Inverse Reinforcement Learning for Team Sports Valuing Actions and Players

Yudong Luo, Oliver Schulte, Pascal Poupart

SFU SIMON FRASER UNIVERSITY

UNIVERSITY OF WATERLOO

SPORT LOGiQ

IJCAI-PRICAI YOKOHAMA 2020

Background

- Sports Analytics provides professional methods for analyzing sports data to facilitate decision making before and during sports events.
- Focus on evaluating performance (player evaluation):
  1. Use NHL ice hockey data to design model and evaluate
  2. Can easily adapt to similar low scoring sports

Related Work

- Most approaches use the total value of player's actions to rank players, this reduces player evaluation to action evaluation
- State-of-the-art methods use RL to learn an action value Q function:

1. Scoring Impact (SI) [Routley and Schulte, 2015]
   - Used Markov model to model game dynamics
   - Advantage value as impact $\mathrm{impact}(s,a) = Q_{H/A}(s,a) - V_{H/A}(s)$
2. Goal Impact Metric (GIM) [Liu and Schulte, 2018]
   - Used Deep RL to learn Q function
   - Difference between two consecutive Qs as impact $\mathrm{impact}(s,a) = Q_{H/A}(s_t, a_t) - Q_{H/A}(s_{t-1}, a_{t-1})$

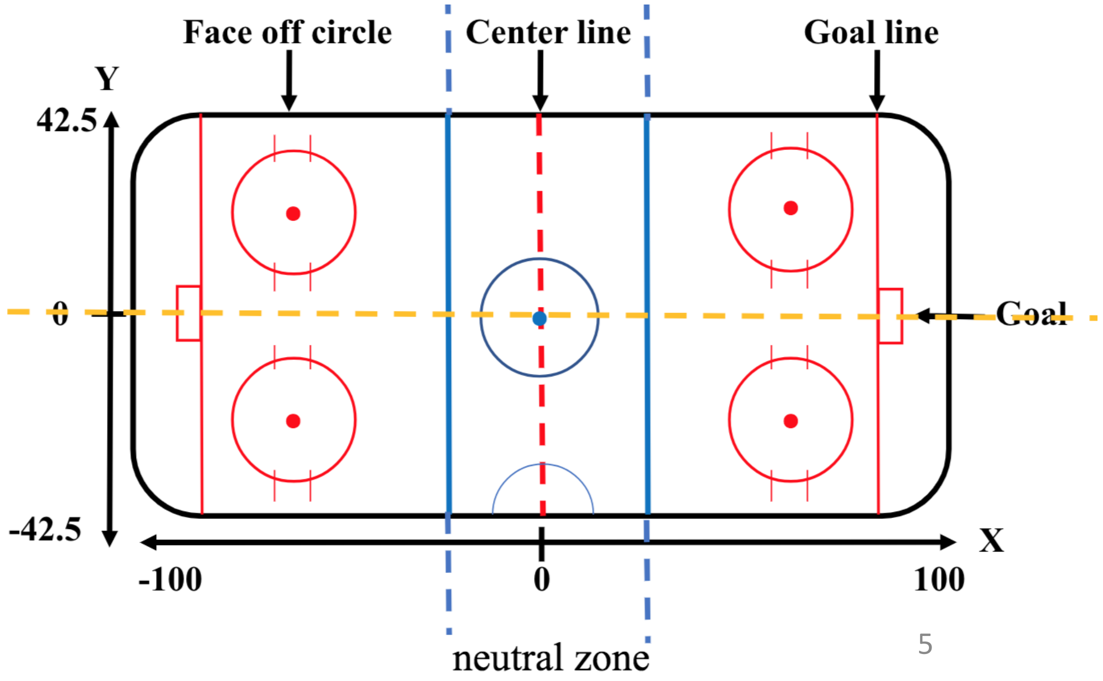
The Score Sparsity Problem

- We notice previous RL models use sparse reward signal
- In low-scoring sports (ice hockey, soccer), explicit values are only attached to rare goal events.
  - Emphasis on goals and related actions (shots, assists)
  - Bias towards offensive players
- Top-50 players for NHL 2018-19 season
  - **SI** : All offensive players
  - **GIM** : Only one Defenceman
- Use Inverse RL to learn reward for game states

Markov Game Model Setup

- Markov Game Model for ice hockey
  - Following **SI**, two agents (H/A), choose defining features as the state
    - Game context: ManPower (MP) : Even strength, Shorthanded, Powerplay
      Goal Diff (GD) : difference between home and away goals
      Period (P) : 1 to 3, do not consider overtime play
    - Team identity: two agents, Home or Away
    - Location (L): divide into 6 regions
  - Transition function calculated using observed frequency

$$T(s,a,s') = \mathrm{p}(s'|s,a) = O(s,a,s')/O(s,a)$$



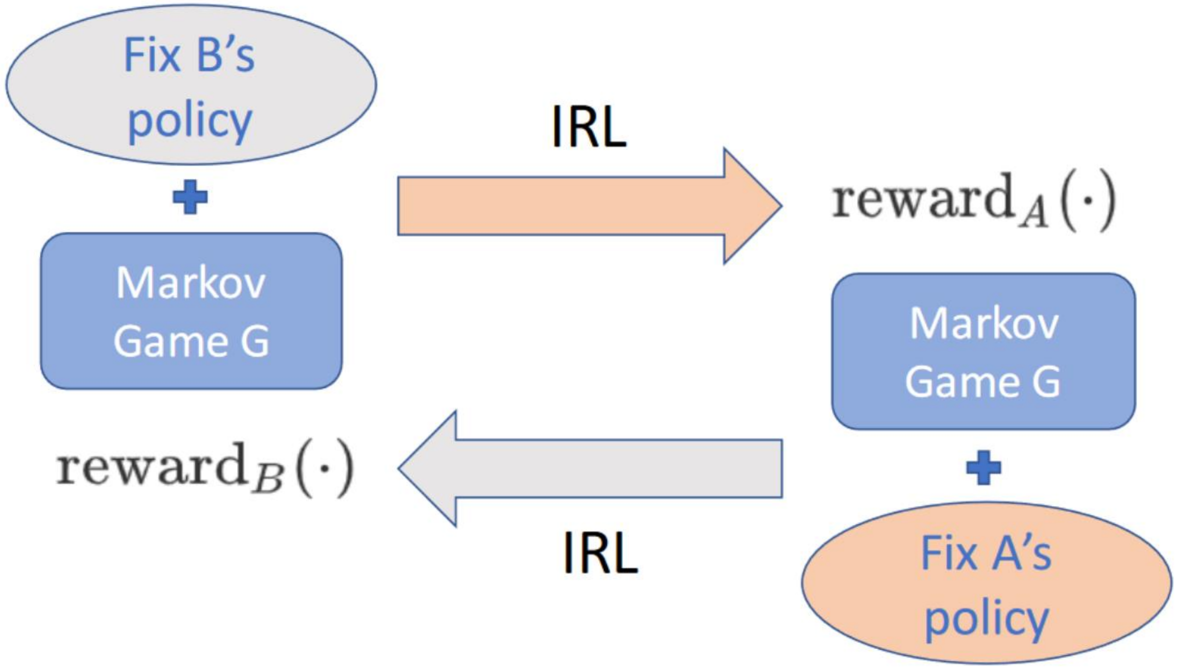
Our approach

- Play-by-play data: only contains info of player who controls the puck
  - Leverage single-agent IRL for multi-agent Markov Game
- Goal is such a rare event in the data
  - Combine knowledge between observed goals and unobserved rewards

Agent	State				Action	Observed Goals	Unobserved Reward	value
team	P	x	y	MP	Event	Score		
16	1	-75.5	-21.5	Even	Check	0	?	?
15	1	-79	-19.5	Even	pass	0	?	?
16	1	-92	-32.5	Even	Lpr	0	?	?
16	1	-92	-32.5	Even	Goal	1	?	?

Alternating IRL

- Treat B as A's environment, learn reward for A using single-agent IRL
- Repeat the procedure with the role of teams A and B reversed

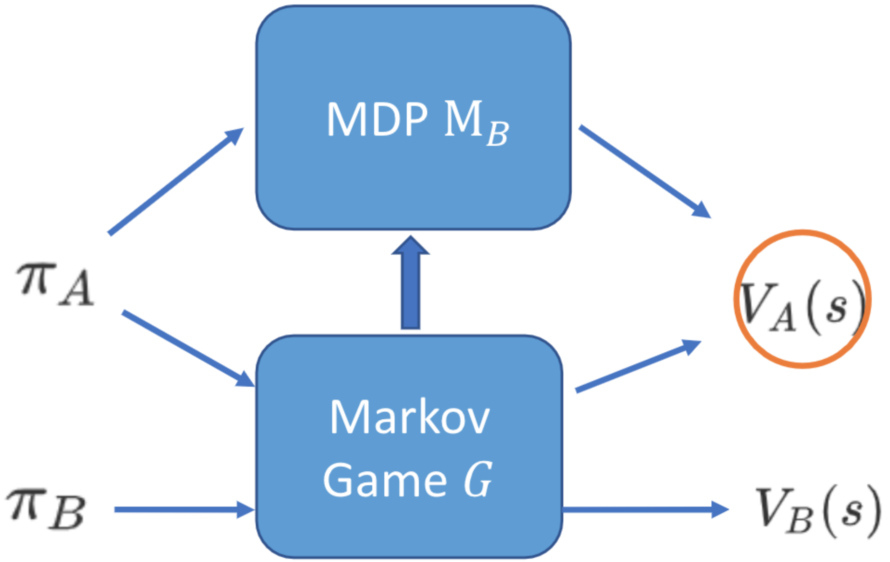


Nash Equilibrium

- Let $\hat{\pi}_A$,$\hat{\pi}_B$ be two policies for agent A and B estimated directly from the data, we assume they satisfy Nash Equilibrium
- Each agent chooses a strategy, and no player can increase its own expected payoff by changing its strategy while the other agents keep theirs unchanged
- Each team optimizes against the observed policies of another team
  - In sports, teams have direct access only to the observed behavior of other teams
  - when an opponent's observed behavior falls shorts of their optimal strategy, successful teams take advantage of it

Transform Multi-agent Model to Single-agent Model

- **Proposition** Consider a two-agent Markov Game model G with two agents A, B, and a policy π_B for agent B. There is a single-agent MDP M_B such that for every policy π_A of agent A, the state value in Markov Game for A equals the state value in MDP.
- Intuition: Single-agent MDP M_B treats B as part of A's environment.



MaxEnt IRL

Maximum Entropy IRL [Ziebart et al, 2008]

- Reward is a linear function of state features, with weights $\boldsymbol{\theta} \in \mathbb{R}^k$
- The reward for a trajectory is the sum of rewards of visited states
- MaxEnt: the likelihood of a trajectory is proportional to exponential reward

$$P(\zeta_i) \propto e^{r_{\zeta_i}}$$

- Maximize the likelihood of trajectories (data) given reward ($\boldsymbol{\theta}$)
- Calculate gradient of likelihood for $\boldsymbol{\theta}$, and update

Combining Observed Goals and Learned Rewards

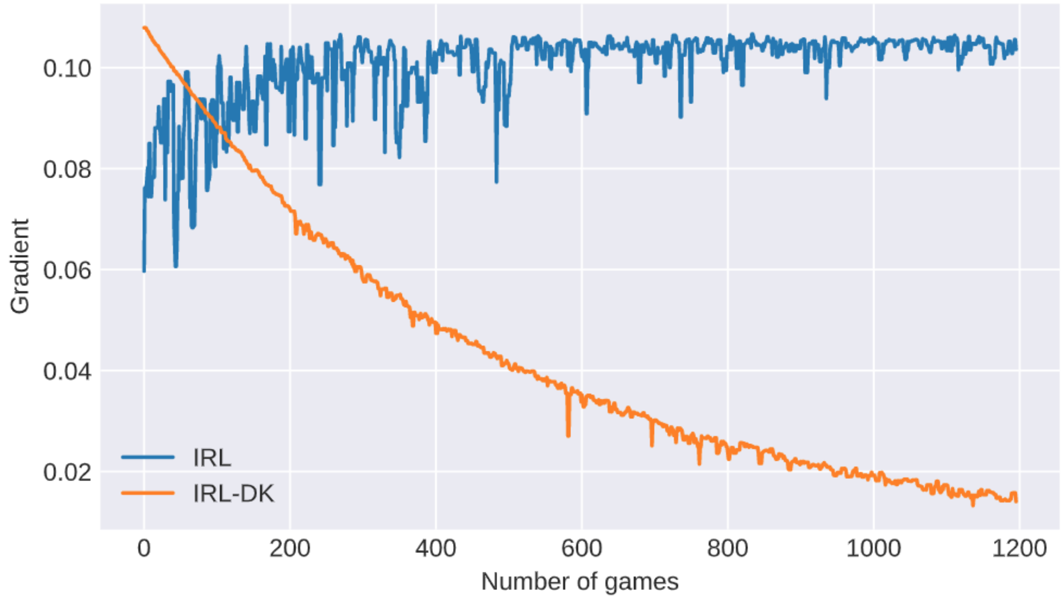
- Choose a kernel function k to measure similarity between observed scores and learned rewards
- Learning procedure maximizes regularized likelihood function

$$\arg\max L(\{\zeta\}|\text{rewards}) + \lambda k(\text{rewards}, \text{goals})$$

- Motivated by maximum mean discrepancy [Gretton et al., 2012] framework for transfer learning
  - Gaussian kernel is usually chosen

Learning Details and Performance

- MaxEnt IRL defines a linear reward function with weight $\boldsymbol{\theta}$
- Define a $\boldsymbol{\theta}_0$ to match goals reward, and initialize $\boldsymbol{\theta}$ with $\boldsymbol{\theta}_0$
- Domain knowledge leads to much more stable and faster convergence



Learned Rewards Solve Sparsity

- Dataset
  - NHL play-by-play dataset from SPORTLOGiQ
  - Game from October 2018 to April 2019

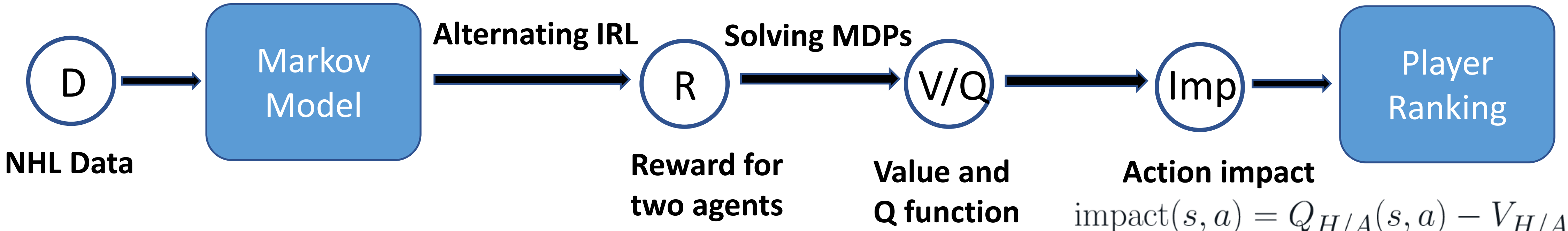
Number of teams	31
Number of players	979
Number of games	1,202
Number of events	4,534,017

- Learned Rewards

Items	STD
Rule reward function (goals)	0.0383
IRL-DK learned reward function	0.1281
Q-values from goals (GIM)	0.0963
Q-values from IRL-DK	1.2207

Player Ranking

- Value/Q function: estimates expected total future reward given current match state
- Use learned reward to calculate value function and Q function for each team (Routley and Schulte, 2015)
- Use value and Q function to assess action impact (Routley and Schulte, 2015; Liu and Schulte, 2018)



$$\text{impact}(s,a) = Q_{H/A}(s,a) - V_{H/A}(s)$$

Player Ranking

- Top-10 offensive and defensive players

Name	Assists	Goals	Points	Team	Salary
Anze Kopitar	38	22	60	LA	11,000,000
Aleksander Barkov	61	35	96	FLA	6,900,000
Dylan Larkin	41	32	73	DET	7,000,000
Nathan Mackinnon	58	41	99	COL	6,750,000
Leon Draisaitl	55	50	105	EDM	9,000,000
Mark Scheifele	46	38	84	WPG	6,750,000
Jonthan Toews	46	35	81	CHI	9,800,000
Connor McDavid	75	41	116	EDM	14,000,000
Jack Eichel	54	28	82	BUF	10,000,000
Ryan O'Reilly	53	30	83	CAR	6,000,000

Table 3: 2018-19 Top-10 offensive players

Name	Assists	Goals	Points	Team	Salary
Drew Doughty	37	8	45	LA	12,000,000
Brent Burns	67	16	83	SJ	10,000,000
Roman Josi	41	15	56	NSH	4,000,000
John Carlson	57	13	70	WSH	12,000,000
Morgan Rielly	52	20	72	TOR	5,000,000
Ryan Suter	40	7	47	MIN	9,000,000
Mark Giordano	57	17	74	CGY	6,750,000
Duncan Keith	34	6	40	CHI	3,500,000
Erik Gustafsson	43	17	60	CHI	1,800,000
Miro Heiskane	21	12	33	DAL	925,000

Table 4: 2018-19 Top-10 defensive players

Started in 2017, Low salary
2019-20 Top-50 Defenceman by NHL

- No obvious bias to player positions (top-50)
  - **SI** : 0 / 50 defensive players
  - **GIM** : 1 / 50 defensive players
  - **Ours** : 32 / 50 defensive players

Correlation with Success Measures

Methods	Assists	GP	Goals	GWG	SHG	PPG	S
+/-	0.269	0.086	0.282	0.278	0.118	0.124	0.156
VAEP	0.215	0.185	0.215	0.089	-0.074	0.160	0.239
WAR	0.591	0.322	0.742	0.571	<u>0.179</u>	<u>0.610</u>	0.576
EG	0.656	0.629	0.633	0.489	0.099	0.391	0.737
SI	0.717	0.633	**0.975**	**0.665**	**0.249**	**0.770**	0.860
GIM	0.757	0.772	0.781	0.518	0.147	0.477	0.795
IRL	0.855	0.872	0.812	0.587	0.123	0.513	0.901
IRL-DK	**0.882**	**0.887**	<u>0.824</u>	<u>0.607</u>	0.125	0.537	**0.907**

Methods	Points	SHP	PPP	FOW	P/GP	SFT/GP	PIM
+/-	0.285	0.179	0.157	0.012	0.306	0.109	0.100
VAEP	0.235	-0.076	0.185	0.021	0.204	0.129	0.172
WAR	0.692	0.147	0.605	0.040	0.699	0.396	0.145
EG	0.694	0.183	0.508	0.254	0.644	0.713	0.355
SI	0.869	0.204	0.708	0.135	0.728	0.639	0.361
GIM	0.818	0.151	0.561	0.289	0.705	0.751	0.372
IRL	0.891	0.207	0.696	0.294	0.741	0.818	0.437
IRL-DK	**0.908**	**0.213**	**0.734**	**0.298**	**0.769**	**0.820**	**0.446**

Table 5: Correlation with success measures (offensive)

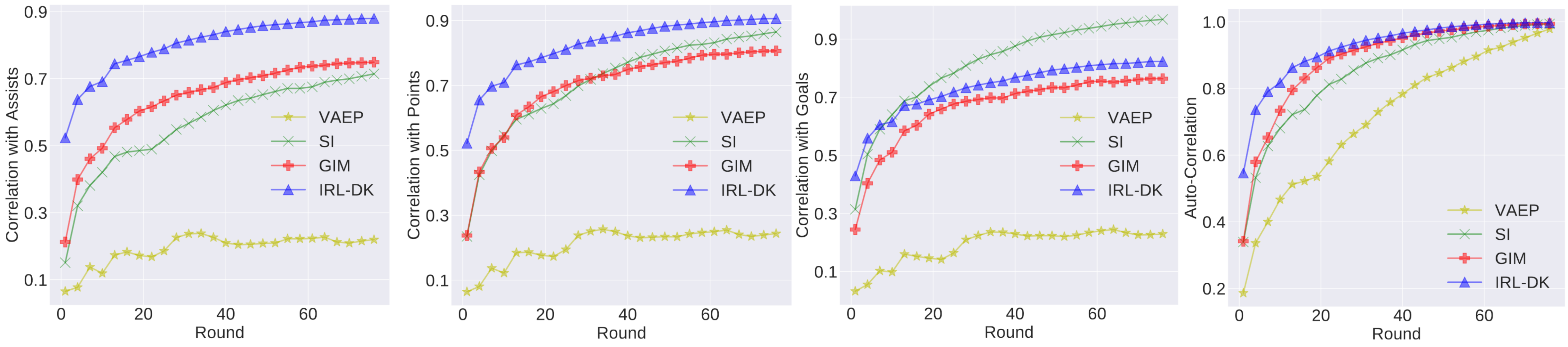
Methods	Assists	GP	Goals	GWG	SHG	PPG	S
+/-	0.173	0.132	0.144	0.177	0.235	-0.116	0.113
VAEP	0.054	-0.045	0.005	0.010	<u>0.384</u>	0.071	-0.016
WAR	0.204	0.028	0.365	0.275	0.097	0.246	0.186
EG	0.589	0.688	0.507	0.321	0.327	0.306	0.679
SI	0.607	0.488	**0.934**	**0.449**	**0.491**	**0.457**	0.709
GIM	0.702	0.862	0.596	0.263	0.130	0.170	0.764
IRL	0.809	0.941	0.686	0.415	0.268	0.347	0.908
IRL-DK	**0.852**	**0.959**	<u>0.701</u>	<u>0.439</u>	0.289	<u>0.360</u>	**0.920**

Methods	Points	SHP	PPP	FOW	P/GP	SFT/GP	PIM
+/-	0.175	0.107	-0.05	0.095	0.169	0.067	0.072
VAEP	0.042	0.065	-0.003	0.101	0.064	-0.036	-0.031
WAR	0.252	0.128	0.266	0.174	0.279	0.006	-0.089
EG	0.611	0.278	0.399	0.118	0.503	0.694	0.360
SI	0.720	0.174	0.488	0.103	0.521	0.499	0.272
GIM	0.730	0.085	0.358	0.140	0.471	0.706	0.438
IRL	0.841	0.281	0.549	0.182	0.557	0.776	0.549
IRL-DK	**0.865**	**0.307**	**0.571**	**0.185**	**0.574**	**0.778**	**0.570**

Table 6: Correlation with success measures (defensive)

Temporal Consistency

- Correlation between **first n rounds** players value and Assists, Points, Goals
- Auto-correlation: **first n rounds** with **entire season** value



Round: players played n games at round n

Summary

- Use inverse reinforcement learning to infer reward for agent that explains its behavior
- Two innovations
  - Alternating learning reduces multi-agent to single-agent IRL
  - Transfer knowledge between observed goals and unobserved rewards
- Learn dense rewards and Q values
- A promising player ranking
  - No obvious bias towards player positions
  - Independent validation through established player metrics

Thank you!