

# How Watson Learns Superhuman Jeopardy! Strategies



Gerald Tesauro, IBM Research

Joint work with: David Gondek, Jonathan Lenchner,  
James Fan, John Prager

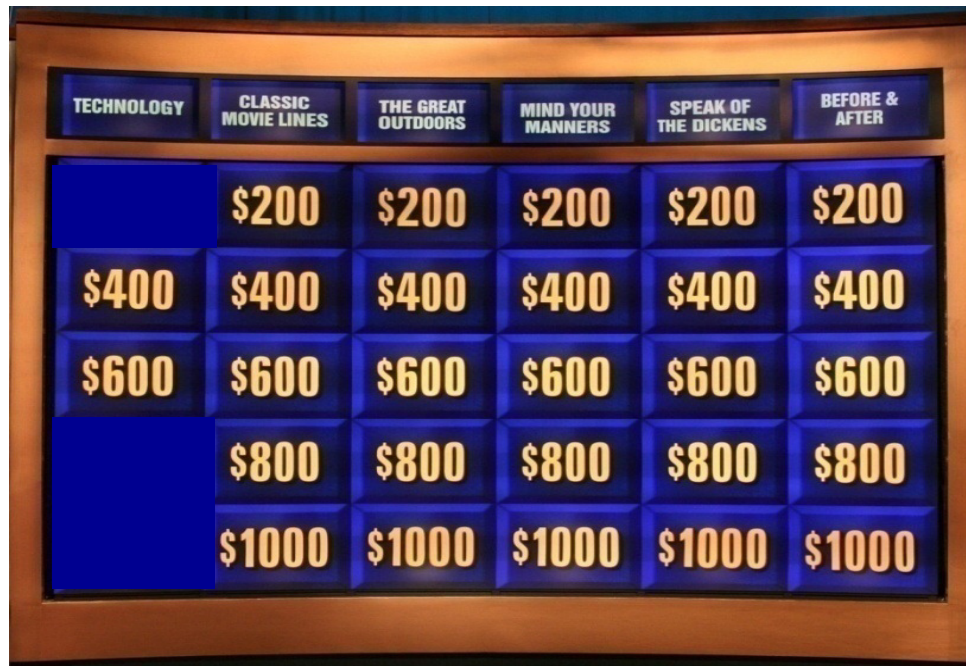


# Outline

- 1) Overview of Jeopardy! Grand Challenge
- 2) Significance of Game Strategy
- 3) Building a Faithful-Enough J! Simulator
- 4) Learning & Optimizing Strategies in Simulation:
  - a) Daily Double betting
    - **Neural nets + Reinforcement Learning:** (TD-Gammon redux)
  - b) Final Jeopardy betting:
    - **“Best Response”** to Human FJ model
  - c) Clue selection
    - DD Seeking using **Bayesian Inference**
  - d) Confidence Threshold for Buzz-in
    - **Approximate Dynamic Programming + real-time “Rollouts”**

## Jeopardy! Gameplay Basics

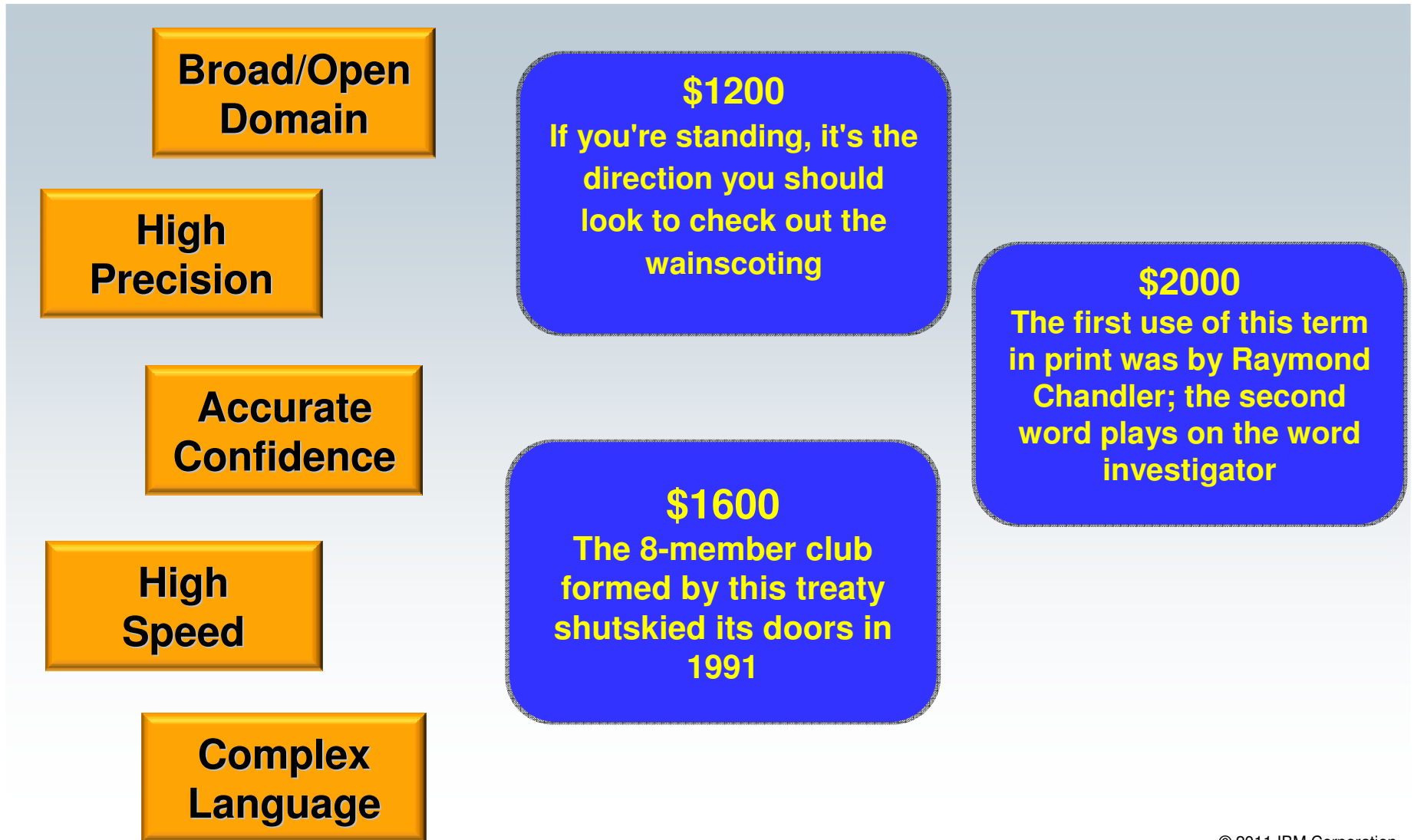
- Select a clue by category and dollar value
- Press buzzer to attempt to answer
- Gain \$\$ if right, lose \$\$ if wrong



**\$600**  
The first chapter of this unfinished novel is titled "The Dawn."

What is "The Mystery of Edwin Drood?"

# The Jeopardy! Challenge: *A compelling and notable way to drive and measure the technology of automatic Question Answering along 5 Key Dimensions*

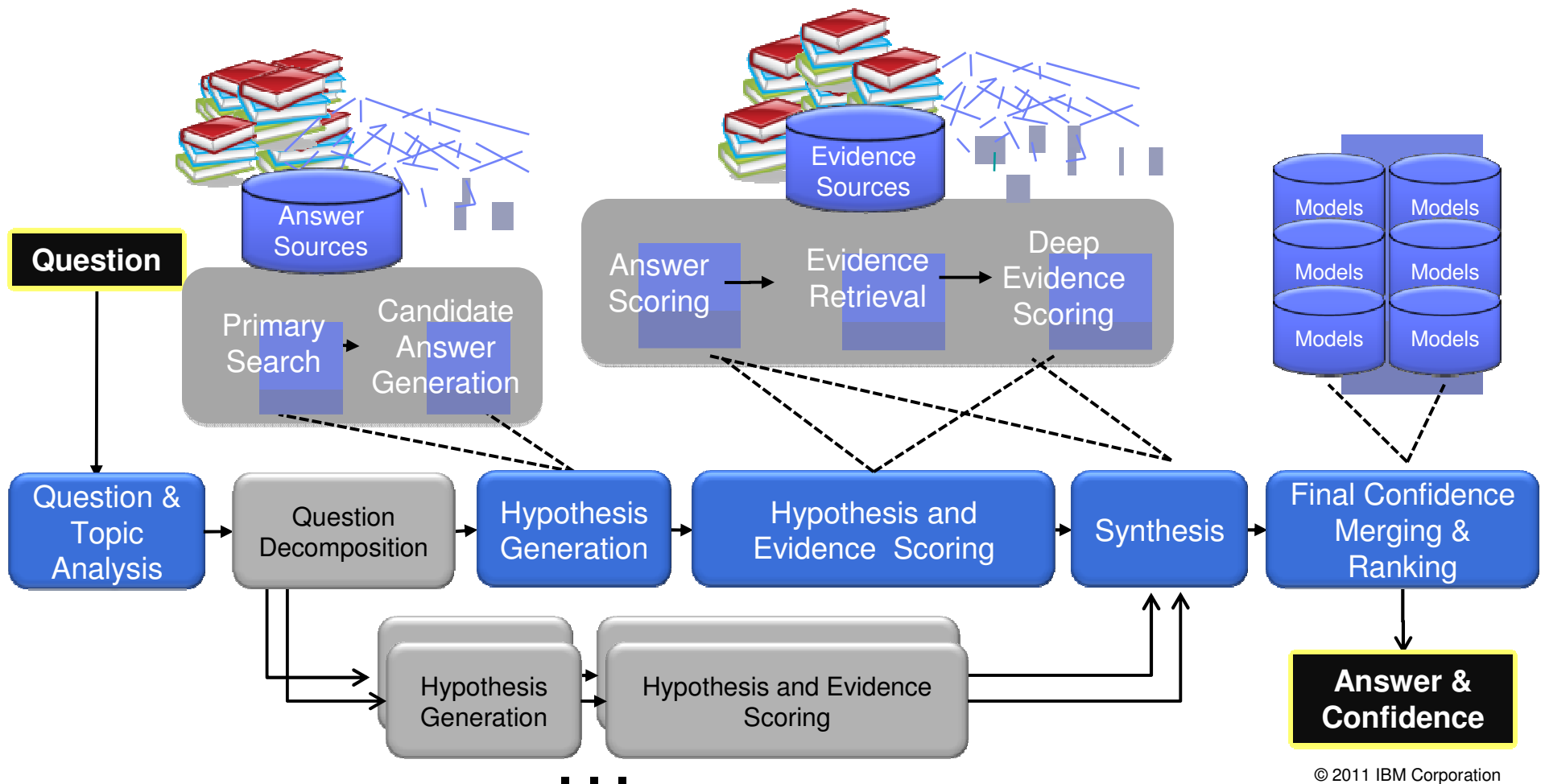


# DeepQA: The Technology Behind Watson



## Massively Parallel Probabilistic Evidence-Based Architecture

Generates and scores many hypotheses using a combination of 1000's **Natural Language Processing, Information Retrieval, Machine Learning** and **Reasoning Algorithms**. These gather, evaluate, weigh and balance different types of **evidence** to deliver the answer with the best support it can find.



# Watson's Competitive Record

- Sparring Games Series #1: 73 games vs. former Jeopardy! contestants (1 or 2 appearances)
  - **47-15-11** cumulative record (**64.4% 1sts**), 21 **lockouts\***
- Sparring Games Series #2: 55 games vs. Jeopardy! masters (Tournament of Champions finalist or semi-finalist)
  - **39-8-8** cumulative record (**70.9% 1sts**), 30 lockouts
- Exhibition Match vs. Ken Jennings and Brad Rutter
  - Watson 77,147 (**1<sup>st</sup> place by lockout**: \$1,000,000)
  - Ken 24,000 (2<sup>nd</sup> place: \$300,000)
  - Brad 21,600 (3<sup>rd</sup> place: \$200,000)

**“Lockout:”** guaranteed win, leader cannot be caught

# Four Challenging Aspects of Jeopardy! Strategy

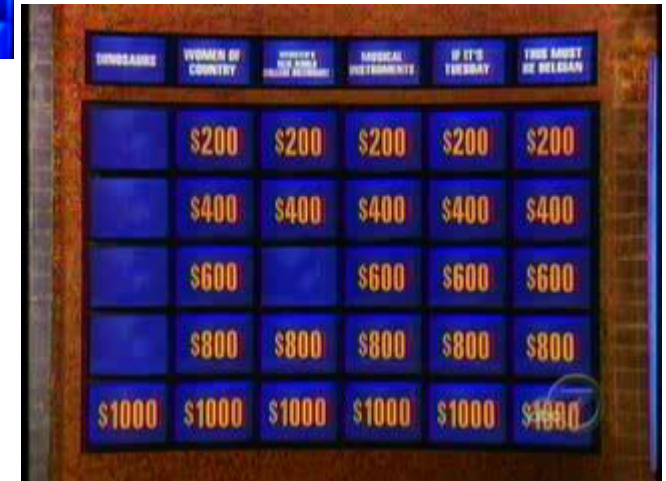
1) Daily Double wager



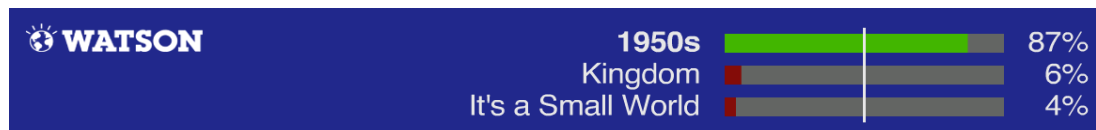
2) Final Jeopardy wager



3) Clue selection



4) Confidence threshold for buzz-in



# Motivating Quantitative J! Strategy

- Our work gives Watson an edge over humans; yields substantial boost in winning chances (vs. simple heuristic strategies)
- We are extending this approach to Decision Analytics in health care, pricing, security domains
- Never been done before; led to new theories of evaluating / playing in various J! game states



# Why Build a Jeopardy Simulator?

- Three classic metrics for optimizing game programs:
  1. Test performance in live games
  2. Test performance in simulated games
  3. Evaluate over a collection of test positions
  
- #1 is the Gold Standard ...
  - But it's very slow and expensive (especially for J!)
- #3 is unreliable (“overfitting” phenomenon)
- #2 is much faster and cheaper than live testing
  - Orders of magnitude more data for learning/optimization
  - Is the simulation model “faithful enough” to be useful??

# Approach to Modeling Jeopardy!

- Detailed modeling more difficult than in classic board games (chess, checkers, Go, etc.)
  - modeling range of contestant knowledge across many categories would be highly challenging
  - modeling distributions of categories, clues would be equally challenging
  
- We resort to extreme simplification: models average over all contestants, categories, clues!
  - average stochastic process models for regular clues, DDs, FJ, clue selection
  - keep in mind, we really want to model **humans vs. Watson**, not humans vs. humans

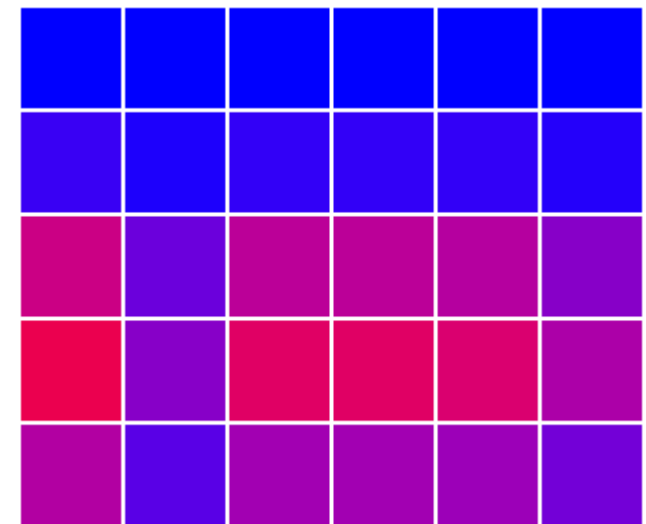
# Components of a J! Game Simulator

- We obtained detailed historical records of thousands of past J! episodes from **J! Archive** ([www.j-archive.com](http://www.j-archive.com))
- Model **Daily Double** placement
- Model **Human Contestant** performance profile
  - How often they attempt to buzz in
  - How often they are right/wrong when they win the buzz
  - Accuracy and betting patterns on Daily Doubles
  - Accuracy and betting patterns in Final Jeopardy
- We built three different human models:
  - “**Average Contestant**” model: average over all non-tournament J! episodes (ex-College, Teen, Celebrity games)
  - “**Champion**” model: All-time top 100 player stats
  - “**Grand Champion**” model: All-time top 10 player stats

# Modeling Daily Double Placement

## ▣ **Statistics over 9k DDs** (3k Round1, 6k Round2):

- (Widely known) DDs most frequent in the high-value rows (third, fourth, fifth) with harder clues
  - Row frequencies published on J! Archive
- (Previously unknown) Some columns are more likely than others to have a DD!
  - First column most likely to have a DD
  - Second column least likely to have a DD



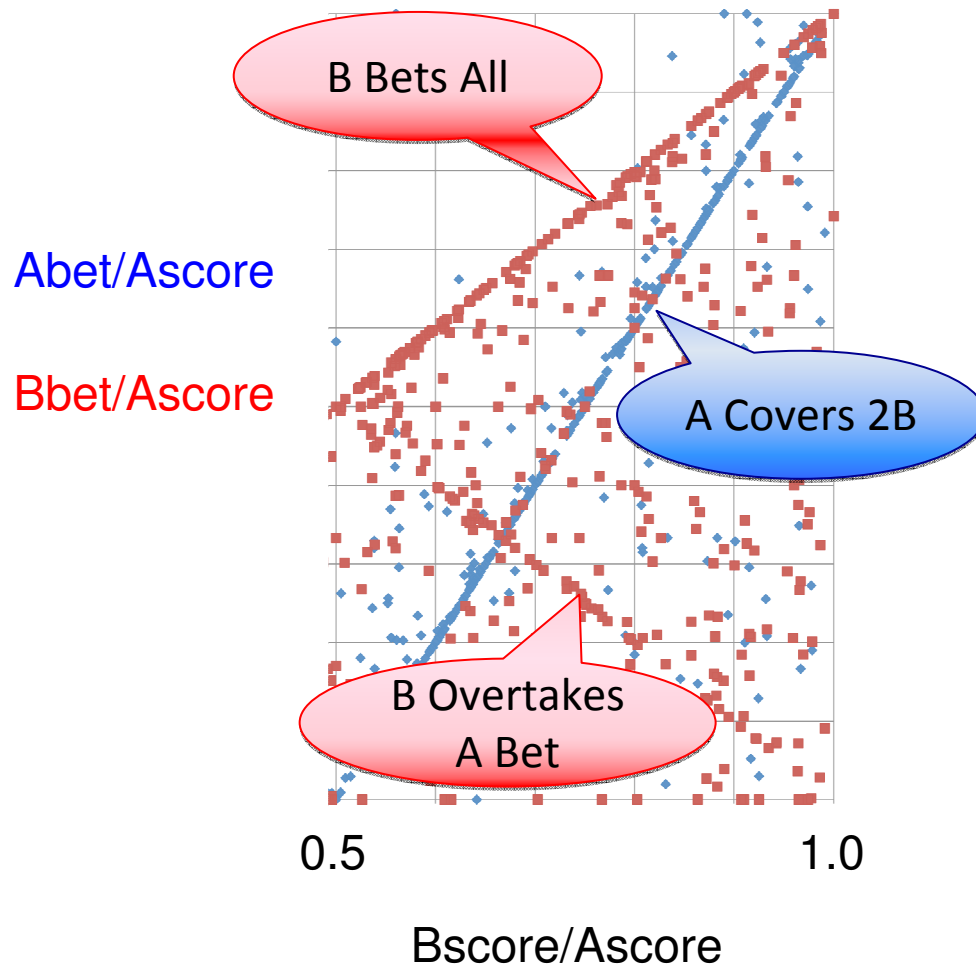
Unlikely  Likely

- ▣ row-column frequencies used to randomly place DDs in simulated games; Watson uses them as Bayesian prior

# Modeling Human Final Jeopardy

Average FJ accuracy  $\approx 50\%$  FJ right/wrong correlation  $\approx 0.3$

Bets depend on score positioning: 1<sup>st</sup> place ("A"), 2<sup>nd</sup> place ("B"), 3<sup>rd</sup> place ("C")

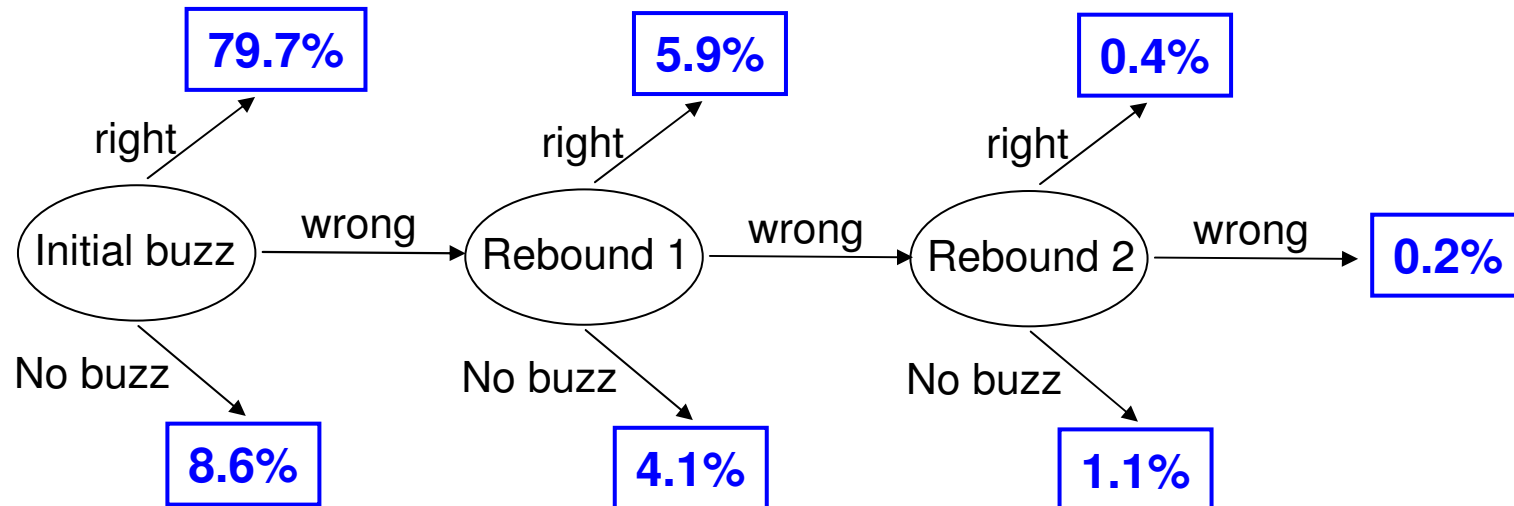


- Segment data into 4 groups by 2 binary splits:
  - $B \geq 2A/3$  ?
  - $B \geq 2C$  ?
- Stochastic betting models for A,B,C fit bets in each group

Win rates in 2092 past episodes:		
	real	model
A	65.3%	64.8%
B	28.2%	28.1%
C	7.5%	7.4%

# Stochastic Process Model of Regular Clues

Statistics over 150K regular-clue (no DD) outcomes:



Derive avg. contestant precision / buzz rate model:

precision  **$p = 0.87$**

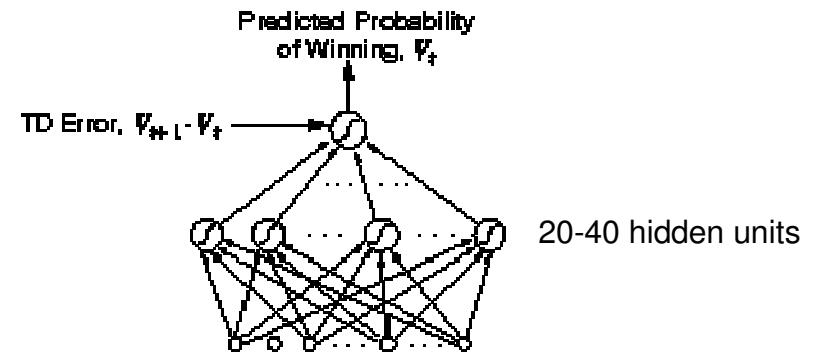
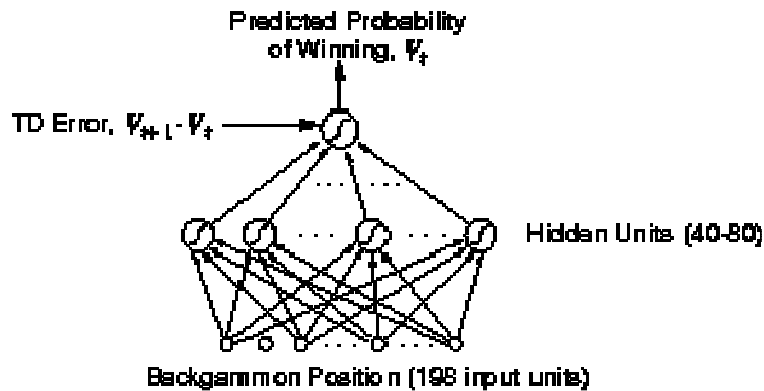
right/wrong correlation  **$\rho_p = 0.2$**  (known from rebound stats)

buzz attempt rate  **$b = 0.61$**

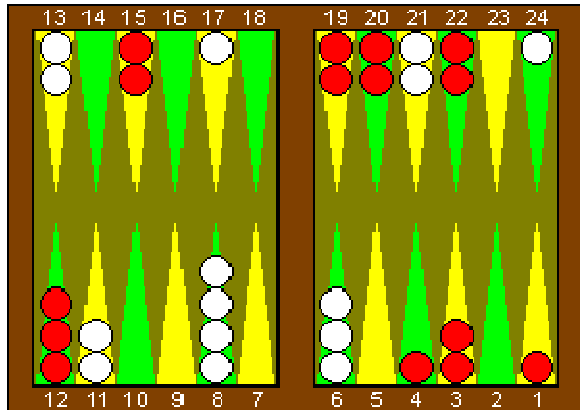
buzz/no-buzz correlation  **$\rho_b = 0.2$**

## Watson's Daily Double Betting Strategy

- Train an Artificial Neural Net over millions of simulated games pitting Watson vs. two simulated human opponents
- Use TD( $\lambda$ ) reinforcement learning algorithm just as in TD-Gammon ☺



Jeopardy game state (23 input units)



BRITISH HISTORY DATEBOOK	TV's SUPPORTING CASTS	POTPOURRI	THE WHISKEY TRAIL	"ARD" STUFF	RHETT-ORIC
\$200		\$200	\$200	\$200	\$200
\$400		\$400	\$400	\$400	\$400
\$600	\$600	\$600	\$600	\$600	\$600
	\$800	\$800	\$800	\$800	\$800
	\$1000	\$1000	\$1000	\$1000	
Watson \$1000		Gerry \$200		Jon \$1000	

## Simplified List of “State” Variables

- Scores of three players
- Round (SingleJ, DoubleJ, FinalJ)
- # of remaining clues
- total \$ value of remaining clues
- # of remaining Daily Doubles
- player with control of board

(Watson also has in-category confidence, from right/wrong answers to previous clues in the category.)



## Computing Optimal DD Bet

- Let  $\mathbf{s}$  = Watson's score and  $\mathbf{V} = \mathbf{V}(\mathbf{s}, \dots)$  = NN output = Watson's win prob in the current game state.

- “Equity” (expected utility/winprob) of a bet:

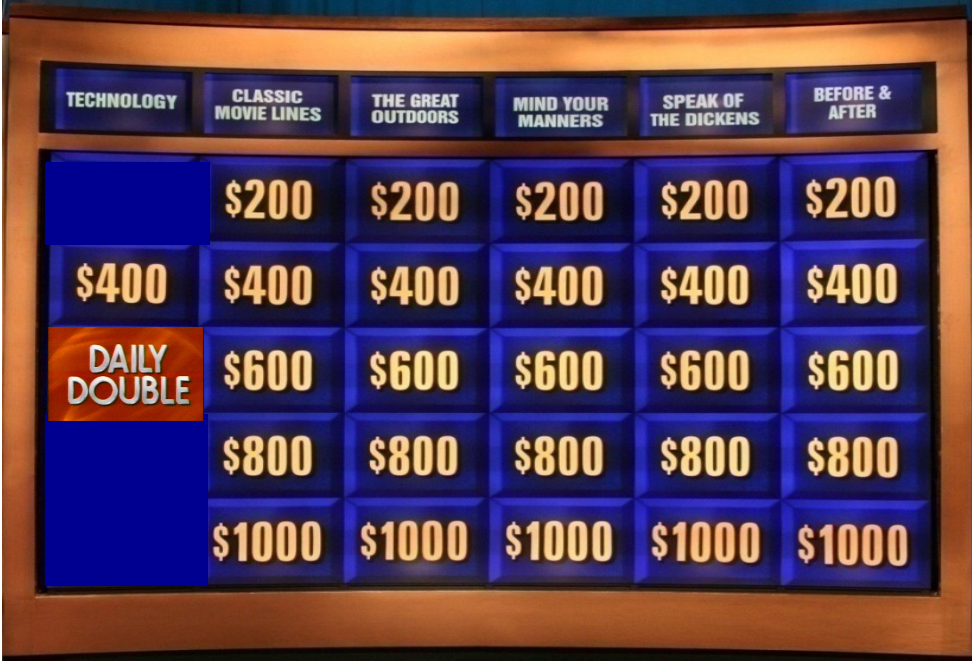
$$\mathbf{E}(\text{bet}) = \text{conf} * \mathbf{V}(\mathbf{s}+\text{bet}) + (1-\text{conf}) * \mathbf{V}(\mathbf{s}-\text{bet})$$

where  $\text{conf}$  = Watson's in-category confidence

- Best **risk-neutral** bet maximizes  $\mathbf{E}(\text{bet})$
- **Risk mitigation:**
  - Penalize bets with high volatility (std. deviation)
  - Prohibit bets that entail “too much” downside risk
  - → significantly reduces risk, only costs 0.3% equity

## Illustrative Example of NN DD betting

- Start of DoubleJ, Watson ran the column and then found the first DD. Watson leads (11000, 4200, 4200).



	TECHNOLOGY	CLASSIC MOVIE LINES	THE GREAT OUTDOORS	MIND YOUR MANNERS	SPEAK OF THE DICKENS	BEFORE & AFTER
	\$200	\$200	\$200	\$200	\$200	\$200
\$400	\$400	\$400	\$400	\$400	\$400	\$400
<b>DAILY DOUBLE</b>	\$600	\$600	\$600	\$600	\$600	\$600
	\$800	\$800	\$800	\$800	\$800	\$800
	\$1000	\$1000	\$1000	\$1000	\$1000	\$1000

**BALLET DANCERS**

**THAT'S FOUL!**

**"E" IN SCIENCE**

**WHAT'S NEXT ON THE  
LIST?**

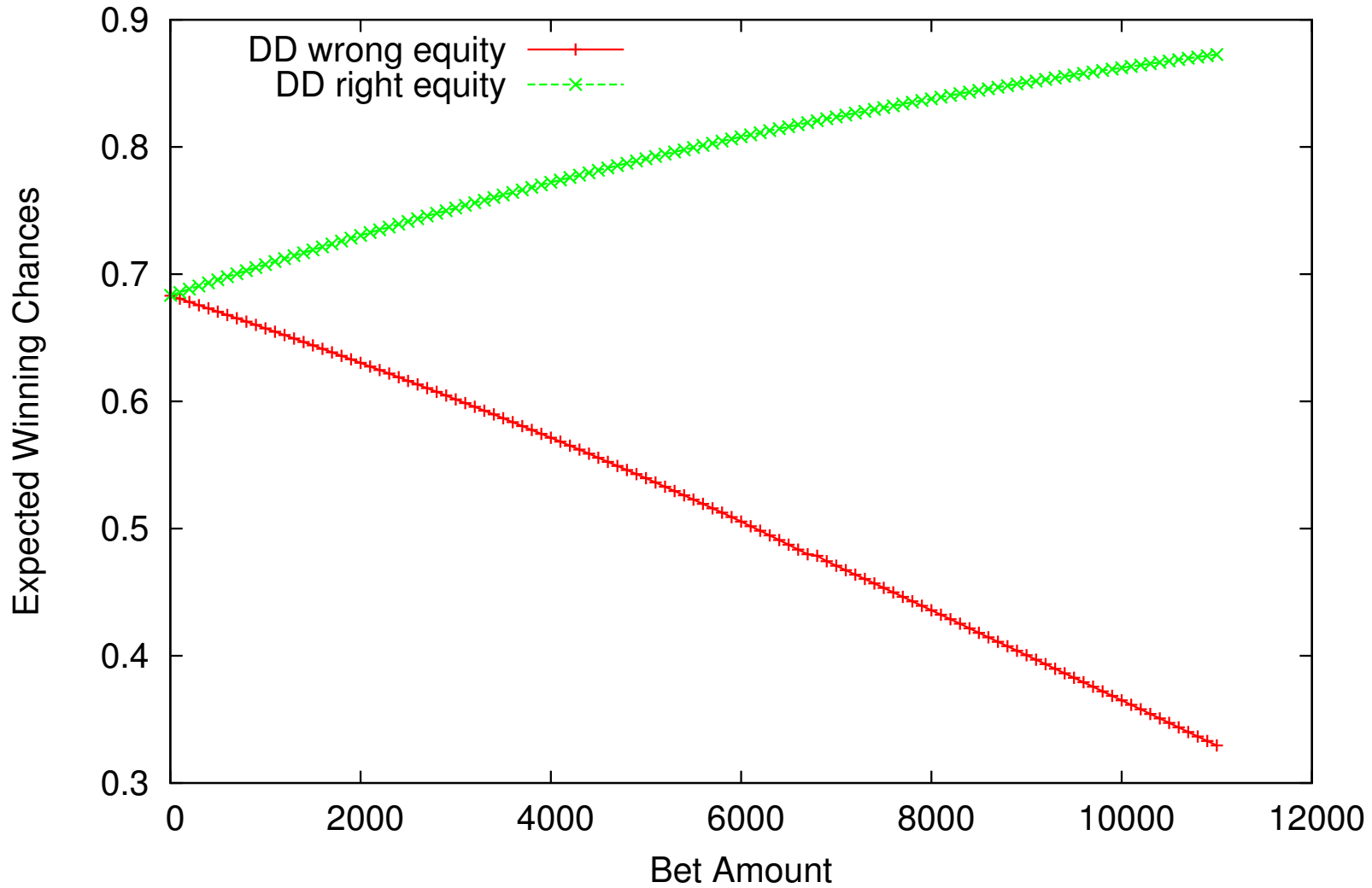
**THE '70s**

**CHANGE A LETTER**

**THIS RUSSIAN  
BALLERINA'S  
LONDON HOME,  
IVY HOUSE,  
WAS FAMOUS FOR ITS  
ORNAMENTAL LAKE  
WITH SWANS**

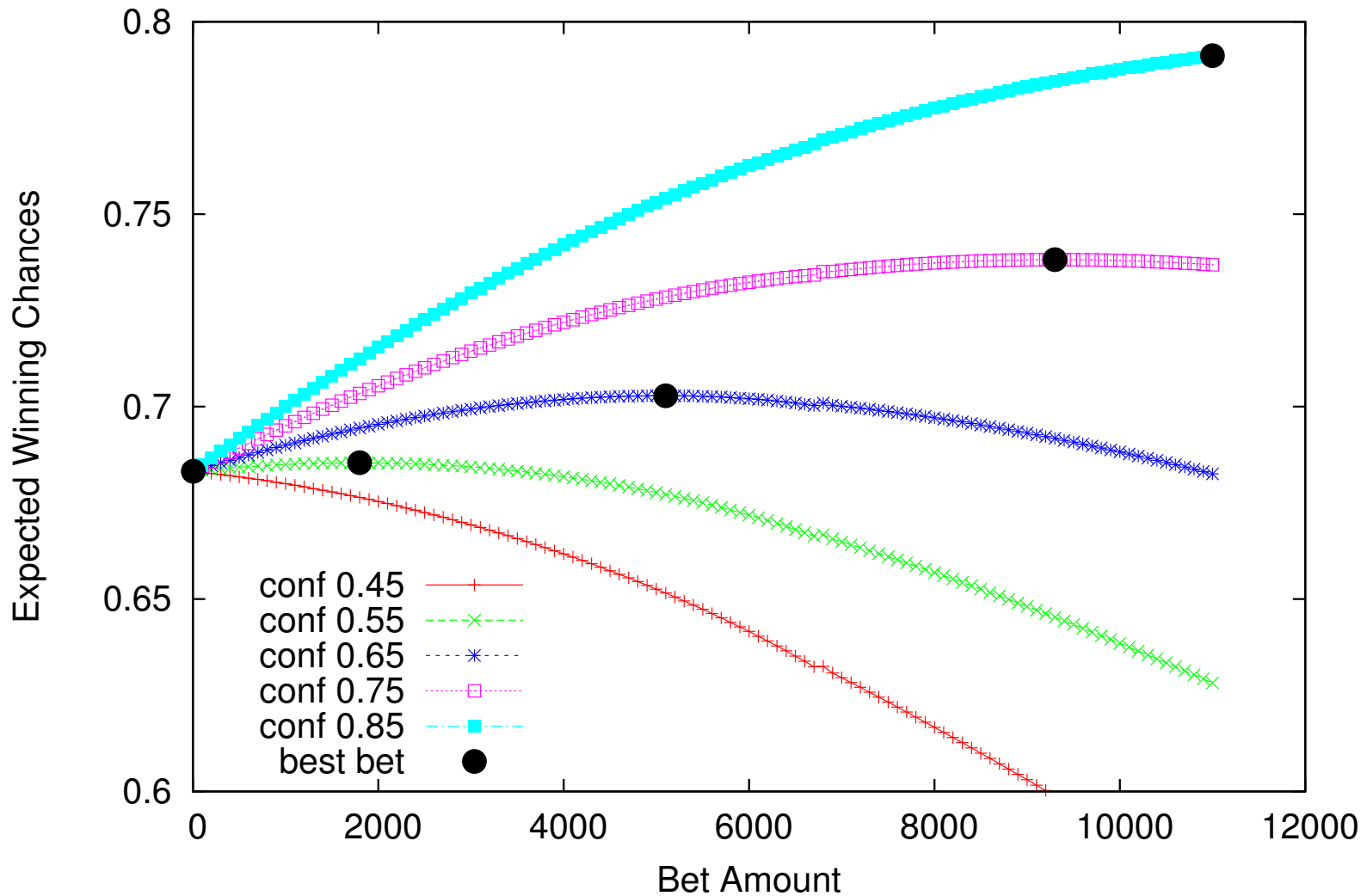
# NNDD Analysis

(11000, 4200, 4200) Watson Daily Double Bet



# NNDD Analysis

(11000, 4200, 4200) Watson Daily Double Bet



## NN DD Performance Metrics

- Increase in simulated wins (vs. average contestant model)
  - Previously we employed a Heuristic DD betting algorithm for Series 1 Sparring Games
    - Heuristic did not take confidence into account
  - Watson simulated win rate using Heuristic DD: **61%**
  - Simulated win rate using NNDD + default confidence: **64%**
  - Simulated win rate using NNDD + live confidence: **67%**
  
- Evaluate “equity loss” of NN DD bets with extensive offline Monte Carlo analysis (essentially perfect analysis of which bet achieves the most simulated wins):
  - Avg. NN DD equity loss = **0.6% per DD bet**
  - Most errors occurred in endgames with ample lockout potential
  - We eliminated these errors in Series 2 Sparring Games using live MC rollouts to make endgame DD bets
    - (NN + endgame MC) equity loss = **0.25% per DD bet**

# Watson's Final Jeopardy Betting Strategy

- Live Best-Response to Randomized Human FJ Model:
  - Analytic probabilities of the eight possible right/wrong triples
  - Draw ~10k samples of human bet pairs
  - For each legal Watson bet, compute prob (Watson wins) given the bet pair and the right/wrong probs
- Can extract logical betting rules from Best-Response output:  
IF  $(B \geq 2A/3)$  AND  $(B < 2C)$  {  
    IF  $(2C-B) \leq (3B-2A)$  THEN BET =  $2C-B$   
    ELSE BET = B  
}
- 3% more wins than simple-minded FJ heuristic

Win rates in 2092 historic FJs:		
	human	BR
A	65.3%	67.0%
B	28.2%	34.4%
C	7.5%	10.5%

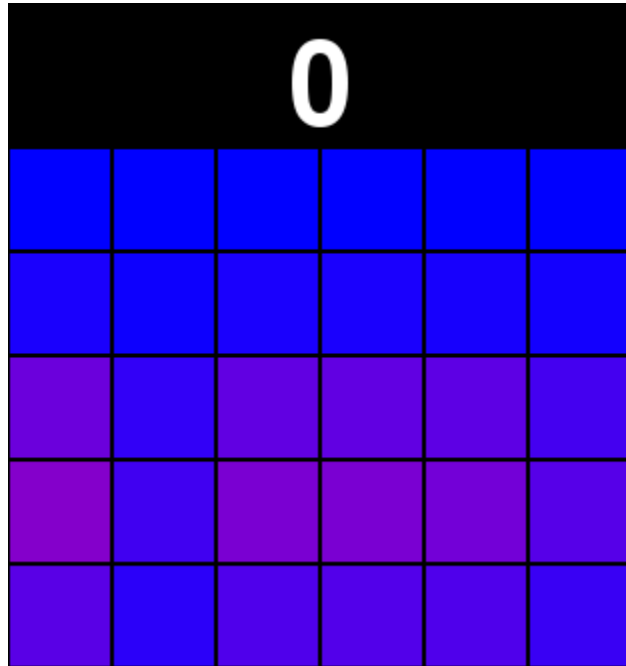
## Watson's Clue Selection Strategy

- Systematically studied trade-offs between three considerations:
  - 1. Finding Daily Doubles:
    - use historical DD locations as Bayesian prior
    - combine prior probs with revealed clue evidence using Bayes' rule to obtain posterior DD probs
  - 2. Keeping control of the board:
    - tend to stay in categories where Watson is doing well
  - 3. Learning the “gist” of a category from revealed clues
    - tend to pick low-value clues, to do better on high-value clues
  
- Resulting strategy that maximizes simulation win rate:
  - If DDs left, ~90% #1 and ~10% #2
  - If no DDs left, 100% #3.

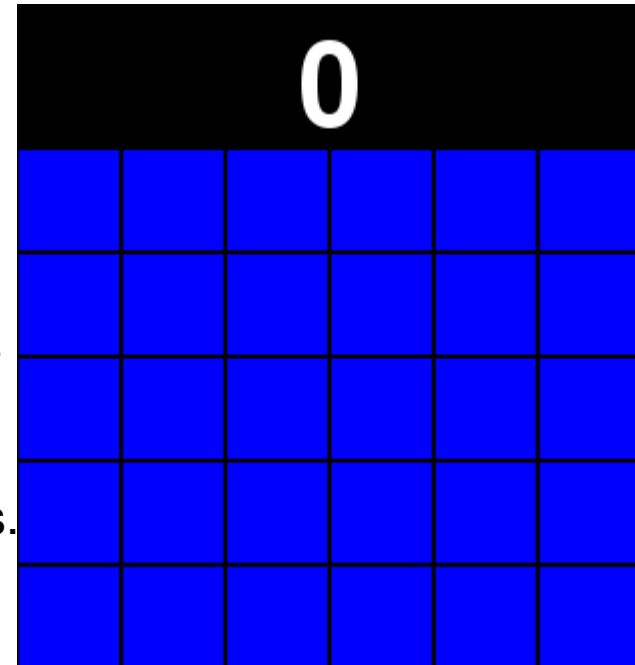


# Daily Double Seeking Animation

## Watson



## Human



Watson finds DDs in 65% of the time it takes humans.



## Endgame Buzz Threshold

Compute a set of buzz thresholds

$$\Theta = (\theta_0, \theta_1, \theta_2, \theta_3)$$

Determines a set of buzz decisions

$$\mathbf{B} = (b_0, b_1, b_2, b_3)$$

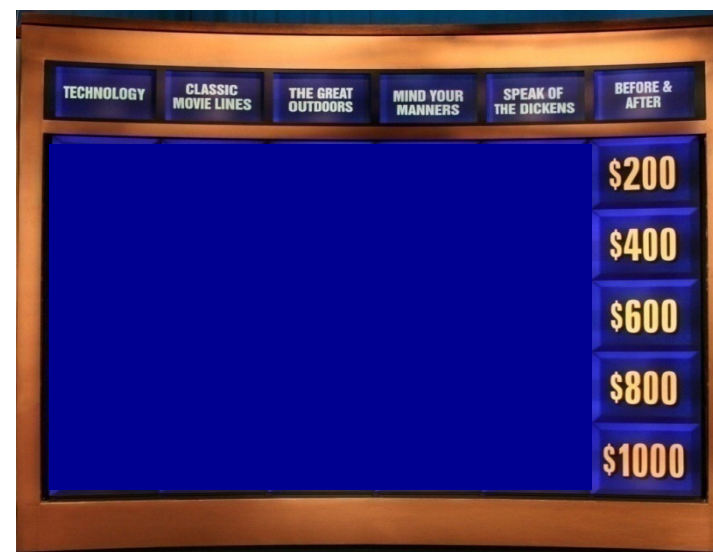
given Watson's confidence

0 = initial buzz

1 = rebound, human #1 wrong

2 = rebound, human #2 wrong

3 = rebound, both humans wrong



Solve using recursion relation between  
V(K clues left) and V(K-1 clues left):

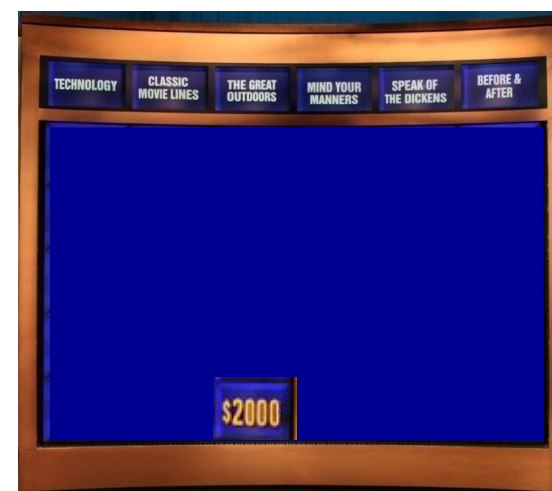
$$V_k(s) = \int \rho(c) \sum_{j=1}^5 p(D_j) \max_{\vec{B}(c, D_j)} \sum_{\delta} p(\delta | \vec{B}, c) V_{k-1}(s'(\delta, D_j)) dc$$

## Dynamic Programming Solution

- Recursion relation can be solved exactly using DP
  - Build the full search tree going from K clues left to 0 clues left (i.e., FJ)
  - Evaluate the FJ states (either MC or pre-tabulated)
  - Work backwards to evaluate 1, 2, ..., K clues left states
  - Blows up exponentially; too slow for live play (we need the buzz threshold in maybe 1-2 seconds tops)
  
- We used an **Approximate DP** technique:
  - Solve the exact recursion relation only for the first step ( $K \rightarrow K-1$  left)
  - Evaluate the (K-1) left states using MC
  - Pretty good approximate solution (at least for  $K \leq 5$ )
  - almost always takes  $< 2$  seconds.

## Buzz Threshold -- Illustrative Examples

- One clue remaining worth \$2000
  - Humans have (13000, 6800); consider various Watson scores:
- 23000 13000 6800 → Thresh 1.0000
  - Watson can't get a lockout, B can get to 2/3, if Watson buzzes and is wrong.
- 25000 13000 6800 → Thresh 0.0000
  - Free shot to try for a lockout
  - no decrease in equity if wrong
- 27000 13000 6800 → Thresh 0.6444
  - Could try to prevent B from answering, need to be pretty confident
- 29000 13000 6800 → Thresh 0.0000
  - Free shot to prevent B from answering – no equity loss if wrong



# Conclusions

- First ever quantitative, principled, comprehensive J! strategy
- Our strategies for Watson perform beyond human capability:
  - FJ: slight edge
  - DD: clear edge
  - Clue selection: moderate edge (Bayesian DD seeking)
  - Endgame buzzing: clear edge in special situations
- Superhuman strategy was a significant factor in Watson's overall competitive record

# References



This Is Watson



G. Tesauro et al., Analysis of Watson’s strategies for playing Jeopardy!, J of AI Research (2013).

G. Tesauro et al., Simulation, learning and optimization techniques in Watson’s game strategies, IBM J of R&D (2012).