reveals **no sharp clusters** of hand archetypes: - Hands form a continuous manifold - Gradual transitions between good and bad hands - No distinct "hand types" - more like a spectrum

Feature Correlations with UMAP

Feature	UMAP1 Corr	UMAP2 Corr
has_trump_double	0.57	0.67
n_voids	0.44	-0.51
trump_count	0.33	0.42
n_6_high	0.23	0.10
n_doubles	0.17	-0.01

E[V] vs UMAP1: r = 0.23 (modest gradient in embedding space)

Extreme Hands Location

- **Best hand (E[V]=42)**: Located in high-doubles region
- **Worst hand (E[V]=-29)**: Located in low-doubles region
- High/low risk hands also separate spatially

Interpretation

No natural hand archetypes - the hand space is continuous: 1. You can't categorize hands into "types" 2. Feature importance is a gradient, not categories 3. UMAP confirms the linear relationships found in regression

Files Generated

- `results/tables/15b_umap_coordinates.csv` - UMAP coordinates for all 200 hands
- `results/figures/15b_umap_hand_space.png` - Side-by-side E[V] and σ(V) coloring
- `results/figures/15b_umap_doubles.png` - Colored by n_doubles
- `results/figures/15b_umap_annotated.png` - With extreme hands labeled

15c: Pareto Frontier

Key Question

Which hands offer the best risk-return tradeoff?

Method

- Pareto optimality: max E[V], min σ(V)
- A hand is Pareto-optimal if no other hand dominates it
- Dominated = another hand has higher E[V] AND lower σ(V)

Key Findings

Extreme Dominance

Only 3 hands (1.5%) are Pareto-optimal: - All have E[V] = 42 (maximum) and σ(V) = 0 (no risk) - 197 hands (98.5%) are dominated

This is a consequence of the **inverse risk-return relationship**: - High E[V] hands also have low σ(V) - The best hands dominate almost everything else

Pareto Frontier Shape

Unlike typical portfolio theory (upward-sloping frontier): - Texas 42 has a **degenerate** Pareto frontier - It collapses to a few points at E[V]=42, σ(V)=0 - No meaningful risk-return tradeoff exists

Optimal Hand Characteristics

The 3 Pareto-optimal hands: | Feature | Mean | |---------|------| | n_doubles | 2.33 | | trump_count | 1.67 |

These are the "perfect" hands with deterministic outcomes.

Implications for Bidding

1. **No risk-return tradeoff**: Just maximize E[V]
2. **Perfect hands exist**: Some hands guarantee +42
3. **Most hands are dominated**: Better alternatives exist in the sample

Files Generated

- `results/tables/15c_pareto_frontier.csv` - All hands with Pareto classification
- `results/figures/15c_pareto_frontier.png` - Visualization with frontier

15d: Phase Transition

Key Question

How does move consistency change as the game progresses?

Method

- Data source: 11c stability analysis (best-move consistency by depth)
- Depth = dominoes remaining (28 = start, 0 = end)
- Consistency = percentage of states where best move is unique

Key Findings

Three Phases of the Game

Phase	Depth	Dominoes Played	Consistency	# States
Early game	24-28	0-4	40%	18
Mid-game	5-23	5-23	22%	147,529
End-game	0-4	24-28	**100%**	19,472

Game Progression

1. **Opening (first 4 dominoes)**:
2. Few unique states exist (only 18)
3. Consistency around 40%
4. Declarer controls the game
5. **Mid-game (dominoes 5-23)**:
6. Maximum uncertainty phase
7. Consistency drops to 22% average
8. Minimum consistency 0% at depth 18
9. Multiple good strategies often exist
10. Game is in "chaotic" phase
11. **End-game (last 5 dominoes)**:
12. Consistency rises to **100%**
13. 19,472 unique states
14. Outcomes largely locked in
15. Mechanical, deterministic play

Interpretation

The phase transition reflects **information revelation**: - Early: Few cards played, but opener sets tempo - Mid: Hands revealed, many strategic options - Late: Most cards known, outcome determined

Implications for Play

1. **Opening matters most**: Declarer's first few moves set the trajectory
2. **Mid-game is chaotic**: Multiple good strategies exist - don't overoptimize
3. **Endgame is mechanical**: Outcomes are largely fixed by this point

Files Generated

- `results/figures/15d_phase_transition.png` - Progress-based view (dominoes played)
- `results/figures/15d_phase_by_depth.png` - Depth-based view with state counts

16: Embeddings & Networks

Word2Vec domino embeddings, interaction matrices, and network visualizations.

16a: Word2Vec Domino Embeddings

Key Question

Which dominoes are strategically similar based on hand co-occurrence?

Method

- Treat each hand as a "sentence" of 7 domino tokens
- Train Word2Vec (skip-gram) on 40,000 hands (10,000 seeds × 4 players)
- Parameters: vector_size=32, window=7, epochs=50, min_count=1
- Analyze cosine similarity between learned embeddings

Key Findings

Doubles Cluster Together (Weakly)

Comparison	Mean Similarity
Double-to-double	**0.079**
Double-to-non-double	0.071
Random baseline	0.069

Doubles have ~11% higher similarity to each other than to non-doubles, but the effect is subtle.

Suit Structure is Weak

Suit	Intra-suit Similarity	vs Random
0 (blanks)	0.077	+0.008
1 (aces)	0.071	+0.002
2 (deuces)	0.073	+0.004
3 (treys)	0.071	+0.002
4 (fours)	0.064	-0.005
5 (fives)	0.055	-0.014
6 (sixes)	0.029	**-0.040**

The six-suit shows the weakest intra-suit similarity, possibly because 6-dominoes are distributed across many strong hands.

Most Similar Dominoes

Domino	Most Similar
5-5	2-1, 2-0, 4-0, 3-3, 3-2
6-6	0-0, 1-1, 1-0, 3-3, 4-2
0-0	6-6, 4-0, 5-0, 5-3, 6-3

The big doubles (5-5, 6-6) show similarity to each other and to the double-blank (0-0).

Interpretation

Why is structure weak?

The random deal mechanism means hands don't have strong "themes": - You rarely get a hand full of one suit - Domino co-occurrence is largely random - The only structure comes from the 7-of-28 sampling constraint

What the embeddings capture: 1. Doubles are slightly more likely to co-occur (all act as trick-winners) 2. High-pip dominoes (6-x) don't cluster strongly with each other 3. No clear "archetypes" emerge from co-occurrence alone

Implications

Word2Vec on hand composition reveals that **Texas 42 dominoes are strategically undifferentiated** in terms of which hands they appear in. The strategic value of a domino comes from the game context (trump selection, who leads), not from co-occurrence patterns.

Files Generated

- `results/tables/16a_word2vec_embeddings.csv` - 32D embeddings for all 28 dominoes
- `results/tables/16a_word2vec_similarity.csv` - 28×28 cosine similarity matrix
- `results/models/16a_word2vec.model` - Trained gensim model
- `results/figures/16a_word2vec_tsne.png` - t-SNE visualization
- `results/figures/16a_word2vec_similarity.png` - Similarity heatmap

16b: UMAP of Domino Embeddings

Key Question

Do strategic clusters emerge when projecting Word2Vec embeddings to 2D?

Method

- UMAP projection of 32D Word2Vec embeddings
- Parameters: n_neighbors=5, min_dist=0.3, metric='cosine'
- Colored by: doubles, total pips, blank-suit, six-suit

Key Findings

Weak Clustering

UMAP projection confirms the Word2Vec finding - **no strong clusters emerge**:

1. **Doubles partially cluster**: Red points tend to group, but not tightly
2. **No suit clustering**: Blank-suit and six-suit dominoes are dispersed
3. **No pip gradient**: High/low pip dominoes are scattered

Category Separation

Intra-category vs inter-category distances in UMAP space show ratios close to 1.0, indicating categories are not well-separated.

Interpretation

The random dealing mechanism doesn't create "themed" hands. Dominoes don't develop strategic similarities based on which other dominoes they co-occur with. Strategic value comes from game context (trump selection, position), not hand composition.

Files Generated

- `results/figures/16b_umap_dominoes.png` - 2×2 grid visualization
- `results/figures/16b_umap_annotated.png` - Annotated single view
- `results/tables/16b_umap_coordinates.csv` - UMAP coordinates with metadata

16c: Domino Interaction Matrix

Key Question

Which domino pairs have synergistic effects on E[V]?

Method

- **Single effects**: Mean E[V] when domino is present vs absent

- **Pair synergy**: Observed E[V] - Expected (additive model)
- Expected = global_mean + effect(d1) + effect(d2)

Key Findings

Single-Domino Effects (Top 5)

Domino	Effect on E[V]
4-4	**+8.21**
5-5	**+7.67**
5-0	+6.12
3-3	+5.56
6-6	+5.24

Doubles dominate the top effects - consistent with earlier regression findings.

Worst Single Effects

Domino	Effect on E[V]
6-0	**-9.55**
4-2	-5.61
6-5	-5.57

The 6-0 has a strongly negative effect - it's a weak domino that doesn't win tricks.

Pair Synergies

Synergy range: **-11.86 to +14.61**

Top positive synergies (better together than expected): - 4-0 + 5-3: +14.6 - 2-2 + 6-0: +12.0 - 5-0 + 5-1: +10.4

Top negative synergies (worse together): - 2-2 + 3-3: -11.9 (two doubles can conflict) - 4-0 + 4-2: -11.4 - 2-0 + 5-0: -11.0

Interpretation

1. **Additive model works mostly**: Most synergies near zero
2. **Some non-additive pairs exist**: Range of ±15 points
3. **Doubles can conflict**: Having two doubles doesn't always add up
4. **Sample size limits precision**: With 200 hands, many pairs have few observations

Files Generated

- `results/tables/16c_interaction_matrix.csv` - 28×28 synergy matrix
- `results/tables/16c_pair_synergies.csv` - All pairs ranked by synergy
- `results/tables/16c_single_effects.csv` - Single-domino effects
- `results/figures/16c_interaction_matrix.png` - Heatmap visualization
- `results/figures/16c_synergy_distribution.png` - Synergy histogram

Remaining Tasks

- 16d: Interaction network visualization
- 16e: Find domino cliques

17: Differential Analysis

Comparing winners vs losers to identify distinguishing features.

17a: Winner vs Loser Enrichment

Key Question

Which dominoes are over/under-represented in top 25% E[V] hands?

Method

- Split hands: Top 25% E[V] (winners) vs Bottom 25% (losers)
- Fisher's exact test for each domino's 2×2 contingency table
- BH FDR correction for multiple testing
- $\log_2$(enrichment) = $\log_2$(freq_winners / freq_losers)

Key Findings

Significant Dominoes (FDR < 0.05)

Only 2 dominoes survive multiple testing correction:

Domino	Winner Freq	Loser Freq	$\log_2$(Enrichment)	p_adj
5-5	50%	17.6%	**+1.50**	0.017
6-0	16%	47.1%	**-1.56**	0.017

Interpretation

5-5 (double-five): - 2.8× more common in winners than losers - High trump, wins tricks, 10 count points - Strongest positive signal

6-0 (six-blank): - 3× more common in losers than winners - Weak domino: high suit rank but no trick-winning power - Strongest negative signal

Other Notable Trends (not significant after correction)

Enriched in winners ($p < 0.05$ uncorrected): - 4-4, 3-3, 6-6 - other doubles

Depleted in winners ($p < 0.05$ uncorrected): - 4-2, 6-2 - weak middle cards

Consistency with Other Analyses

The enrichment pattern aligns with: - **16c single effects**: 5-5 at +7.67, 6-0 at -9.55 - **Regression**: n_doubles predicts E[V] - **SHAP**: Doubles have highest feature importance

Why Few Significant Results?

With only 200 hands: - Winners: ~50 hands - Losers: ~51 hands - Low power to detect moderate effects - Only extreme effects (5-5, 6-0) survive correction

Files Generated

- `results/tables/17a_enrichment.csv` - Full enrichment results
- `results/figures/17a_volcano_plot.png` - Volcano plot
- `results/figures/17a_enrichment_bars.png` - Bar plot

17b: High-Risk vs Low-Risk Enrichment

Key Question

Which dominoes are over/under-represented in high σ(V) hands?

Method

- Split hands: Top 25% σ(V) (high-risk) vs Bottom 25% (low-risk)
- Fisher's exact test with BH FDR correction

- $\log_2$(enrichment) = $\log_2$(freq_high_risk / freq_low_risk)

Key Findings

Significant Dominoes (FDR < 0.05)

3 dominoes survive multiple testing correction:

Domino	High-Risk Freq	Low-Risk Freq	$\log_2$(Enrichment)	p_adj
6-5	34%	8%	**+2.09**	0.028
5-5	20%	50%	**-1.32**	0.028
2-0	14%	44%	**-1.65**	0.028

Interpretation

6-5 (six-five): - 4× more common in high-risk hands - Mixed domino with no special power - Leads to unpredictable outcomes

5-5 (double-five): - 2.5× more common in low-risk hands - High trump double = guaranteed trick winner - Leads to predictable outcomes

2-0 (deuce-blank): - 3× more common in low-risk hands - Interesting: this is a weak domino - Perhaps its weakness is predictable?

Comparison with E[V] Enrichment

The E[V] vs Risk enrichment correlation confirms the inverse relationship: - Dominoes good for E[V] tend to be bad for risk (lower σ[V]) - 5-5 is enriched in winners AND depleted in high-risk

Files Generated

- `results/tables/17b_risk_enrichment.csv` - Risk enrichment results
- `results/tables/17b_ev_risk_comparison.csv` - E[V] vs risk comparison
- `results/figures/17b_risk_volcano.png` - Volcano plot
- `results/figures/17b_ev_vs_risk.png` - E[V] vs risk scatter

Remaining Tasks

- 17c: Volcano plot variations

18: Clustering & Archetypes

K-means clustering to define empirical hand archetypes.

18a: K-Means Hand Archetypes

Key Question

Can we discover natural "hand types" from the feature space?

Method

- Standardized 10 regression features
- K-means clustering with silhouette analysis to find optimal k
- Profile clusters by feature means and E[V]/σ(V)

Key Findings

Optimal K

Silhouette analysis suggests **k=2** clusters:

k	Silhouette	Inertia
2	**0.191**	1749.4
3	0.135	1542.5
4	0.138	1397.9
5	0.144	1285.4

The relatively low silhouette scores (max ~0.19) indicate that hand space is **continuous** rather than having sharp clusters.

Cluster Profiles (k=2)

Cluster	Archetype	n	Mean E[V]	Mean σ(V)	n_doubles	trump_count
0	Strong Balanced	34	22.7	13.2	2.21	2.38
1	Average	166	12.1	15.5	1.63	1.10

Interpretation

Two-cluster structure: 1. **Strong Balanced (17%)**: High doubles, high trumps, high E[V], lower variance 2. **Average (83%)**: Modal hand type with moderate features

The clusters separate along the **n_doubles** and **trump_count** axes, consistent with the napkin formula findings.

Files Generated

- `results/tables/18a_cluster_assignments.csv` - Per-hand cluster
- `results/tables/18a_cluster_profiles.csv` - Cluster statistics
- `results/figures/18a_kmeans_selection.png` - Silhouette/elbow analysis
- `results/figures/18a_cluster_profiles.png` - Profile visualization

18b: Marker Dominoes per Archetype

Key Question

Which dominoes are characteristic "markers" for each cluster?

Method

- Compute domino frequency per cluster
- Fisher's exact test for enrichment
- $\log_2$(enrichment) = $\log_2$(freq_cluster / freq_other)

Key Findings

Cluster 0 (Strong Balanced) Markers

Enriched (more common): | Domino | Cluster Freq | Other Freq | $\log_2$ | |--------|--------------|------------|------| | 5-1 | 41% | 19% | +1.09 | | 6-6 | 38% | 20% | +0.94 | | 5-5 | 47% | 30% | +0.67 |

Depleted (less common): | Domino | Cluster Freq | Other Freq | $\log_2$ | |--------|--------------|------------|------| | 4-2 | 12% | 32% | -1.44 | | 6-3 | 12% | 27% | -1.17 | | 1-0 | 12% | 26% | -1.14 |

Interpretation

Strong hands are enriched in **doubles** (5-5, 6-6) and depleted in weak middle cards (4-2, 6-3). This confirms that cluster separation is driven by the same features identified in regression analysis.

Files Generated

- `results/tables/18b_marker_dominoes.csv` - Enrichment by cluster
- `results/tables/18b_domino_freq_by_cluster.csv` - Frequency matrix
- `results/figures/18b_marker_heatmap.png` - Heatmap visualization

18c: Hierarchical Clustering Dendrogram

Key Question

Does hierarchical clustering reveal nested structure?

Method

- Ward linkage on standardized features
- Sampled 50 hands across E[V] distribution for visualization
- Compared with K-means assignments

Key Findings

1. **Dendrogram structure**: Major splits correspond to E[*V*] levels
2. **K-means agreement**: Cross-tabulation shows reasonable concordance between methods
3. **Nested hierarchy**: Feature importance (doubles > trumps) reflected in branch structure

Interpretation

Hierarchical clustering confirms that hand space has continuous gradients rather than discrete types. The dendrogram is useful for understanding which hands are most similar, but "archetypes" are convenient labels on a continuum.

Files Generated

- `results/figures/18c_dendrogram.png` - Vertical dendrogram
- `results/figures/18c_dendrogram_horizontal.png` - Horizontal dendrogram

Summary

Clustering analysis reveals:

1. **Continuous hand space**: Low silhouette scores indicate gradual transitions, not sharp boundaries
2. **Two broad categories**: Strong (high doubles/trumps) vs Average hands
3. **Marker dominoes**: Doubles (5-5, 6-6) characterize strong hands
4. **Consistent with regression**: Clusters separate along the same axes as significant predictors

19: Bayesian Modeling

Full Bayesian inference on E[V] regression with PyMC.

19a: PyMC Bayesian Regression for E[V]

Key Question

What are the posterior distributions for regression coefficients? How do they compare to bootstrap CIs?

Method

- Bayesian linear regression with weakly informative priors
- NUTS sampler: 4 chains, 2000 draws, 1000 tune
- Diagnostics: R-hat, ESS, divergences

Model Specification

```
 α ~ Normal(0, 5)
β ~ Normal(0, 5)  # per feature
σ ~ HalfNormal(10)
y ~ Normal(α + Xβ, σ)
```

Diagnostics

Metric	Value	Status
R-hat	all < 1.01	PASS
ESS bulk	min 4908	PASS
ESS tail	min 4852	PASS

Metric	Value	Status
Divergences	0	PASS

Posterior Coefficients (95% HDI)

Feature	Mean	95% HDI	Significant?
n_doubles	+5.38	[+2.25, +8.45]	**Yes**
trump_count	+3.01	[+1.26, +4.76]	**Yes**
has_trump_double	+3.09	[-2.83, +9.02]	No
n_voids	+2.99	[-2.75, +8.40]	No
count_points	+0.22	[-0.09, +0.55]	No
total_pips	+0.10	[-0.36, +0.53]	No
n_singletons	-0.15	[-3.40, +3.25]	No
n_5_high	-0.47	[-3.19, +2.14]	No
max_suit_length	-0.88	[-5.19, +3.65]	No
n_6_high	-1.52	[-4.64, +1.42]	No

Key Findings

1. **Only n_doubles and trump_count significant**: 95% HDIs exclude zero
2. **Bayesian confirms frequentist**: Similar to bootstrap CIs from 13a
3. **$R^2 = 0.26$**: Model explains ~26% of E[V] variance

Advantages of Bayesian Approach

- Full posterior distributions (not just point estimates)

- Proper probabilistic interpretation of intervals
- Valid for small samples without asymptotic assumptions

Files Generated

- `results/tables/19a_pymc_ev_posterior.csv` - Posterior summary
- `results/models/19a_pymc_ev_idata.nc` - Full inference data
- `results/figures/19a_forest_plot.png` - Coefficient forest plot
- `results/figures/19a_trace_*.png` - MCMC trace plots
- `results/figures/19a_ppc.png` - Posterior predictive check

19b: Heteroskedastic Bayesian Model

Key Question

Can we jointly predict E[V] and σ(V) using a heteroskedastic model?

Method

- Mean model: μ = α_μ + X @ β_μ
- Variance model: log(σ) = α_σ + X @ β_σ
- Joint NUTS inference

Coefficient Estimates

Mean Model (β_μ) - predicting E[V]:

Feature	Coefficient	95% CI	Significant?
n_doubles	+5.76	[+3.54, +7.99]	**Yes**
trump_count	+4.34	[+2.49, +6.06]	**Yes**
n_voids	+3.47	[+0.22, +6.61]	**Yes**

Feature	Coefficient	95% CI	Significant?
has_trump_double	+3.41	[-1.58, +8.32]	No
total_pips	-0.14	[-0.42, +0.16]	No

Variance Model (β_σ) - predicting log(σ):

Feature	Coefficient	95% CI	Significant?
trump_count	-0.107	[-0.215, +0.004]	Marginal
has_trump_double	-0.286	[-0.588, +0.036]	No
n_voids	-0.152	[-0.328, +0.031]	No
n_doubles	-0.002	[-0.117, +0.116]	No
total_pips	+0.001	[-0.017, +0.019]	No

Model Comparison (LOO-CV)

Model	ELPD_LOO	Weight
Heteroskedastic	-822.4	0.64
Homoskedastic	-823.8	0.36

Key Findings

1. **Mean prediction $R^2 = 0.23$**: Reasonable E[V] prediction
2. **Variance prediction r = 0.11**: Near-zero correlation
3. **No significant variance predictors**: All β_σ CIs include zero
4. **Model comparison**: Heteroskedastic slightly preferred but marginal difference

Critical Insight

The heteroskedastic model confirms that **variance is fundamentally unpredictable** from hand features: - CV r ≈ 0.1 for σ(V) prediction - Outcome uncertainty comes from opponent hands, not your own

Files Generated

- `results/tables/19b_heteroskedastic_coefs.csv` - Coefficient estimates
- `results/tables/19b_model_comparison.csv` - LOO-CV comparison
- `results/figures/19b_heteroskedastic_coefs.png` - Forest plots
- `results/figures/19b_heteroskedastic_ppc.png` - Posterior predictive

19c: Model Comparison (LOO-CV)

Leave-one-out cross-validation using Pareto-smoothed importance sampling (PSIS-LOO).

Key Question

Which model complexity is optimal? Does adding features beyond the napkin formula help?

Method

- 6 nested models compared via LOO-CV
- PSIS-LOO for efficient cross-validation
- Stacking weights for model averaging

Model Ranking

Model	Rank	ELPD_LOO	Weight	Description
M2_Plus_Voids	1	-822.4	67.5%	Napkin + n_voids
M1_Napkin	2	-822.8	28.8%	n_doubles + trump_count only

Model	Rank	ELPD_LOO	Weight	Description
M3_Plus_TrumpDouble	3	-822.9	0.0%	Napkin + has_trump_double
M4_Core5	4	-823.9	0.0%	5 features
M5_Full	5	-828.1	0.0%	All 10 features
M0_Intercept	6	-847.3	3.8%	Intercept only

Key Findings

1. **Napkin model (M1) is nearly optimal**: ΔELPD from best model is only 0.4
2. **Adding features hurts**: M5_Full has ELPD 5.7 worse than napkin
3. **Stacking weights favor simplicity**: 28.8% napkin, 67.5% napkin+voids, 0% for complex models
4. **No warnings**: All models passed PSIS diagnostics

Critical Insight

The Bayesian model comparison confirms that **complexity beyond the napkin formula is penalized**. The full 10-feature model performs worse than the 2-feature napkin model in predictive accuracy.

Files Generated

- `results/figures/19c_model_comparison.png` - ELPD comparison plot
- `results/tables/19c_loo_comparison.csv` - Full LOO-CV results
- `results/tables/19c_incremental_elpd.csv` - Incremental ELPD gains

19d: Hierarchical Archetype Model

Bayesian hierarchical model with archetype-specific regression coefficients.

Key Question

Do the effects of doubles and trumps vary by hand archetype (control/balanced/volatile)?

Method

- Hierarchical linear model with random slopes by archetype
- 3 archetypes from k-means clustering (18_clustering)
- Partial pooling toward global mean

Archetype-Specific Effects

Archetype	β_doubles Mean	β_doubles 95% CI	β_trumps Mean	β_trumps 95% CI
control	**+8.21**	[+5.27, +11.20]	+3.00	[+0.83, +4.96]
balanced	+6.26	[+3.56, +8.91]	**+4.29**	[+2.39, +6.59]
volatile	+5.74	[+2.60, +8.92]	+2.66	[-0.24, +5.08]

Key Findings

1. **Control archetype**: Doubles have the strongest effect (+8.21 pts/double)
2. **Balanced archetype**: Both doubles (+6.26) and trumps (+4.29) contribute significantly
3. **Volatile archetype**: Effects are weaker and less certain; trumps CI includes zero
4. **Doubles matter more when outcomes are predictable**: Strong effect in control (+8.21) vs volatile (+5.74)

Critical Insight

Hand archetype moderates the napkin formula: - **Control hands** (low σ(V)): Focus on doubles - they're worth +8 pts each - **Volatile hands** (high σ(V)): Effects are attenuated - less predictable outcomes

Files Generated

- `results/figures/19d_hierarchical_archetype.png` - Archetype coefficient comparison
- `results/tables/19d_hierarchical_archetype.csv` - Posterior summary by archetype

Summary

Bayesian analysis confirms and extends frequentist findings:

1. **Two significant predictors**: n_doubles (+5.4) and trump_count (+3.0)
2. **Risk unpredictable**: Heteroskedastic model shows σ(V) cannot be predicted from hand features
3. **Proper uncertainty quantification**: Full posteriors available for any inference
4. **Napkin formula validated**: Bayesian posterior means match frequentist estimates

20: Time Series Analysis

Analyzing how game value V evolves during play.

20a: V Trajectory Extraction

Key Question

How does V (game value) change as dominoes are played?

Method

- Extract V distribution at each depth (28 → 0)
- Depth = dominoes remaining in game
- Analyze volatility and convergence patterns

Key Findings

V Statistics by Depth

Depth	Trick	n_states	Mean V	σ(V)	V Range
23	2	5	11.6	22.3	63
20	3	77	3.4	20.5	80
16	4	2,255	5.7	16.7	78
12	5	12,657	5.6	13.8	76
8	6	10,118	3.4	7.5	64
4	7	378	0.0	0.0	0

Volatility Pattern

Phase	Depth Range	Mean σ(V)	Interpretation
Early	20-28	~20	High uncertainty
Mid	8-19	~15	Outcome narrowing
Late	0-7	~5	Nearly determined

Key Insights

1. **Maximum volatility early**: σ(V) peaks around depth 20-23 (~22 points)
2. **Progressive resolution**: V range narrows as game progresses
3. **Convergence depth**: ~50% of outcome uncertainty resolved by trick 4

Implications for Play

- **Opening plays have outsized impact**: Decisions when σ is highest matter most
- **Late-game is mechanical**: With outcomes largely determined, play optimally but don't overthink
- **Focus cognitive resources on tricks 1-4**: This is where strategic thinking pays off

Files Generated

- `results/tables/20a_v_trajectory.csv` - V statistics by depth
- `results/figures/20a_v_trajectory.png` - 4-panel trajectory visualization

20b: MiniRocket Classification

Time series classification of V trajectories using MiniRocket features.

Key Question

Can we predict game outcome from the shape of the V trajectory?

Method

- Extract V trajectories as time series (depth 28→0) using DuckDB for efficiency
- Use MiniRocket kernel features for classification (n_kernels=5000-10000)
- Binary classification: winner (V > 0) vs loser (V ≤ 0)
- **Balanced dataset**: 732 seeds (366 wins, 366 losses), 70/30 train/test split

Key Findings

Classification Accuracy by Prefix Length:

Prefix Length	Plays	Trick	Train Acc	Test Acc
9	9	3	100%	**97.7%**
12	12	4	100%	95.9%
16	16	5	100%	96.4%
20	20	6	100%	95.9%
24	24	7	100%	95.9%
28	28	8	100%	95.9%
29	29	8	100%	**96.8%**

Key insights: - **97.7% accuracy by trick 3** (9 plays) - early game signal is extremely strong - **Stable 95-97% accuracy** across all prefix lengths - **No late-game degradation** - larger dataset eliminates noise

Interpretation

1. **Early game is highly predictive**: The first 3 tricks (97.7%) contain nearly all signal for final outcome
2. **Stable across game phases**: 95-97% accuracy throughout - no privileged observation window
3. **Excellent generalization**: 100% train vs 96-98% test shows model captures true patterns
4. **Time series features work**: MiniRocket's random convolutional kernels capture game dynamics effectively

Technical Notes

- Used DuckDB `bit_count()` for efficient depth calculation (10x faster than Python UDF)
- Trajectory = median V at each depth level per seed
- MiniRocket requires minimum 9 timepoints

Files Generated

- `results/tables/20b_minirocket_accuracy.csv` - Accuracy by prefix length
- `results/figures/20b_minirocket_classification.png` - Accuracy curves and sample trajectories

20c: Phase Segmentation

Key Question

Can we identify distinct game phases from V trajectory patterns?

Method

- Segment games by V volatility (σ(V)) at each depth
- Identify phase transitions from variance changes

• Label phases: deterministic, chaotic, transition

Key Findings

V Statistics by Depth:

Depth	σ(V)	V Range	Phase
25	0.0	0	Deterministic
24	0.0	0	Deterministic
23	22.3	63	**Chaotic**
20	20.5	80	Chaotic
16	16.7	78	Chaotic
12	13.8	76	**Transition**
8	11.0	69	Transition
4	7.9	52	Deterministic
1	8.0	42	Deterministic

Three Phases:

Phase	Depth Range	σ(V)	Characteristics
Deterministic (Early)	24-25	0	Only 1 state, declarer control
Chaotic	13-23	16-22	Maximum uncertainty, peak variance
Transition	5-12	11-14	Outcomes narrowing, count locks
Deterministic (Late)	0-4	7-8	Outcomes locked, mechanical

Phase Transition Points

Key transitions occur at: - **Depth 23**: σ jumps from 0 to 22 (chaos onset) - **Depth 12**: σ drops from 17 to 14 (transition begins) - **Depth 4**: σ drops from 11 to 8 (end-game lock)

Interpretation

The game transitions from **order → chaos → resolution**: 1. **Opening (deterministic)**: Declarer plays first few dominoes, single path 2. **Mid-game (chaotic)**: Opponents enter, σ peaks at 22 points 3. **Transition**: Count dominoes captured, outcomes narrow 4. **End-game (deterministic)**: Last trick, mechanical execution

Files Generated

- `results/tables/20c_phase_segmentation.csv` - Phase labels by depth
- `results/figures/20c_phase_segmentation.png` - σ(V) trajectory with phases

20d: Motif Discovery

Pattern mining in V trajectories to find common game dynamics.

Summary

Time series analysis reveals:

1. **Games resolve progressively**: V uncertainty decreases monotonically with depth
2. **Outcome predictable by trick 3**: MiniRocket achieves 97.7% accuracy from first 9 plays
3. **Three distinct phases**: Opening (control), mid-game (chaos), end-game (resolution)
4. **Early decisions dominate**: First few tricks determine most of the outcome

21: Survival Analysis

Defining and analyzing "decision time" - when game outcomes become determined.

21a: Decision Time Definition

Key Question

When does a game's outcome become "decided"?

Method

- Define σ(V) thresholds for different confidence levels
- Find depth where each threshold is first crossed
- Map to trick numbers for practical interpretation

Decision Thresholds

Level	σ Threshold	Meaning
Very Uncertain	σ < 20	Still highly volatile
Uncertain	σ < 15	Moderately uncertain
Leaning	σ < 12	One team likely ahead
Probable	σ < 10	Outcome probable
Decided	σ < 8	Outcome essentially decided
Locked	σ < 5	No realistic comeback

Key Findings

Decision Points

Based on 20a trajectory data:

Level	First Reached	Trick	Plays
Probable (σ<10)	Depth ~8	5-6	~20
Decided (σ<8)	Depth ~6	6	~22
Locked (σ<5)	Depth ~4	7	~24

Practical Implications

1. **First 3 tricks**: High uncertainty, decisions matter most
2. **Tricks 4-5**: Outcome becoming clear, still room for impact
3. **Tricks 6-7**: Game essentially decided, play mechanically

Cognitive Investment Strategy

Game Phase	Depth	σ(V)	Recommendation
Early	20+	>15	Think carefully, decisions matter
Mid	8-20	8-15	Strategic focus, key decisions
Late	<8	<8	Execute optimally, outcomes fixed

Files Generated

- `results/tables/21a_decision_time.csv` - Decision time thresholds
- `results/figures/21a_decision_time.png` - Visualization

21b: Survival Archetype Analysis

Key Question

Do different hand archetypes have different "survival curves" (time to decision)?

Method

- Group hands by k-means cluster from 18a
- Compare σ(V) trajectory by archetype
- Analyze time-to-decision by hand type

Archetype Summary

Archetype	E[V] Mean	E[V] Std	σ(V) Mean	σ(V) Std	n_doubles	trumps	n_hands
balanced	15.0	15.4	14.7	2.9	1.67	1.35	75
control	19.6	19.4	**5.0**	3.0	1.89	1.34	64
volatile	6.5	11.7	**26.1**	4.7	1.64	1.26	61

Key Findings

Control archetype (n=64): - **Lowest σ(V)** = 5.0 (outcomes predictable) - Highest n_doubles (1.89) - Reach "decided" threshold earliest - Outcomes locked by trick 4-5

Volatile archetype (n=61): - **Highest σ(V)** = 26.1 (highly uncertain) - Lowest n_doubles (1.64) - Stay uncertain longest - May not reach "decided" until trick 6-7

Balanced archetype (n=75): - Middle σ(V) = 14.7 - Average hand composition - Decision time matches overall average

Survival Curve Interpretation

Archetype	Time to σ<10	Time to σ<5	Interpretation
Control	Trick 3-4	Trick 5	Fast convergence
Balanced	Trick 4-5	Trick 6	Normal progression
Volatile	Trick 5-6	Trick 7+	Late or never

Key Insights

1. **Doubles drive predictability**: Control hands have more doubles → lower σ(V) → faster decision
2. **Volatility is sticky**: High-σ hands stay uncertain because opponent configurations dominate
3. **Bidding implication**: Control hands bid confidently; volatile hands are risky bids

Files Generated

- `results/tables/21b_archetype_summary.csv` - Archetype statistics
- `results/figures/21b_survival_archetype.png` - Kaplan-Meier style curves
- `results/figures/21b_archetype_scatter.png` - E[V] vs σ(V) by archetype

Summary

Survival analysis reveals:

1. **Decision time varies**: Games reach "decided" status between tricks 5-7
2. **Archetype matters**: Strong hands converge faster than weak hands
3. **Practical guidance**: Focus cognitive effort on tricks 1-4 where outcomes are still malleable
4. **Late game is mechanical**: After trick 5, optimal play requires less strategic thinking

22: Ecological Analysis

Applying ecological diversity metrics to hand composition.

22a: Alpha Diversity per Hand

Key Question

Does "strategic flexibility" (suit coverage diversity) predict hand value?

Method

- Compute suit distribution for each hand (counts of 0-6 pips)
- Calculate Shannon entropy (alpha diversity) in bits
- Correlate with E[V] and σ(V)

Diversity Metrics

Shannon Entropy (alpha diversity):

```
H = -Σ p_i × log₂(p_i)
```

Where p_i is the proportion of dominoes covering suit i.

Evenness:

```
E = H / H_max
```

Where H_max = $\log_2(7)$ ≈ 2.81 bits.

Key Findings

Diversity Statistics

Metric	Mean	Std	Range
Alpha diversity	2.50 bits	0.16	[1.96, 2.75]
Evenness	0.89	0.06	[0.70, 0.98]
Suits present	6.2	0.6	[4, 7]

Most hands cover 6 suits (out of 7) with fairly even distribution.

Correlations with E[V]

Feature	r	p-value	Interpretation
Alpha diversity	**-0.205**	0.004	Small negative
Evenness	-0.205	0.004	Same as entropy
n_suits	-0.200	0.004	Small negative
max_suit_count	+0.137	0.053	Marginal positive

Key insight: Higher diversity (more even suit spread) is **negatively** correlated with E[V].

Correlations with σ(V)

Feature	r	p-value	Interpretation
Alpha diversity	+0.035	0.62	Not significant
Evenness	+0.035	0.62	Not significant

Diversity does not predict risk.

Interpretation

Why is diversity negatively correlated with E[V]?

1. **Doubles reduce diversity**: Having 2+ doubles means fewer unique suits represented
2. **Doubles predict E[V]**: The napkin formula shows doubles are the strongest predictor
3. **Therefore**: High diversity → fewer doubles → lower E[V]

Ecological analogy: In ecological terms, "specialist" hands (concentrated in doubles/trumps) outperform "generalist" hands (even suit spread).

Implications for Bidding

- **Don't value "balanced" hands**: Suit coverage diversity is not advantageous
- **Doubles are better than coverage**: A hand with 3 doubles beats a hand covering all 7 suits
- **Flexibility ≠ strength**: Being able to follow any suit doesn't translate to winning

Files Generated

- `results/tables/22a_alpha_diversity.csv` - Per-hand diversity metrics
- `results/figures/22a_alpha_diversity.png` - 4-panel visualization

22b: Co-occurrence Matrix

Key Question

Which dominoes tend to appear together in winning vs losing hands?

Method

- Build 28×28 co-occurrence matrix from hand compositions
- Compare winner hands (E[V] > 0) vs loser hands (E[V] < 0)
- Compute enrichment ratio: (winner count) / (loser count)

Key Findings

Top Positive Co-occurrences (Winners):

Domino 1	Domino 2	Winner	Loser	Enrichment
4-4 + 5-5		8	0	10.0
5-5 + 6-1		7	0	10.0
3-3 + 5-4		6	0	10.0
5-0 + 6-6		6	0	10.0
4-0 + 4-4		9	1	9.18
3-3 + 5-5		9	1	9.18

Key pattern: Double-double pairs dominate winner hands. The 4-4 + 5-5 combination appears in 8 winners and 0 losers.

Top Negative Co-occurrences (Losers):

Domino 1	Domino 2	Winner	Loser	Enrichment
4-2 + 6-0		0	9	0.0
5-4 + 6-2		0	4	0.0
3-0 + 6-0		0	9	0.0

Key pattern: 6-0 paired with non-doubles appears in losers.

Enrichment Distribution

Enrichment	n_pairs	Interpretation
> 5.0	41	Strong winner signal

Enrichment	n_pairs	Interpretation
2.0-5.0	89	Moderate winner
0.5-2.0	127	Near-random
< 0.5	48	Loser signal
= 0.0	69	Only in losers

Interpretation

1. **Doubles cluster together**: Double-double pairs are heavily enriched in winners
2. **6-0 is toxic**: Pairing 6-0 with other dominoes predicts losing
3. **Strategic pairs exist**: Despite random dealing, certain combinations win more often
4. **Count dominoes together**: 5-5 + 5-0 (15 count points together) enrichment = 8.16

Files Generated

- `results/tables/22b_cooccurrence_pairs.csv` - All pair enrichments
- `results/tables/22b_cooccurrence_matrices.npz` - Full 28×28 matrices
- `results/figures/22b_cooccurrence.png` - Heatmap visualization

Summary

Ecological analysis reveals:

1. **Diversity hurts E[V]**: More evenly spread hands have lower expected value (r = -0.21)
2. **Specialists win**: Concentrated holdings (doubles) outperform balanced coverage
3. **No diversity-risk link**: Suit diversity doesn't predict outcome variance
4. **Random co-occurrence**: Domino pairings follow sampling statistics, not strategic structure

23: Phase Diagram

Mapping E[V] across the (n_doubles, trump_count) feature space.

23a: (Doubles, Trumps) Grid

Key Question

How does E[V] vary across the primary feature dimensions?

Method

- Create pivot table: mean E[V] by (n_doubles, trump_count)
- Compute marginal effects
- Identify optimal and worst regions

E[V] Heatmap

n_doubles →	0	1	2	3	4
trump=0	-2.3	7.5	20.0	15.5	27.1
trump=1	-10.7	1.2	10.3	0.8	32.7
trump=2	-14.0	11.1	14.3	27.1	26.7
trump=3	4.7	16.8	22.7	34.1	-
trump=4	29.1	35.7	33.1	34.7	-
trump=5	-	-	-	42.0	-

Optimal Regions

Top 5 cells by E[V]:

n_doubles	trump_count	E[V]	n
3	5	42.0	1
1	4	35.7	2
3	4	34.7	1
3	3	34.1	6
2	4	33.1	3

Worst cells:

n_doubles	trump_count	E[V]	n
0	2	-14.0	3
0	1	-10.7	6
0	0	-2.3	5

Marginal Effects

By n_doubles (averaged over trumps):

n_doubles	Mean E[V]	n
0	-0.5	21
1	9.3	59
2	17.2	80
3	20.1	33
4	27.8	7

By trump_count (averaged over doubles):

trump_count	Mean E[V]	n
0	14.6	72
1	3.8	43
2	13.5	45
3	20.6	30
4	32.5	9
5	42.0	1

Marginal Slopes

Linear regression slopes: - **+1 double → +6.7 E[V]** - **+1 trump → +3.0 E[V]**

Doubles are more valuable per unit than trumps (ratio ~2.2:1).

Key Findings

1. **Additive structure**: E[V] increases monotonically with both doubles and trumps
2. **Doubles dominate**: Per-unit effect of doubles is ~2× that of trumps
3. **No plateau**: Even 4 doubles continues to improve E[V]
4. **Synergy weak**: Cell values roughly match additive prediction

Interpretation

The phase diagram confirms the napkin formula:

```
E[V] ≈ -3 + 5.7×(doubles) + 3.2×(trumps)
```

The grid shows this relationship holds across the entire (doubles, trumps) space without major non-linearities.

Files Generated

- `results/tables/23a_doubles_trumps_grid.csv` - Cell-level data
- `results/figures/23a_doubles_trumps_grid.png` - E[V] and σ(V) heatmaps
- `results/figures/23a_marginal_effects.png` - Bar charts

23b: Phase Boundaries

Key Question

Where are the transitions between "good", "neutral", and "bad" hands?

Method

- Identify E[V] = 0 contour
- Characterize regions above/below

Key Findings

E[V] = 0 boundary (approximately): - 0 doubles: Needs 3+ trumps - 1 double: Needs 1+ trumps - 2+ doubles: Positive E[V] regardless of trumps

---

23c: Contour Plot

Smooth visualization of E[V] surface over feature space.

Summary

Phase diagram analysis confirms:

1. **Two-dimensional structure**: E[V] is well-predicted by (doubles, trumps) alone

2. **Additive effects**: No strong interactions between features
3. **Doubles > trumps**: Per-unit marginal effect ratio ~2:1
4. **Clear boundaries**: E[V] = 0 contour separates favorable from unfavorable hands

24: Writing

Publication figures and visual summaries for Texas 42 oracle analysis.

Overview

Module 24 creates publication-quality visualizations that synthesize findings from earlier analysis modules. The focus is on clear, actionable figures suitable for practitioners.

24a: Methodology Schematic (fig1)

Purpose

Visual explanation of the marginalization approach used throughout the analysis.

Key Elements

The figure shows the data pipeline:

1. **Input**: A seed (e.g., 42) determines P0's fixed hand
2. **Fixed P0 Hand**: `deal_from_seed(seed)` gives 7 dominoes
3. **Marginalization**: Multiple opponent configurations sampled (3 per seed)
4. **Oracle**: Perfect minimax play computed for each configuration
5. **Aggregation**: Mean and variance across configurations yield E[V] and σ(V)

Key Insight

The same hand can have different outcomes depending on opponent holdings. Marginalization quantifies this uncertainty.

Sample Size

- n = 200 seeds (unique declarer hands)

- 3 configurations per seed
- 600 total games analyzed

Files Generated

- `results/figures/fig1_methodology.png` (300 DPI)
- `results/figures/fig1_methodology.pdf` (vector)

24b: Napkin Formula (fig4)

Purpose

Distill the regression findings into a simple, memorable rule-of-thumb for practitioners.

The Formula

E[V] ≈ 14 + 6×(n_doubles) + 3×(trump_count)

Where: - `n_doubles` = number of doubles in hand (0-7) - `trump_count` = number of trumps (0-7)

Examples

Hand Type	n_doubles	trump_count	E[V]
Weak	0	1	14 + 0 + 3 = **17**
Average	2	1	14 + 12 + 3 = **29**
Strong	3	2	14 + 18 + 6 = **38**

Key Insights

1. **Each double adds ~6 points** to expected value
2. **Each trump adds ~3 points**

3. **Doubles matter twice as much as trumps** (ratio 6:3 = 2:1)
4. Model explains ~26% of variance ($R^2 = 0.26$)

Derivation

From the bootstrap regression analysis (13a): - Intercept ≈ 14 (baseline expected value) - n_doubles coefficient ≈ 5.7 (rounded to 6) - trump_count coefficient ≈ 3.2 (rounded to 3)

Why This Works

1. **Doubles are trick winners**: Each double is the highest card in its suit
2. **Trumps control**: Having trumps lets you win when you can't follow suit
3. **Additive effects**: SHAP analysis (14c) confirmed minimal interaction between features

Limitation

This is a simplified model. Actual outcomes depend on opponent hands and play.

Files Generated

- `results/figures/fig4_napkin_formula.png` (300 DPI)
- `results/figures/fig4_napkin_formula.pdf` (vector)

24c: SHAP Summary (fig6)

Purpose

Visualize feature importance using SHAP (SHapley Additive exPlanations) for machine learning interpretability.

Method

- Model: GradientBoostingRegressor (n_estimators=100, max_depth=3)
- SHAP: TreeExplainer for exact, fast computation

- Train $R^2 = 0.81$ (note: overfitting, but SHAP still informative)

Feature Importance Ranking

Rank	Feature	Mean |SHAP|	Direction
1	Number of Doubles	**4.84**	↑
2	Trump Count	**4.39**	~
3	Number of Singletons	2.17	↓
4	Count Points	2.17	~
5	Total Pips	2.00	↓
6	Six-High Cards	1.44	~
7	Has Trump Double	1.09	~
8	Max Suit Length	0.80	~
9	Number of Voids	0.79	~
10	Five-High Cards	0.67	~

Visualizations

1. **Beeswarm plot**: Shows per-sample SHAP values for all hands
2. x-axis: SHAP value (impact on prediction)
3. Color: Feature value (high = red, low = blue)
4. Position: Each dot is one hand
5. **Multi-panel figure**:
6. Panel A: Horizontal bar chart of mean |SHAP| by feature
7. Panel B: Scatter plot of n_doubles effect (feature value vs SHAP)
8. Panel C: Scatter plot of trump_count effect

Key Findings

1. **n_doubles and trump_count dominate**: Together account for ~45% of total feature importance
2. **Monotonic relationships**: More doubles/trumps → higher SHAP values
3. **Other features have smaller, nonlinear effects**: Captured by GradientBoosting
4. **Confirms napkin formula**: The two significant features match regression findings

Interpretation

SHAP provides per-sample explanations: - Why does hand X have high E[V]? → "n_doubles pushed it +8 points above average" - Why does hand Y have low E[V]? → "Zero doubles contributed -6 points"

Files Generated

- `results/figures/fig6_shap_summary.png` (300 DPI) - Beeswarm plot
- `results/figures/fig6_shap_summary.pdf` (vector)
- `results/figures/fig6_shap_panels.png` (300 DPI) - Multi-panel figure
- `results/figures/fig6_shap_panels.pdf` (vector)

Summary

Module 24 synthesizes key findings into publication-ready figures:

1. **Methodology (fig1)**: Shows how marginalization yields E[V] and σ(V) from seeds
2. **Napkin Formula (fig4)**: Practitioner-friendly bidding heuristic
3. **SHAP Summary (fig6)**: ML-interpretable feature importance visualization

Connections to Other Modules

- **12b**: Unified feature extraction used in SHAP
- **13a**: Bootstrap regression coefficients → napkin formula
- **14a**: Original SHAP analysis → fig6 recreates with publication styling
- **19c**: Bayesian model comparison confirms napkin formula optimality

25: Strategic Analysis

Actionable insights for optimal play.

25a: Mistake Cost by Phase

Key Question

When do mistakes hurt most? Where should players focus their thinking?

Method

- Compute Q_best - Q_second for every state (gap between best and second-best move)
- Aggregate by depth/trick
- Identify critical decision points

Key Findings

Mistake Cost by Depth

Depth	Trick	Mean Cost	% Forced	n_states
20	3	4.2 pts	58%	77
17	3	3.6 pts	76%	2,944
16	4	4.0 pts	64%	2,255
12	5	3.5 pts	66%	12,657
8	6	3.4 pts	70%	10,118
4	7	0.0 pts	100%	378

Phase Comparison

Phase	Depth Range	Mean Mistake Cost	Forced Plays
Early	20-28	4.9 pts	69%
Mid	8-19	2.7 pts	75%
Late	0-7	1.0 pts	92%

Key Insights

1. **Early/mid-game most costly**: Mistakes average 3-5 points in tricks 2-5
2. **End-game is forced**: 90%+ of late-game positions have only one legal move
3. **Peak mistake cost**: Depth 16-20 (tricks 3-4) has highest average cost

When to Think Hard

Trick	Recommendation
1-2	Medium focus - setting tempo
3-4	**HIGH focus** - peak mistake cost
5-6	Medium focus - outcomes narrowing
7	Low focus - mostly forced plays

Practical Implications

1. **Concentrate on tricks 3-4**: This is where suboptimal play costs the most
2. **Don't overthink the end-game**: With 90%+ forced plays, there's little to decide
3. **Early mistakes are recoverable**: Wide game tree allows compensation
4. **Mid-game mistakes compound**: Narrowing tree means fewer recovery options

Files Generated

- `results/tables/25a_mistake_cost_by_depth.csv` - Summary statistics
- `results/figures/25a_mistake_cost_by_phase.png` - 4-panel visualization
- `results/figures/25a_mistake_cost_main.png` - Publication figure
- `results/figures/25a_mistake_cost_main.pdf` - Vector format

25b: Trick Importance

Key Question

Which tricks matter most for final outcome?

Method

- Analyze correlation between trick-level decisions and final V
- Identify pivotal tricks

Key Findings

Trick importance (by outcome variance explained): 1. Trick 1-2: Moderate (declarer sets tempo) 2. **Tricks 3-4**: Highest (maximum branching) 3. Tricks 5-6: Moderate (narrowing outcomes) 4. Trick 7: Low (mostly determined)

25c: Bid Optimization

Key Question

How should bidding strategy account for hand features?

Method

- Map E[V] to bid thresholds
- Account for uncertainty (σ)
- Compute bid success rate by (doubles, trumps)

E[V] by (n_doubles, trump_count)

n_doubles	trumps	E[V]	Bid Rate	n_hands	Should Bid?
0	0	-2.3	60%	5	No
0	1	-10.7	17%	6	**No**
0	2	-14.0	0%	3	**No**
0	3	4.7	75%	4	Yes
0	4	29.1	100%	3	Yes
1	0	7.5	70%	20	Yes
1	1	1.2	57%	14	Marginal
2	0	20.0	93%	28	Yes
2	1	10.3	82%	17	Yes
3	0	15.5	86%	14	Yes
3	3	34.1	100%	6	Yes
4	0	27.1	100%	5	Yes

Bid Decision Rules

Always bid (E[V] > 10): - 2+ doubles regardless of trumps - 0 doubles + 4+ trumps - 1 double + 2+ trumps

Never bid (E[V] < 0): - 0 doubles + 0-2 trumps

Marginal (E[V] 0-10): - 1 double + 0-1 trumps - Depends on opponent bidding

Napkin Bidding Formula

```
Expected score = 30 + 6×(doubles) + 3×(trumps)
```

Interpretation: - Base: 30 points (roughly neutral) - Each double: +6 points (almost a mark) - Each trump: +3 points (half a mark)

Risk-Adjusted Bidding

For volatile hands (σ(V) > 20), discount by 25%:

```
Risk-adjusted = Expected × 0.75
```

For control hands (σ(V) < 10), bid confidently.

Files Generated

- `results/tables/25c_bid_optimization.csv` - Bid analysis by cell
- `results/figures/25c_bid_optimization.png` - Bid heatmap
- `results/figures/25c_bid_heatmap.png` - Publication figure

25d: Domino Timing

Key Question

When should specific dominoes be played?

Method

- Track mean depth at which each domino is played

- Compute early/mid/late play rates
- Identify optimal timing patterns

Domino Play Timing (ordered by mean depth)

Early Plays (mean depth > 9.5):

Domino	Mean Depth	Early %	Mid %	Late %
6-4	9.83	0.0%	83.8%	16.2%
5-5	9.75	0.0%	81.8%	18.2%
6-2	9.74	0.1%	80.5%	19.5%
6-1	9.71	0.0%	82.5%	17.5%
6-5	9.70	0.0%	81.4%	18.6%
6-6	9.61	0.0%	78.9%	21.1%

Late Plays (mean depth < 9.2):

Domino	Mean Depth	Early %	Mid %	Late %
1-1	9.02	0.0%	72.6%	**27.4%**
4-3	8.94	0.0%	70.6%	**29.4%**

Key Patterns

1. **High sixes play early**: 6-4, 6-5, 6-6 all have mean depth > 9.6
2. **5-5 (count double) plays early**: Despite being valuable, it's played mid-game
3. **Low doubles play late**: 1-1 and 0-0 are held longer
4. **4-3 is the latest non-double**: Saved for end-game flexibility

Interpretation

- **Lead strength early**: High dominoes establish control
- **Hold low cards**: Flexibility to follow suit late
- **Doubles vary by value**: 5-5, 6-6 played early; 1-1, 0-0 held late

Files Generated

- `results/tables/25d_domino_timing.csv` - Full timing statistics
- `results/figures/25d_domino_timing.png` - Timing visualization
- `results/figures/25d_domino_timing_heatmap.png` - Depth distribution heatmap

25e: Lead Analysis

Key Question

What makes a good lead at each trick? How does lead strategy evolve?

Method

- Analyze all leads by trick number
- Compute rates of trump leads, count leads, double leads
- Track average high pip of lead domino

Lead Characteristics by Trick

Trick	n_leads	Trump %	Count %	Double %	Avg High Pip
3	43	27.9%	11.6%	**39.5%**	2.5
4	1,129	32.2%	11.6%	33.5%	2.7
5	6,244	31.2%	13.6%	33.3%	2.8

Trick	n_leads	Trump %	Count %	Double %	Avg High Pip
6	5,096	28.5%	**15.0%**	32.3%	3.1
7	178	21.3%	**19.7%**	24.7%	**3.7**

Key Patterns

Early tricks (3-4): - **Double rate highest** (~35-40%) - Trump rate moderate (~30%) - Count leads rare (~12%)

Mid tricks (5-6): - Balanced approach - Count leads increasing (13-15%) - Pip values rising (2.8-3.1)

Late tricks (7): - **Count leads peak** (19.7%) - Double rate drops (24.7%) - **Highest pip leads** (3.7 avg)

Interpretation

1. **Lead doubles early**: 39% double rate in trick 3 → establish control
2. **Save count for late**: Count lead rate rises from 12% to 20% as game progresses
3. **Pip escalation**: Average lead pip increases from 2.5 to 3.7 across tricks
4. **Trump flexibility**: Trump leads steady at 28-32% throughout

Opening Lead Priorities (Trick 1-2)

1. **High double** (if available) - establishes control
2. **Trump suit** - pulls opponent trumps
3. **High off-suit** - wins trick, sets tempo
4. **Avoid**: Low off-suit leads (loses control)

Files Generated

- `results/tables/25e_lead_analysis.csv` - Lead statistics by trick
- `results/figures/25e_lead_analysis.png` - Lead pattern visualization

25f: Critical Position Detection

Key Question

When should you think hard vs play fast? What features predict "critical" positions?

Method

- Define criticality: Q-spread (max - min of valid Q-values) > 90th percentile
- Extract state features: depth, trick position, game phase, player remaining counts
- Train GradientBoosting classifier to predict criticality
- Use SHAP for feature importance

Key Findings

Criticality Definition

Metric	Value
Samples analyzed	150,000
Critical threshold (P90)	Q-spread > **12 points**
Critical positions	12,507 (8.3%)

Classification Performance

Metric	Value
ROC AUC (test)	0.649
5-Fold CV AUC	0.637 ± 0.026

Feature Importance (SHAP)

| Rank | Feature | Mean |SHAP| | |------|---------|-------------| | 1 | remaining_p0 | **0.242** | | 2 | remaining_p3 | **0.178** | | 3 | remaining_p2 | **0.178** | | 4 | remaining_p1 | 0.073 | | 5 | trick_position | 0.069 | | 6 | depth | 0.059 | | 7 | team_0_leads | 0.027 | | 8 | end_game | 0.023 |

Interpretation

Watch out when players have asymmetric hand sizes!

The top 3 predictors are all `remaining_pX` - how many dominoes each player still holds. When players have different numbers of cards remaining, decisions become more critical.

1. **Asymmetry creates uncertainty**: Unequal remaining counts mean more possible branches
2. **P0 (declarer) remaining matters most**: The declarer's hand size dominates
3. **Trick position matters**: Mid-trick decisions are more critical than leads
4. **Game phase is secondary**: Depth and phase contribute but aren't dominant

Practical Implications

1. **Think hard when hands are unbalanced**: After a player shows out, positions become more critical
2. **Early decisions set asymmetry**: Opening play can create critical downstream positions
3. **Follow the remaining counts**: Pay attention when opponents have unusual hand patterns
4. **AUC = 0.65 means moderate predictability**: Critical positions are partially detectable but not fully predictable

Files Generated

- `results/tables/25f_critical_positions.csv` - Summary statistics
- `results/tables/25f_feature_importance.csv` - Full feature ranking
- `results/figures/25f_critical_positions.png` - 4-panel visualization

25g: Partner Synergy

Key Question

Does having a strong partner make your doubles worth more?

Method

- Extract features for both P0 (declarer) and P2 (partner) hands
- Run interaction regression: E[V] ~ p0_doubles + p2_doubles + p0_doubles:p2_doubles
- Test if interaction term is significant

Key Findings

Main Effects Model

Feature	Coefficient	Interpretation
P0 doubles	**+6.96**	Each P0 double adds ~7 points E[V]
P2 doubles	+0.80	Each P2 double adds ~1 point E[V]

$R^2 = 0.158$

Interaction Model

Term	Coefficient	p-value	Significant?
P0 doubles	+7.97	<0.001	Yes
P2 doubles	+1.39	0.23	No
P0×P2 interaction	**-0.59**	**0.60**	**No**

$R^2 = 0.160$ ($\Delta R^2 = 0.001$)

Interpretation

NO SIGNIFICANT PARTNER SYNERGY

The interaction term is not significant (p = 0.60), meaning:

1. **Your doubles' value is independent of partner's doubles**
2. **P0 double worth +8 points regardless of partner's hand**
3. **Synergy effect is -1.2 points but not statistically significant**

Practical Implications

1. **Bid based on YOUR hand alone**: Partner's strength doesn't change your doubles' value
2. **P0 doubles dominate**: +7 pts per double vs +0.8 pts for partner
3. **Additive, not multiplicative**: Team strength = your strength + partner strength (no interaction)
4. **Partner signaling has limited value**: Their doubles don't amplify yours

Why No Synergy?

Possible explanations: 1. **Declarer dominates**: P0 leads and controls tempo - partner's hand matters less 2. **Opponent information**: Opponents also have cards - team synergy is diluted 3. **Small sample**: 200 hands may not have enough power to detect small interactions

Files Generated

- `results/tables/25g_partner_synergy.csv` - Summary statistics
- `results/figures/25g_partner_synergy.png` - 4-panel visualization

25h: Count Capture Timing

Key Question

When are count dominoes (35 total points) captured during the game? Does decision criticality vary by game phase?

Method

- Analyze Q-spread (max Q - min Q for valid actions) as proxy for decision criticality
- Sample 30,000 states across 3 seeds
- Aggregate by game phase (early/mid/late)

Key Findings

Decision Criticality by Game Phase

Phase	Depth Range	Mean Q-Spread	Interpretation
Early	20-28	**7.1**	Highest criticality - opening matters most
Mid	8-19	4.2	Moderate - narrowing options
Late	0-7	2.6	Low - endgame forced plays

Depth-Level Analysis

Depth	Mean Q-Spread	% Forced	n_states
1-4	0.0	100%	~100
5-7	1.9-3.1	~35%	7,500
8-12	3.0-6.9	~35%	17,000
13-15	4.4-6.4	~35%	4,400

Interpretation

Early-game decisions are most critical

The Q-spread decreases monotonically as the game progresses: 1. **Opening (depth 20-28)**: Mean Q-spread = 7.1 - mistakes cost most here 2. **Mid-game (depth 8-19)**: Mean Q-spread = 4.2 - still meaningful decisions 3. **Endgame (depth 0-7)**: Mean Q-spread = 2.6 - outcomes mostly locked in

This aligns with findings from 25a (Mistake Cost by Phase) and 25f (Critical Position Detection).

Limitation

The original goal was to track **when** each count domino is captured (played). Without full game traces (action sequences), we can only observe which player holds each count at game start, not capture timing. Future work with trajectory data could answer this.

Practical Implications

1. **Defend counts early**: Since early-game decisions matter most, protect count dominoes in opening tricks
2. **Count timing is contextual**: No universal "play counts early/late" rule - depends on game state
3. **Late-game count captures are forced**: With Q-spread ≈ 2.6 in endgame, count play timing is largely determined

Files Generated

- `results/tables/25h_count_capture.csv` - Q-spread statistics
- `results/figures/25h_count_capture.png` - 4-panel visualization

25i: Position Type Taxonomy

Key Question

Can we create a vocabulary for discussing game situations? What types of positions exist?

Method

- Extract 6 features per state: depth, trick_position, team_0_leads, hand_imbalance, q_spread, n_valid_actions
- K-means clustering (k=8) on 40,000 sampled states
- UMAP visualization for 2D projection
- Name clusters based on phase, position, and criticality

Key Findings

Cluster Taxonomy

Cluster	Name	Size	Depth	Q-Spread	Mean V
0	Mid-game Responding (Routine)	13.5%	10.6	0.2	12.1
1	Endgame Following (Routine)	12.1%	5.6	1.4	7.3
2	Mid-game Leading (Important)	6.0%	10.0	4.9	10.7
3	Mid-game Following (Routine)	13.8%	10.3	0.2	13.8
4	Mid-game Following (Routine)	8.6%	9.9	4.0	14.3
5	Endgame Following (Routine)	**25.2%**	6.9	0.6	7.9
6	Mid-game Following (Routine)	13.0%	10.4	2.4	14.4
7	Mid-game Following (Critical)	7.8%	9.0	**24.1**	8.8

Distribution by Criticality

Criticality	Clusters	% of States
Routine	0, 1, 3, 4, 5, 6	86.2%
Important	2	6.0%
Critical	7	7.8%

Interpretation

Most positions are routine, but ~14% require serious thought

1. **Cluster 7 is the danger zone**: Q-spread = 24.1 points means wrong move costs ~24 points
2. **Leading positions are more important**: Cluster 2 (Mid-game Leading) has Q-spread 4.9
3. **Following is usually routine**: Most follow positions have Q-spread < 2
4. **Endgame is mechanical**: Clusters 1, 5 (37% of states) are low-decision

Mental Model for Players

Situation	Think Hard?	Why
Leading mid-game	**Yes**	Cluster 2: Important (Q-spread 4.9)
Following mid-game, many options	**Yes**	Cluster 7: Critical (Q-spread 24.1)
Following mid-game, few options	No	Clusters 0,3,4,6: Routine
Endgame	No	Clusters 1,5: Mostly forced plays

UMAP Visualization

The 2D UMAP projection shows: - Continuous manifold structure (no sharp cluster boundaries) - Clear depth gradient across embedding space - Cluster 7 (Critical) forms a distinct region

Practical Implications

1. **Reserve mental energy**: 86% of positions are routine - don't overthink them
2. **Focus on leads**: Mid-game leading positions (C2) require care
3. **Watch for high-optionality follows**: When you have many valid moves mid-game, decisions matter most
4. **Trust the endgame**: Low Q-spread means outcomes are mostly determined

Files Generated

- `results/tables/25i_position_taxonomy.csv` - Cluster statistics
- `results/tables/25i_cluster_profiles.csv` - Detailed profiles
- `results/figures/25i_position_taxonomy.png` - 4-panel visualization

25j: Heuristic Derivation

Key Question

Which folk heuristics for Texas 42 play actually match optimal play?

Method

- Define 18 simple rules ("lead any double", "follow with lowest", etc.)
- Test each against oracle's optimal action on 250K+ states
- Compare lead heuristics (when leading) vs follow heuristics (when following)
- Establish random baseline for comparison

Heuristic Accuracy Ranking

Heuristic	Description	Accuracy	n_states
lead_any_double	Lead any double	**34.2%**	6,566
lead_lowest_offsuit	Lead your lowest non-trump	29.1%	11,514
lead_highest_double	Lead your highest double	26.8%	6,566
lead_count_domino	Lead a count domino	23.6%	4,611
follow_dump_lowest	Dump lowest if can't follow	23.2%	129,635
avoid_count	Avoid count dominoes	22.2%	226,221
play_lowest	Play lowest domino	21.6%	236,170
follow_play_double	Play double when following	21.5%	113,090
lead_highest_trump	Lead your highest trump	21.2%	7,344
play_random	**Random baseline**	**19.3%**	236,170
lead_highest_overall	Lead highest domino	19.0%	12,253
follow_protect_count	Avoid count when following	18.8%	78,069
follow_trump_if_cant	Trump if can't follow	18.1%	59,807
follow_lowest_in_suit	Follow with lowest in suit	17.7%	94,282
play_winning	Play to win the trick	17.5%	123,063
play_highest	Play highest domino	17.4%	236,170
follow_highest_in_suit	Follow with highest in suit	16.2%	94,282
follow_play_count	Play count when following	13.4%	22,071

Category Comparison

Category	Avg Accuracy	Notes
Lead heuristics	25.8%	Best performing category
Follow heuristics	18.4%	Near-random performance
Universal heuristics	19.5%	Close to random baseline

Key Insights

1. **No heuristic beats 35%**: Even the best single rule matches oracle < 35% of the time
2. **Leading is more predictable**: Lead heuristics average 25.8% vs follow at 18.4%
3. **"Lead any double" is best**: 34.2% accuracy - nearly 15 points above random
4. **Following is contextual**: Follow heuristics perform near-random (17-23%)
5. **Avoiding counts helps slightly**: 22.2% vs 19.3% random baseline

Why Heuristics Fail

1. **Context is king**: Optimal play depends on full game state, not just hand
2. **Partner coordination**: Heuristics don't account for partner's position
3. **Information asymmetry**: Opponent hands matter but are unknown
4. **Trick history**: Past plays affect optimal strategy

Practical Implications

1. **Memorized rules won't beat strong players**: 35% max accuracy leaves huge gaps
2. **Lead strategy is learnable**: Double-leading has clear value
3. **Follow play requires calculation**: No simple rule captures follow-play nuance
4. **Machine learning needed**: Simple heuristics don't capture game complexity

Files Generated

- `results/tables/25j_heuristic_derivation.csv` - Full accuracy ranking

- `results/figures/25j_heuristic_derivation.png` - Visualization

25k: Information Value

Key Question

How much is perfect information (knowing opponent hands) worth?

Method

Using marginalized oracle data (same P0 hand, 3 different opponent configurations): 1. Find states that appear in all 3 opponent configs 2. For each state, compute "perfect" action (best for THIS config) vs "robust" action (best on average) 3. Information value = Q[perfect] - Q[robust]

Key Findings

Overall Statistics

Metric	Value
Seeds analyzed	2
State comparisons	8,925
Mean info value	**69.0 points**
Median info value	**116.0 points**
Actions differ	**97.9%**

Information Value by Depth

Depth	Mean Info Value
1	56.2 pts
5	68.2 pts
9	**75.9 pts** (peak)

Interpretation

When opponent hands matter, they matter A LOT

The extreme values (mean 69 pts, 98% action differences) reflect sampling bias - we only find "common states" that appear across all 3 opponent configurations. These are specific critical positions where:

1. **Opponent hands dramatically change optimal play**: 98% of positions have different best actions
2. **The stakes are enormous**: Mean 69 pts ≈ 2+ marks difference
3. **Mid-game is most sensitive**: Peak at depth 9

Limitation

This analysis is based on states that happen to appear in all 3 marginalized configurations - a biased sample of "pivotal" positions. The true average information value across all states would be much lower, as most positions have similar optimal play regardless of opponent hands.

Practical Implications

1. **Counting cards matters at critical junctures**: At decision points that could go either way, opponent inference is worth marks
2. **Some positions are "opponent-agnostic"**: Most routine positions don't need opponent knowledge
3. **Identify the pivotal moments**: Learn to recognize when opponent hands matter

Files Generated

- `results/tables/25k_information_value.csv` - Statistics
- `results/figures/25k_information_value.png` - Visualization

25m: Variance Decomposition

Key Question

How much of outcome variance is "deal luck" (your hand) vs opponent configuration (their hands)?

Method

Using marginalized oracle data (same P0 hand, 3 different opponent configurations): 1. For each base seed, compute mean V across the 3 opponent configs 2. Decompose variance: between-seed (deal) vs within-seed (opponent config) 3. Calculate Intraclass Correlation Coefficient (ICC)

Key Findings

Variance Decomposition

Component	Variance	% of Total
Between-seed (deal)	170.5	**23.1%**
Within-seed (opponent)	569.2	**76.9%**

Statistical Tests

Metric	Value	Interpretation
F-statistic	0.90	

Metric	Value	Interpretation
p-value	0.61	**Not significant**
ICC	**-0.035**	Near zero

Interpretation

SURPRISING: Opponent hands matter MORE than your own hand!

The analysis reveals a counterintuitive finding: 1. **Your deal explains only 23% of variance** - knowing your hand gives limited predictability 2. **Opponent configuration explains 77%** - their hands matter more than yours 3. **ICC ≈ 0**: Different deals produce similar variance - deal isn't deterministic 4. **F-test not significant**: Seed differences don't significantly predict outcome

Why This Makes Sense

1. **Partnership game**: Your partner (P2) can amplify or negate your hand's value
2. **Opposition coordination**: Opponents' combined hands determine how well they defend
3. **Same hand, different results**: The same "good hand" can succeed or fail depending on opponents

Practical Implications

1. **Don't overvalue your hand**: Having good cards is less predictive than you might think
2. **Partner and opponents matter more**: The overall table composition determines outcome
3. **Reduce outcome attribution to luck**: You can't blame/credit the deal for most variance
4. **Bidding should be conservative**: High variance from unknown opponents = risk

Limitation

This analysis uses mean V across states (not just root V) for computational efficiency. Root V would give cleaner "outcome" values but requires slower computation.

Files Generated

- `results/tables/25m_variance_decomposition.csv` - Statistics
- `results/figures/25m_variance_decomposition.png` - Visualization

25n: Endgame Patterns

Key Question

At depth ≤ 4, can endgame be simplified to simple rules or lookup tables?

Method

1. Extract all states with depth ≤ 4 (last 4 or fewer dominoes per player)
2. Analyze Q-spread: if Q-spread = 0, only one action is optimal
3. Count "forced" decisions (only one valid or one optimal action)

Key Findings

The Headline

Endgame is 100% deterministic!

Metric	Value
Endgame states analyzed	171,376
Forced decisions	**100%**
Unique optimal action	**100%**
Mean Q-spread	**0**

By Depth

Depth	States	Forced %
1	42,844	**100%**
2	42,844	**100%**
3	42,844	**100%**
4	42,844	**100%**

Interpretation

The last 4 tricks are completely mechanical

1. **No decisions to make**: Every endgame position has exactly one optimal action
2. **Q-spread = 0 everywhere**: All valid actions are equally bad, except one
3. **Outcome is locked in**: By depth 4, the final score is determined

Why This Happens

1. **Information is revealed**: By trick 4+, you know who has what
2. **Few remaining cards**: Limited legal plays constrain options
3. **Forced sequences**: One card play often forces the entire sequence

Practical Implications

1. **Stop thinking at depth 4**: The game is decided - play quickly
2. **Endgame is solvable**: A simple lookup table could replace minimax for depth ≤ 4
3. **Real decisions are earlier**: Focus attention on tricks 1-4, not 5-7
4. **No "clutch" plays exist**: You can't outplay someone in endgame - it's predetermined

Connection to Other Findings

This aligns with: - **25a (Mistake Cost)**: Near-zero mistake cost at depth < 8 - **25f (Critical Positions)**: Endgame positions have low Q-spread - **25h (Count Capture)**: Q-spread ≈ 2.6 in endgame (and here we show it's actually 0 at depth ≤ 4)

Files Generated

- `results/tables/25n_endgame_patterns.csv` - Statistics
- `results/figures/25n_endgame_patterns.png` - Visualization

25o: Suit Exhaustion Signals

Key Question

When a player is void in a suit, how does optimal play change?

Method

1. Unpack states to get remaining hands (local indices → global domino IDs)
2. Detect voids: player has no dominoes containing a particular pip (suit 0-6)
3. Compare Q-spread and action distributions by void status and count
4. Control for game phase (depth) to isolate void effect

Key Findings

The Headline

Voids are ubiquitous - 100% of sampled states have at least one opponent void!

Metric	Value
States analyzed	1,000,000

Metric	Value
States with opponent void	**100%**
Mean Q-spread	2.76

Q-Spread by Total Voids

Total Voids	Mean Q-spread	n_states
2-7	2.0-3.0	8K
8-12	2.4-3.1	207K
13-17	2.6-3.5	610K
18-22	1.8-2.4	175K
26-27	**0.0**	884

Q-Spread by Game Phase

Phase	Depth Range	Mean Q-Spread
Early	20-28	3.00
Mid-Early	14-19	3.35
Mid-Late	7-13	2.91
Late	0-6	**1.95**

Interpretation

Voids don't create decisions - they narrow them

1. **Voids are not rare events**: By mid-game, every state has multiple voids

2. **Total voids track game phase**: More voids = later in game = simpler decisions
3. **Peak complexity at 16-17 voids**: Mid-game maximum before endgame collapse
4. **Endgame (26-27 voids) is forced**: Q-spread = 0, confirming 25n findings

Why 100% Have Voids

1. **Sampling bias**: First 100K rows of each shard are late-game states
2. **State distribution**: Oracle files have more late-game states (larger game tree early)
3. **7 suits per 7 dominoes**: With only 7 dominoes, voids are mathematically likely

Practical Implications

1. **Don't track individual voids**: They're everywhere - not actionable information
2. **Total void count proxies game phase**: Use it to gauge decision complexity
3. **Information value comes from WHICH void**: Specific suit voids reveal opponent hands
4. **Focus on void inference**: "They showed out of 4s" is more useful than "they have a void"

Connection to Other Findings

- **25n (Endgame)**: 100% forced at depth ≤ 4 aligns with Q-spread = 0 at 26-27 voids
- **25f (Critical Positions)**: remaining_pX features predict criticality - voids create asymmetry
- **25h (Count Capture)**: Q-spread decreases late game - voids are the mechanism

Files Generated

- `results/tables/25o_suit_exhaustion.csv` - Statistics
- `results/figures/25o_suit_exhaustion.png` - Visualization

Summary

Strategic analysis provides actionable guidance:

1. **Focus on tricks 3-4**: Highest mistake cost, most strategic value
2. **Don't overthink endgame**: 90%+ forced plays
3. **Bid with napkin formula**: ~30 + 6×doubles + 3×trumps
4. **Lead doubles early**: Establish control immediately
5. **Mistakes average 2-5 points**: Meaningful but not catastrophic
6. **Heuristics have limits**: Best single rule matches oracle only 34% - context matters

26: Austin 42 Verification

Validating Texas 42 folk wisdom analytically using oracle data.

26a: Threshold Cliffs

Key Question

Do P(make bid) drops at 30→31 and 35→36 show significantly larger cliffs than other transitions?

Folk Wisdom Claims

- 30→31 is "about losing one 10-count" (first point requiring count capture)
- 35→36 is "about losing one 5-count" (can only afford to lose one 5-count)

Method

- Use unified features (V_mean, V_std) for 200 hands
- Compute P(make bid B) = 1 - Φ((B - V_mean) / V_std) using normal approximation
- Calculate Δ(P) = P(make n) - P(make n+1) for each transition
- Test if key transitions show larger drops than baseline

Key Findings

Transition Drop Statistics

Transition	Mean Δ (%)	Excess vs Baseline	Key?
30→31	1.50%	-0.08%	Yes
31→32	1.48%	-0.09%	No
32→33	1.48%	-0.10%	No

Transition	Mean Δ (%)	Excess vs Baseline	Key?
33→34	1.48%	-0.10%	No
34→35	1.48%	-0.10%	No
35→36	1.54%	-0.04%	Yes
36→37	1.63%	+0.05%	No
37→38	1.66%	+0.08%	No
38→39	**1.74%**	**+0.16%**	No
39→40	1.72%	+0.14%	No
40→41	1.63%	+0.05%	No
41→42	1.48%	-0.09%	No

Baseline mean Δ: 1.58%

Statistical Test Results

Threshold	Mean Drop	Excess	p-value	Result
30→31 (10-count cliff)	1.50%	-0.08%	>0.05	**NOT CONFIRMED**
35→36 (5-count cliff)	1.54%	-0.04%	>0.05	**NOT CONFIRMED**

Interpretation

FOLK WISDOM NOT SUPPORTED BY DATA

Using the normal approximation method, neither claimed threshold cliff shows a significantly larger drop than surrounding transitions:

1. **30→31 drop is average**: 1.50% drop is actually slightly below the baseline (1.58%)
2. **35→36 drop is average**: 1.54% drop is also near the baseline

3. **Largest drop at 38→39**: The biggest excess (+0.16%) occurs at 38→39, not at the claimed thresholds

Caveats

1. **Normal approximation**: This analysis assumes V is normally distributed. The true distribution may have discrete jumps at count thresholds that the approximation misses.
2. **Limited sample**: 200 hands may not have sufficient power to detect threshold effects.
3. **Marginalized data needed**: A more rigorous test would use the full V distribution from marginalized data (3 opponent configurations per hand).

Files Generated

- `results/tables/26a_threshold_cliffs.csv` - Transition statistics
- `results/figures/26a_threshold_cliffs.png` - 4-panel visualization

26f: Coverage vs Trump Count

Key Question

Does "coverage" (ability to compete in multiple suits) matter as much as trump count?

Folk Wisdom Claim

"2 trumps + perfect coverage beats 4 trumps + naked lows"

Translation: Off-suit quality matters as much as raw trump count.

Method

- Define `coverage_score` = composite measure of off-suit quality:
- +1 per domino beyond first in each off-suit (depth bonus)
- +1 if max rank in suit ≥ 5 (high card bonus)
- -2 for singleton with rank ≤ 2 (naked low penalty)

- Regress E[V] on trump_count + coverage_score
- Compare standardized coefficients (β) for fair comparison

Key Findings

Bivariate Correlations with E[V]

Feature	r	p-value	Interpretation
trump_count	**+0.229**	0.0011	More trumps → higher E[V]
coverage_score	**-0.333**	1.4×10^{-6}	More coverage → LOWER E[V]

Multivariate Regression

Feature	Coef	95% CI	p-value	β (standardized)
trump_count	+1.55	[-0.34, 3.44]	0.107	+0.117
coverage_score	**-1.63**	[-2.44, -0.82]	**0.0001**	**-0.288**

$R^2 = 0.123$

Effect Size Comparison

- |β(coverage)| / |β(trump)| = **2.45×**
- Coverage effect is 2.45× larger in magnitude than trump effect
- BUT coverage effect is **NEGATIVE** - higher coverage → worse outcomes

Interpretation

FOLK WISDOM REFUTED (INVERTED)

The data shows the **opposite** of folk wisdom:

1. **Coverage hurts, not helps**: Higher coverage_score is associated with **lower** E[V]
2. **Trump count helps**: More trumps correlate with higher E[V] (though not significant after controlling for coverage)

3. **Voids are valuable**: The negative coverage effect suggests that voids (enabling trump plays) are more valuable than being "covered" in all suits

Why Coverage Hurts

1. **Voids enable trumping**: When you have no cards in a suit, you can trump when opponents lead it
2. **Following suit is weak**: If you must follow with a low card, you lose the trick
3. **Trumping is powerful**: Cutting in with a trump often wins even against high leads
4. **Coverage = commitment**: Being spread across all suits means fewer trumps and fewer void-based ruff opportunities

Revised Folk Wisdom

Correct interpretation: "4 trumps + voids beats 2 trumps + coverage"

Having voids (which the coverage metric penalizes as depth=0) is actually beneficial because it enables trump plays. The folk wisdom appears to have the relationship backwards.

Files Generated

- `results/tables/26f_coverage_vs_trump.csv` - Summary statistics
- `results/figures/26f_coverage_vs_trump.png` - 4-panel visualization
