Deep Learning for Poker: Inference From Patterns in an Adversarial Environment

Nikolai Yakovenko, PokerPoker LLC
CU Neural Networks Reading Group
Dec 2, 2015

- This work is not complicated
- Fully explaining the problem would take all available time
- So please interrupt, for clarity and with suggestions!

Convolutional Network for Poker

Our approach:

- 3D Tensor representation for any poker game
- Learn from self-play
- Stronger than a rule-based heuristic
- Competitive with expert human players
- Data-driven, gradient-based approach

Poker as a Function

Private Cards

Public Cards

Previous Bets (Public)

(action policy)

Bet/Raise = X%

Check/Call = Y%

Fold = Z%

X + Y + Z = 100%

(action value)

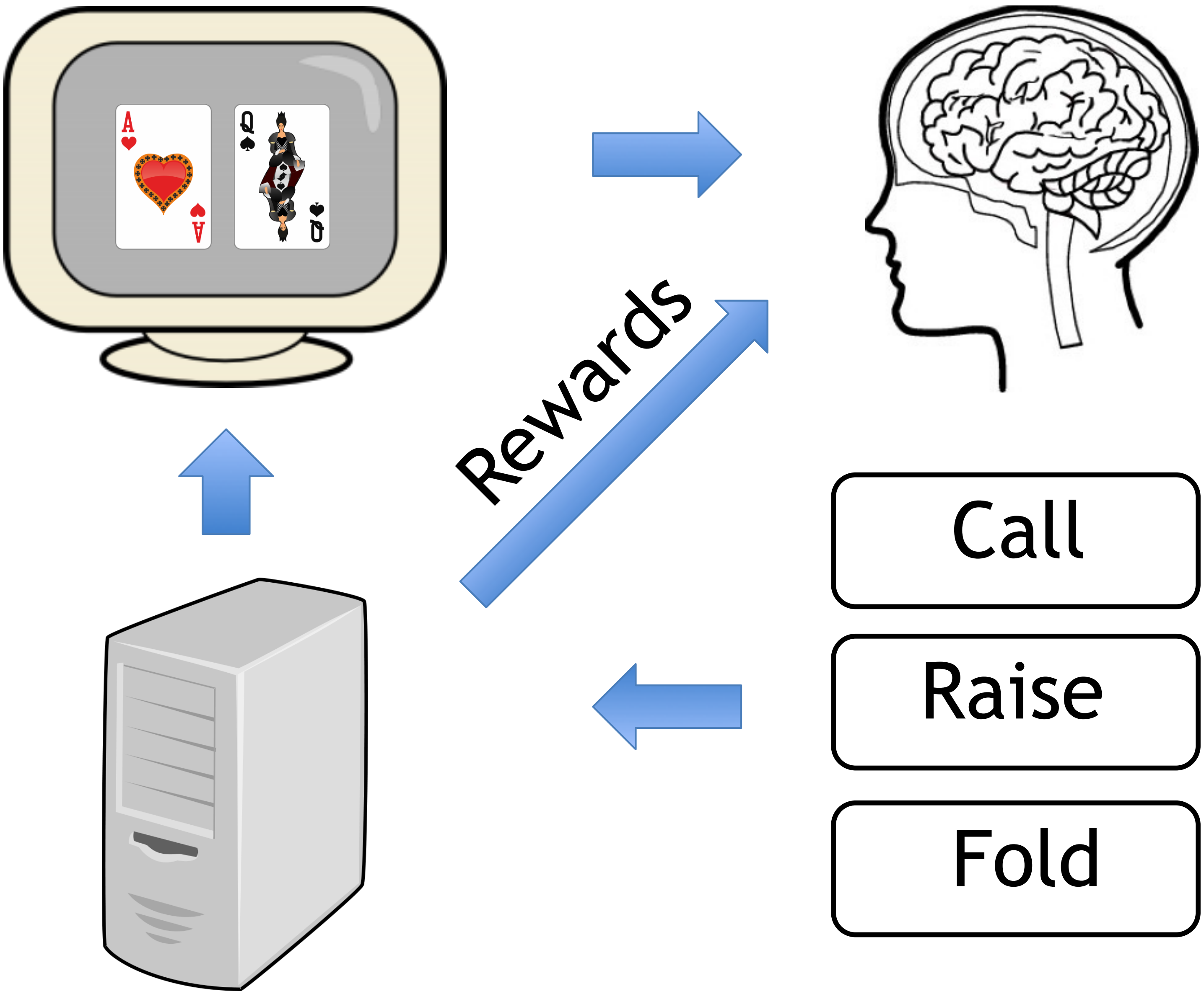
Bet = $X

Check = $Y

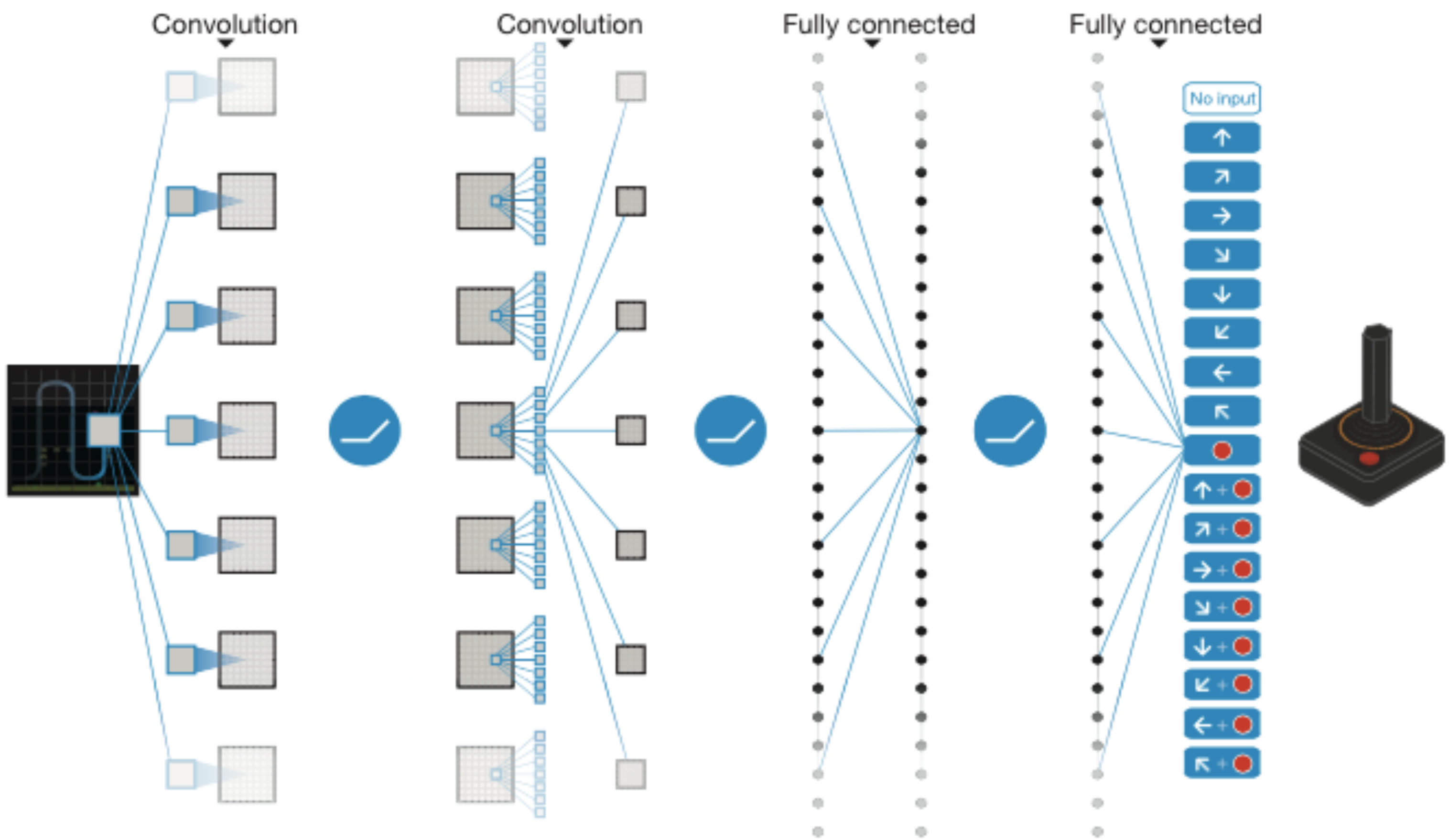
Fold = $0.0

Explore & Exploit

Poker as Turn-Based Video Game

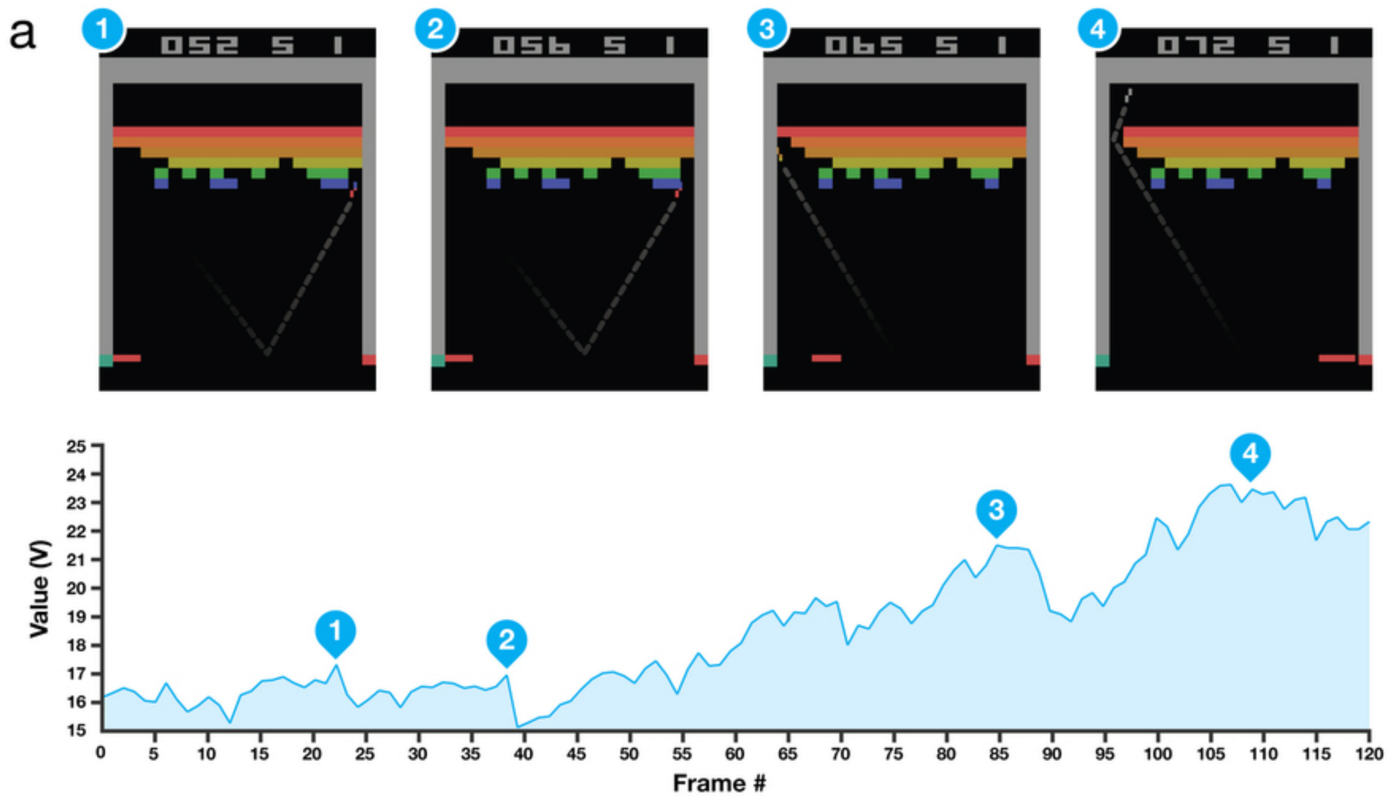


Special Case of Atari Games?



Input Convolutional Network Action Values

Value Estimate Before Every Action



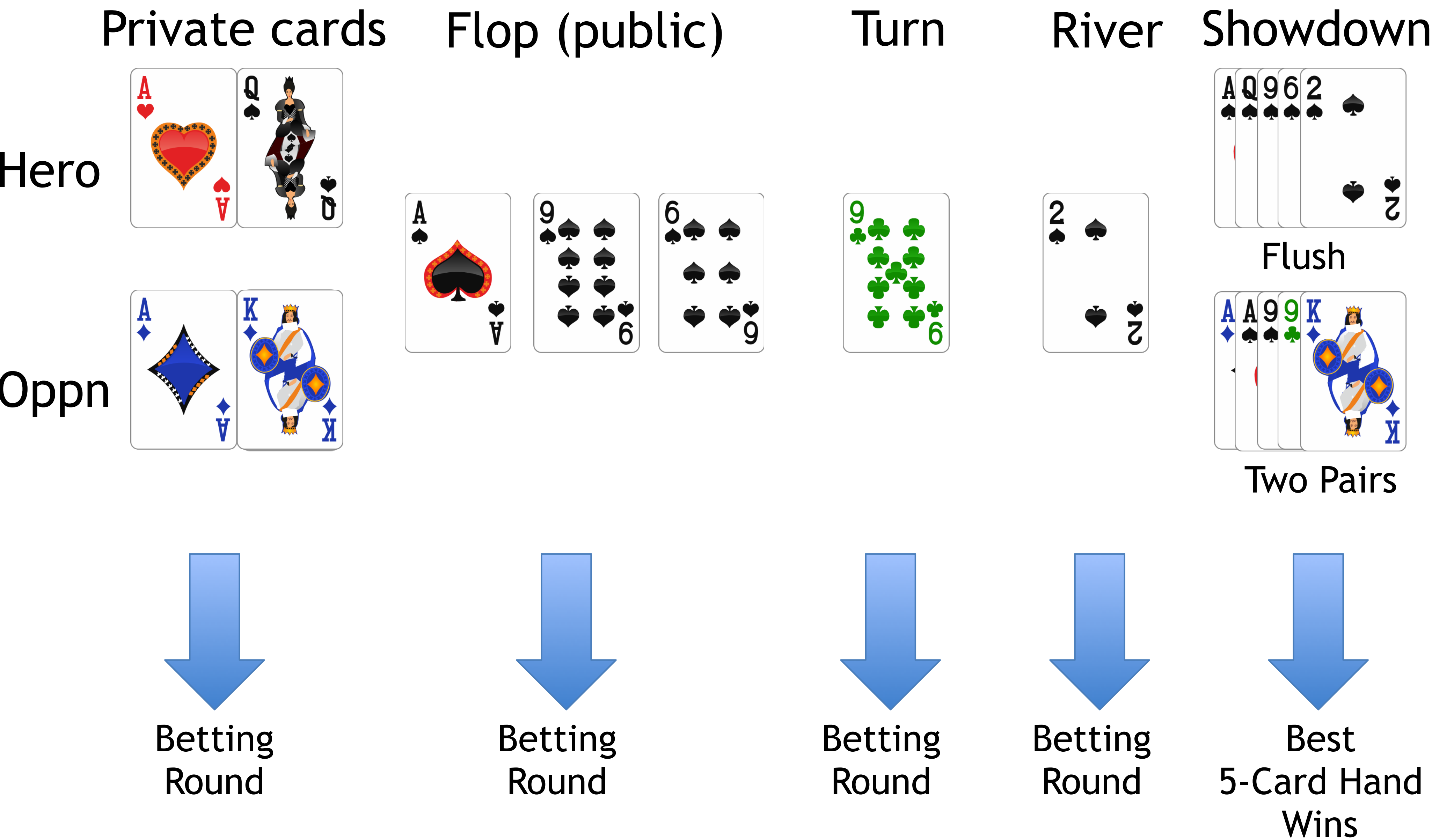
Frame ~ turn-based poker action

Discounted reward ~ value of hand before next action [how much you'd sell for?]

More Specific

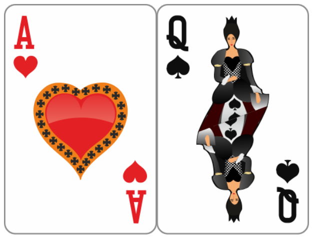
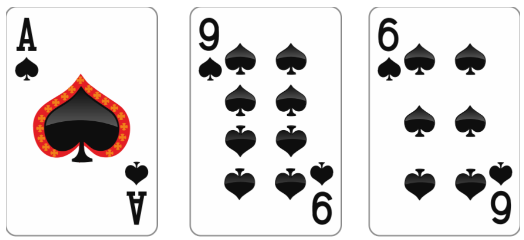
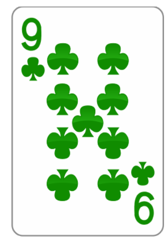
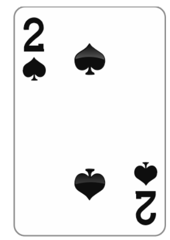
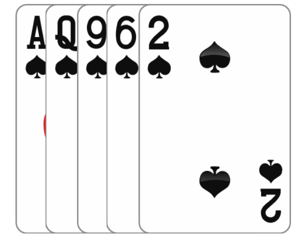
- Our network plays three poker games
  - Casino video poker
  - Heads up (1 on 1) limit Texas Hold'em
  - Heads up (1 on 1) limit 2-7 Triple Draw
  - Can learn other heads-up limit games
- We are working on heads-up no-limit Texas Hold'em
- Let's focus on Texas Hold'em

Texas Hold’em



Representation: Cards as 2D Tensors

Private cards | Flop (public) | Turn | River | Showdown

Flush

```
[AhQs]

x23456789TJQKA
c.............
d.............
h............1
s...........1.
```

```
[AhQs]+[As9s6s]

x23456789TJQKA
c.............
d.............
h............1
s....1..1...11
```

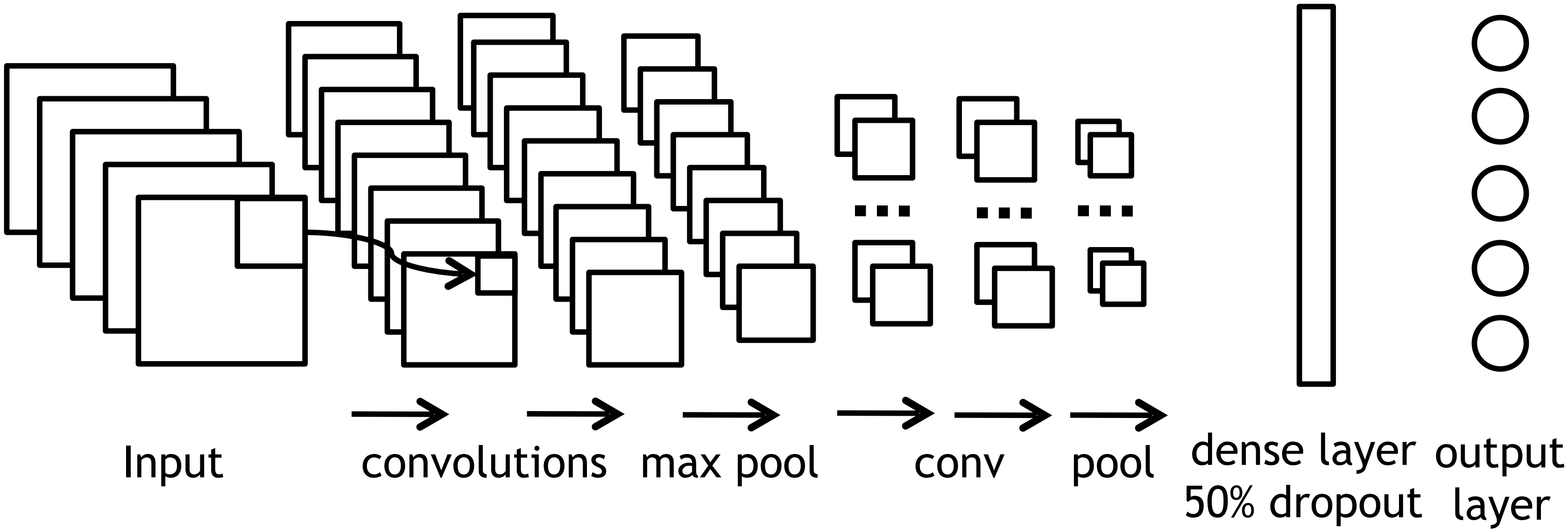
Flush draw

Pair (of Aces)

```
[AhQsAs9s6s9c2s]

x23456789TJQKA
c.......1.....
d.............
h............1
s1...1..1...11
```

Convnet for Texas Hold'em Basics



Win % against random hand

Private cards
Public cards
[No bets]

(6 x 17 x 17 3D tensor)

Probability (category)
- pair, two pairs, flush, etc

(as rectified linear units)

98.5% accuracy, after 10 epochs
(500k Monte Carlo examples)

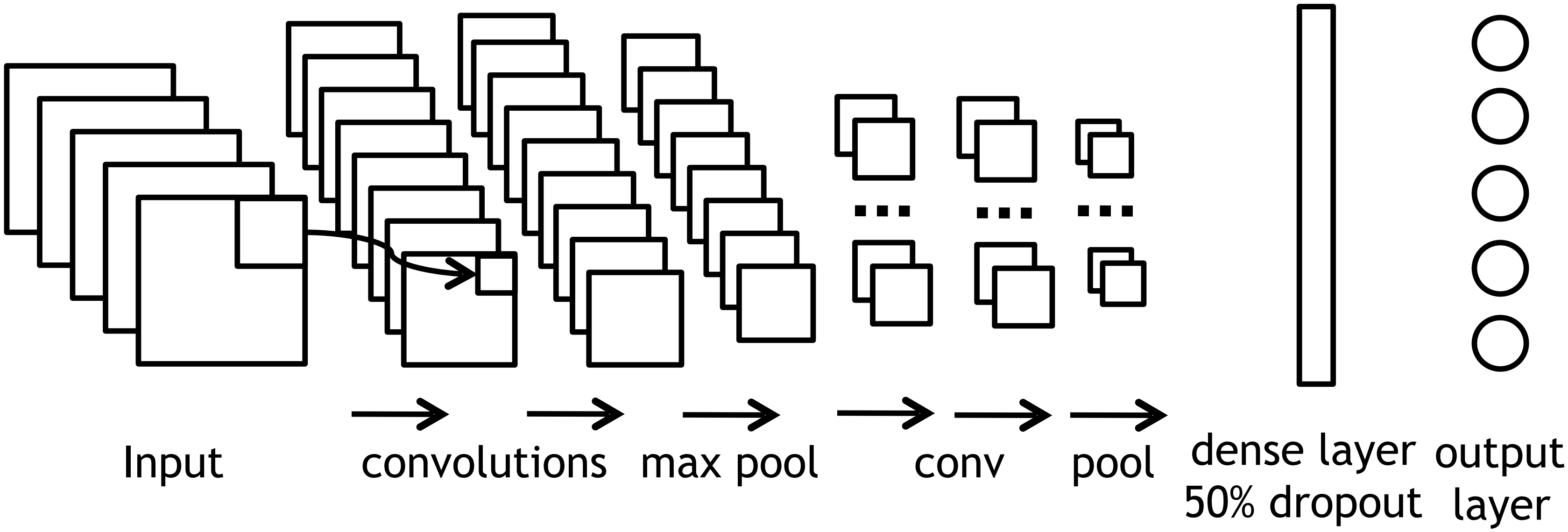
What About the Adversary?

- Our network learned the Texas Hold'em probabilities.
- Can it learn to bet against an opponent?

- Three strategies:
  - Solve for equilibrium in 2-player game
    - [huge state space]
  - Online simulation
    - [exponential complexity]
  - Learn value function over a dataset
    - Expert player games
    - Generated with self-play
    - [over-fitting, unexplored states]

- We take the data-driven approach…

Add Bets to Convnet



- Private cards
- Public cards
- Pot size as numerical encoding
- Position as all-1 or all-0 tensor
- Up to 5 all-1 or all-0 tensors for each previous betting round

(31 x 17 x 17 3D tensor)

Output action value:

- Bet/Raise
- Check/Call
- Fold ($0.0, if allowed)

Masked loss:

- single-trial $ win/loss
- only for action taken (or implied)

That's it?

Table 4: Players' earnings when playing against Poker-CNN in heads up limit Texas Hold'em, with \$50-\$100 blinds. The $\pm$ amount indicates error bars for statistical significance.

Player	Player earnings	# hands
ACPC sample player	-\$90.9 $\pm$7.0	10000
Heuristic player	-\$29.3 $\pm$5.6	10785
CFR-1	-\$93.2 $\pm$7.0	10000
Professional human player	+\$21.1$\pm$30.5	527

- Much better than naïve player models
- Better than heuristic model (based on allin value)
- Competitive with expert human players

What is everyone else doing?

CFR: Equilibrium Approximation

- Counterfactual regret minimization (CFR)
  - Dominant approach in poker research
  - University of Alberta, 2007
  - Used by all Annual Computer Poker Competition (ACPC) winners since 2007
- Optimal solutions for small 1-on-1 games
- Within 1% of unexploitable for 1-on-1 limit Texas Hold'em
- Statistical tie against world-class players
  - 80,000 hands of heads-up no limit Texas Hold'em
- Useful solutions for 3-player, 6-player games

CFR Algorithm

- Start with a balanced strategy.
- Loop over all canonical game states:
  - Compute “regret” for each action by modeling opponent’s optimal response
  - Re-balance player strategy in proportion to “regret”
  - Keep iterating until strategy is stable
- Group game-states into “buckets,” to reduce memory and runtime complexity

Equilibrium vs Convnet

- Visits every state
- Regret for every action
- Optimal opponent response
- Converges to an un-exploitable equilibrium

- Visits states in the data
- Grad on actions taken
- Actual opponent response
- Over-fitting, even with 1M examples
- No explicit balance for overall equilibrium

It's not even close!

But Weaknesses Can Be Strengths

- Visits only states in the data
- Gradient only for actions taken
- Actual opponent response
- Over-fitting, even with 1M examples
- No explicit balance for overall equilibrium

- Usable model for large-state games
- Train on human games without counter-factual
- Optimize strategy for specific opponent
- Distill a network for generalization?
- Unclear how important balance is in practice…

Balance for Game Theory?

- U of Alberta's limit Hold'em CFR within 1% of un-exploitable
- 90%+ of preflop strategies are not stochastic
- Several ACPC winners use "Pure-CFR"
  - Opponent response modeled by single-action strategy

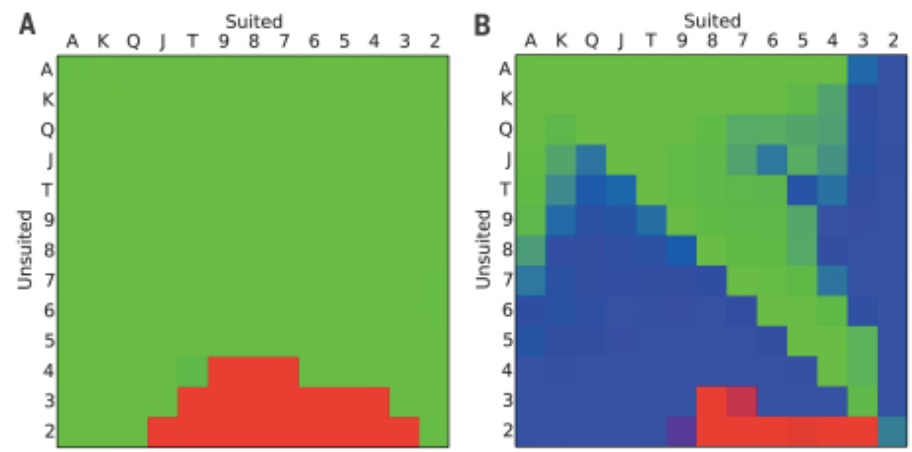


Fig. 4. Action probabilities in the solution strategy for two early decisions. (**A**) The action probabilities for the dealer's first action of the game. (**B**) The action probabilities for the nondealer's first action in the event that the dealer raises. Each cell represents one of the possible 169 hands (i.e., two private cards), with the upper right diagonal consisting of cards with the same suit and the lower left diagonal consisting of cards of different suits. The color of the cell represents the action taken: red for fold, blue for call, and green for raise, with mixtures of colors representing a stochastic decision.

Explore & Exploit for Limit Hold'em

- Sample tail-distribution noise for action values
  - ε * Gumbel
  - Better options?
- We also learn an action-percentage
  - (bet_values) * action_percent / norm(action_percent)
  - 100% single-action in most cases
  - Generalizes more to game context than to specific cards
    - No intuition why
  - Useful for exploration
- Similar cases from other problems??

Observations from Model Evolution

- First iteration of the learned model bluffs like crazy
- Each re-training beats the previous version, but sometimes weaker against older models
  - Over-fitting, or forgetting?
- Difficulty with learning hard truths about extreme cases
  - Can not possibly win, can not possibly lose
  - We are fixing with side-output re-enforcing Hold’em basics
- Extreme rollout variance for single-trial training data
  - Over fitting after ~10 epochs, even with 1M dataset
  - Prevents learning higher-order patterns?

Network Improvements

- Training with cards in canonical form
  - Improves generalization
  - ≈0.15 bets/hand over previous model
- Training with "1% leaky" rectified linear units
  - Released saturation in negative network values
  - ≈0.20 bets/hand over previous model

- Gains are not cumulative

TODO: Improvements

- Things we are not doing…
  - Input normalization
  - Disk-based loading for 10M+ data points per epoch
  - Full automation for batched self-play
  - Database sampling for experience replay
- Reinforcement learning
  - Bet sequences are short, but RL would still help
  - “Optimism in face of uncertainty” – real problem
- RNN for memory…

Memory Units Change the Game?

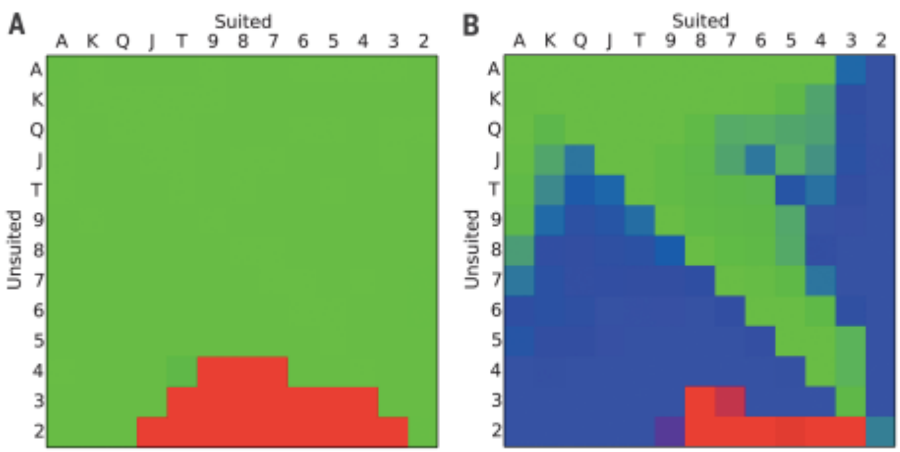


Fig. 4. Action probabilities in the solution strategy for two early decisions. (**A**) The action probabilities for the dealer's first action of the game. (**B**) The action probabilities for the nondealer's first action in the event that the dealer raises. Each cell represents one of the possible 169 hands (i.e., two private cards), with the upper right diagonal consisting of cards with the same suit and the lower left diagonal consisting of cards of different suits. The color of the cell represents the action taken: red for fold, blue for call, and green for raise, with mixtures of colors representing a stochastic decision.

- If opponent called preflop, his hand is in the blue
- If he raised, it is in the green

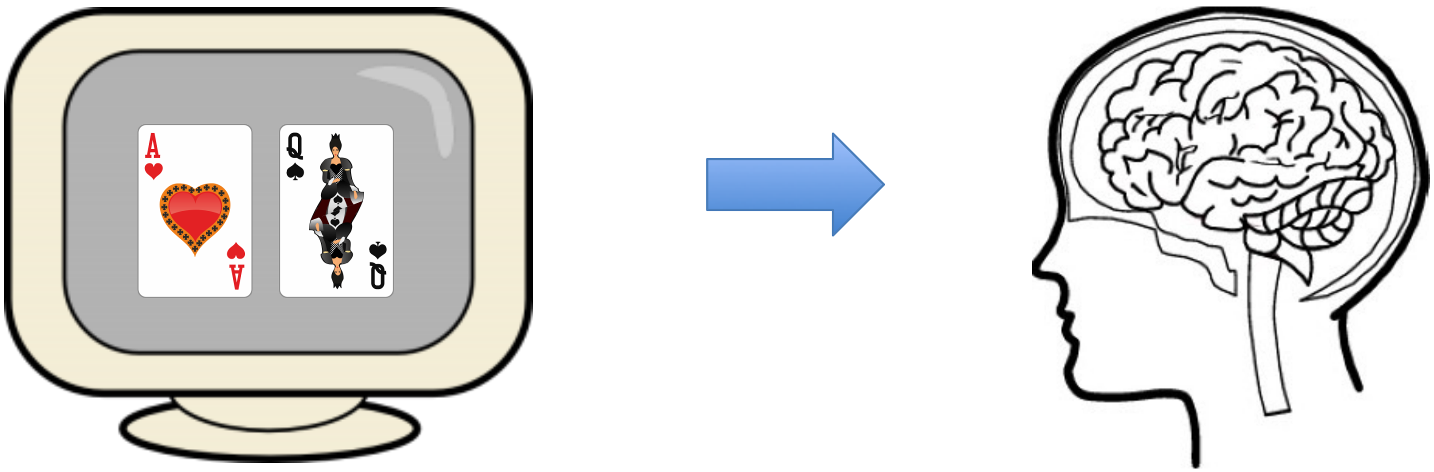
- Use LSTM/GRU memory units to explicitly train for this information?

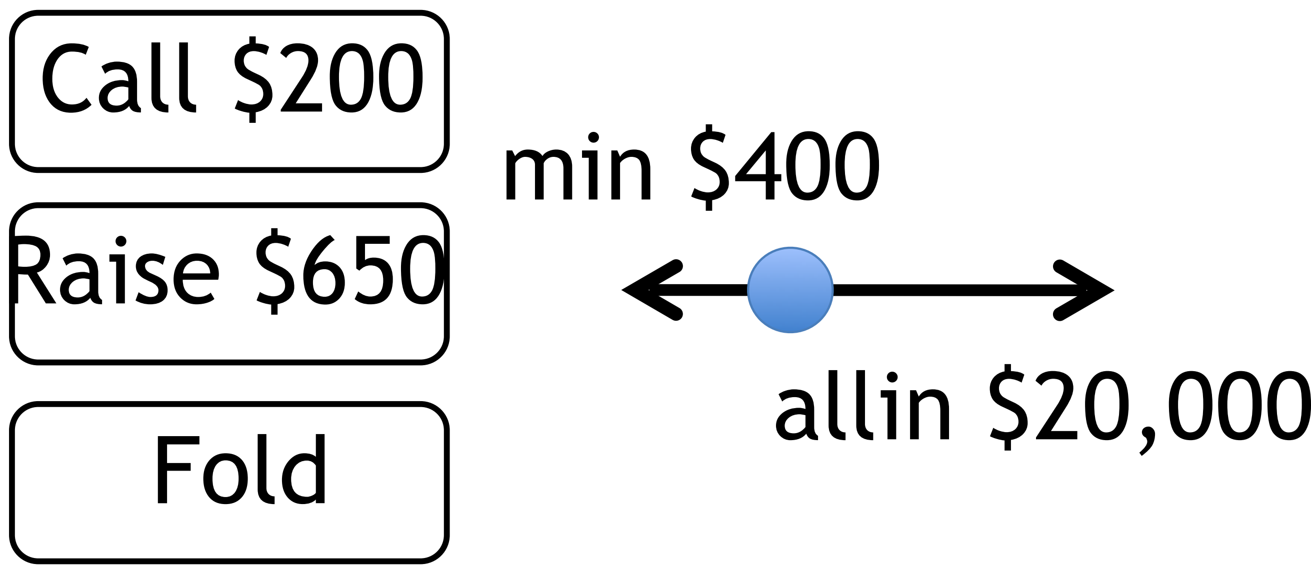
Next: No Limit Texas Hold’em

Take It to the Limit

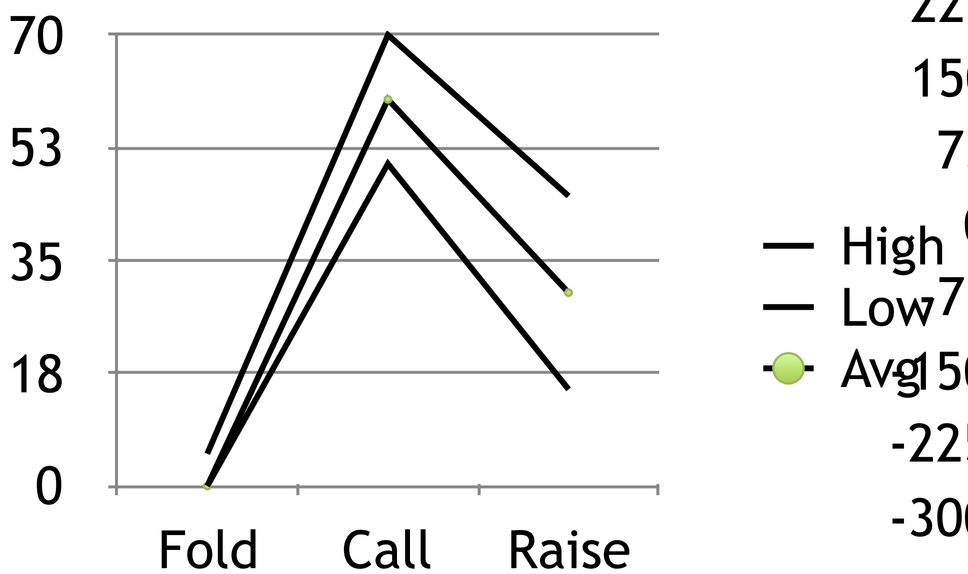
- Vast majority of tournament poker games are no limit Texas Hold’em
- With limit Hold’em “weakly solved,” 2016 ACPC is no limit Hold’em only
- Despite Carnegie Mellon team’s success, no limit Hold’em is not close to a perfect solution

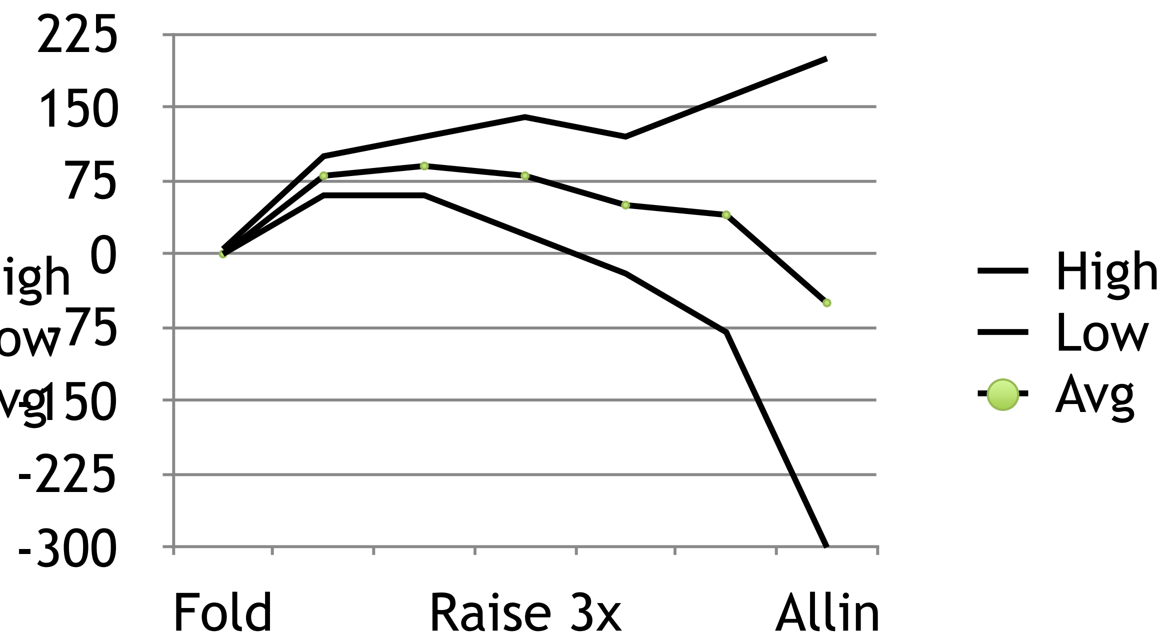
No Limit Hold'em: Variable Betting





From Binary to Continuous Control





Limit Hold’em

No Limit Hold’em

CFR for No Limit Hold'em

- "Buttons" for several fixed bet sizes
  - Fixed at % of chips in the pot
- Linear (or log) interpolation between known states
- Best-response rules assume future bets increase in size, culminating in an allin bet

- Without such rules, response tree traversal is impossible

Call

Raise 2x

Raise 5x

Raise 10x

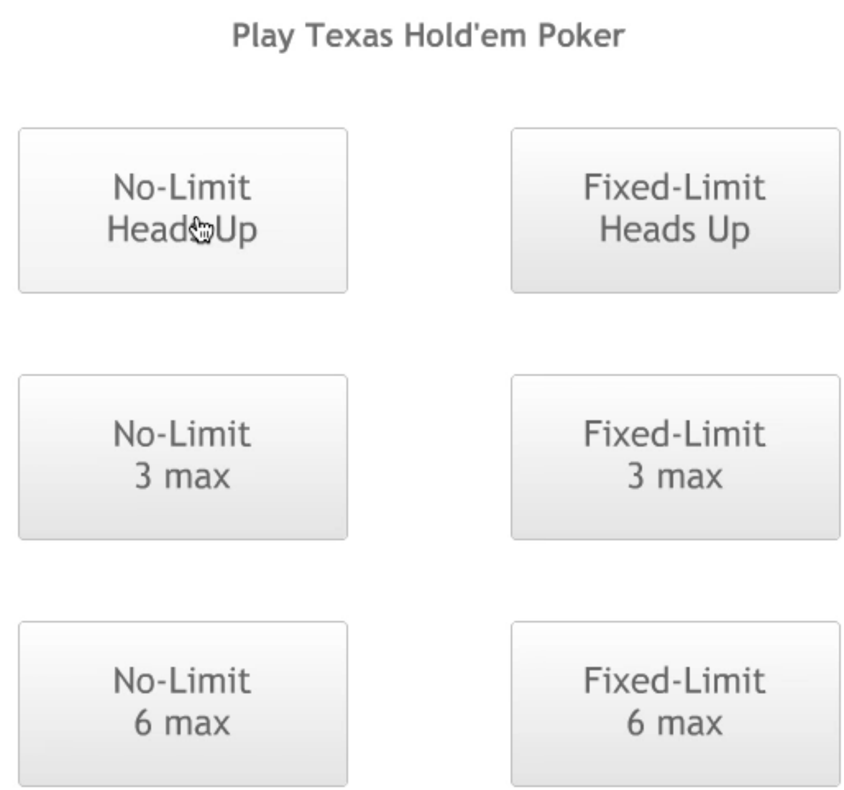
Raise Allin

Fold

CFR for NLH: Observations

- Live demo from 2012-2013 ACPC medal-winner NeoPoker http://www.neopokerbot.com/play
  - It was easy to find “3x bet” strategy that allowed me to win most hands
  - This does not win a lot, but requires no poker knowledge to beat the “approximate equilibrium”
  - Effective at heads-up NLH, 3-player NLH, 6-max NLH

45 hands in 2.5 minutes. I raised 100%



A human would push back…

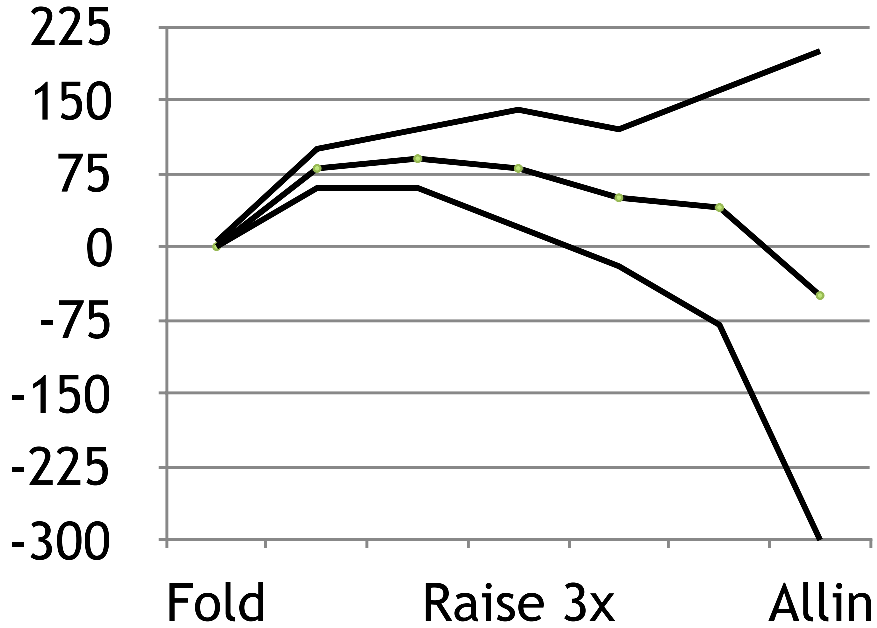
Next Generation CFR

- 2014 ACPC NLH winner Slumbot, based on CFR
- Much harder to beat!
- Better than most human players (including me)
  - 2014 Slumbot +0.12 bets/hand over 1,000+ hands
- Still easy to win 80%+ hands preflop with well-sized aggressive betting
- Why?
  - Game-theory equilibrium does not adjust to opponent
  - Implicit assumptions in opponent response modeling

CFR is an Arms Race

- Slumbot specs (from 2013 AAAI paper)
  - 11 bet-size options for first bet
    - Pot * {0.25, 0.5, 0.75, 1.0, 1.5, 2.0, 4.0, 8.0, 15.0, 25.0, 50.0}
  - 8, 3 and 1 bet-sizes for subsequent bets
  - 5.7 billion information sets
  - 14.5 billion information-set/action pairs
  - Each state sampled with at least 50 run-outs
  - Precise stochastic strategies, for each information set
- Exclusively plays heads-up NLH, resetting to 200 bets after every hand
- 2016 ACPC competition increasing agent disk allotment to 200 GB…

Another Way: Multi-Armed Bandit?



- Beta-distribution for each bucket
- How to update with a convolutional network?

Hack:

- SGD update for Beta mean
- Offline process or global constant for σ

Using Convnet Output for No Limit Betting

- Fold_value = 0.0
- Call_value = network output
- Bet_value = network output
- Can the network estimate a confidence?

If (Bet):

- Sample bet-bucket distributions
- OR
- stats.beta.fit (buckets)
- Fit multinomial distribution to point estimates?
- MAP estimator?
- Ockham's Razor?

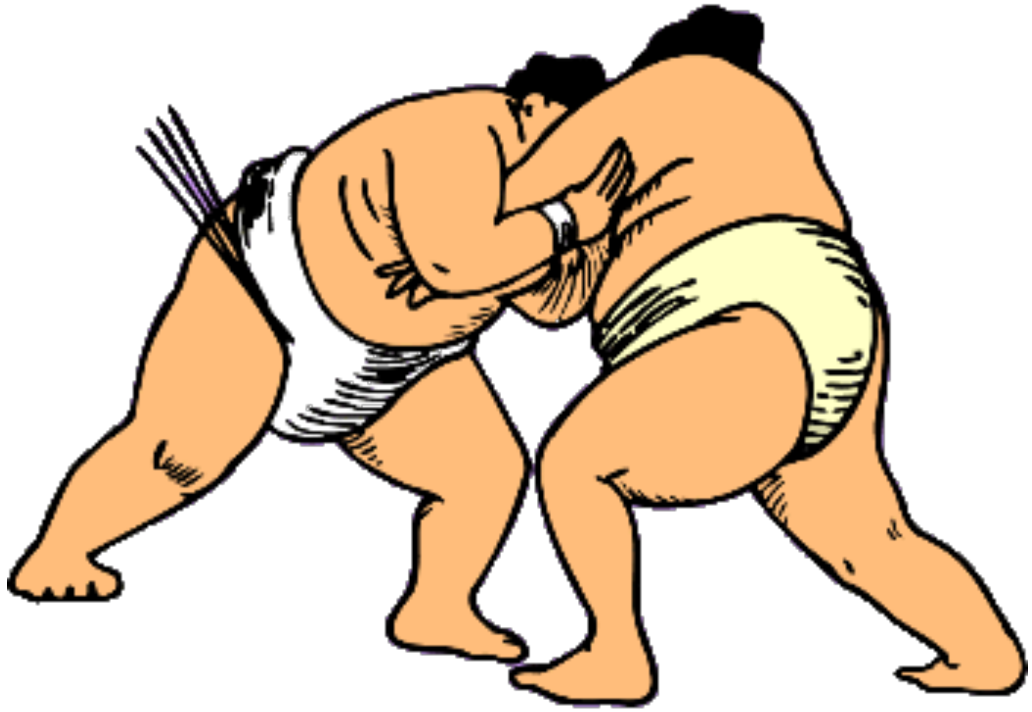
Advantages of Betting with ConvNet

- Forced to generalize from any size dataset
  - CFR requires full traversal, at least once
  - CFR requires defining game-state generalization
- Model can be trained with actual hands
  - Such as last year's ACPC competition
  - Opponent hand histories are not useful for CFR
- Tune-able explore & exploit
- Adaptable to RL with continuous control
  - Learn optimal bet sizes directly

Build ConvNet, then Add Memory

- Intra-hand memory
  - Remember context of previous bets
  - Side-output [win% vs opponent] for visualization

- Inter-hand memory
  - Exploit predictable opponents
  - “Coach” systems for specific opponents
  - Focus on strategies that actually happen



This is a work in progress…

ACPC no limit Hold’em: code due January 2016

Thank you!

Questions?

Citations, Links

- Poker-CNN paper, to appear in AAAI 2016: http://arxiv.org/abs/1509.06731
- Source code (needs a bit of cleanup): https://github.com/moscow25/deep_draw
- Q-Learning for Atari games (DeepMind): http://www.nature.com/nature/journal/v518/n7540/full/nature14236.html
- Counterfactual regret minimization (CFR)
  - Original paper (NIPS 2007) http://webdocs.cs.ualberta.ca/~games/poker/publications/AAMAS13-abstraction.pdf
  - Heads-up Limit Holdem is Solved (within 1%) https://www.sciencemag.org/content/347/6218/145
  - Heads-up No Limit Holdem "statistical tie" vs professional players https://www.cs.cmu.edu/brains-vs-ai
- CFR-based AI agents:
  - NeoPoker, 2012-2013 ACPC medals http://www.neopokerbot.com/play
  - Slumbot, 2014 ACPC winner (AAAI paper) https://www.aaai.org/ocs/index.php/WS/AAAIW13/paper/viewFile/7044/6479