Physics system machine learning application

Yi

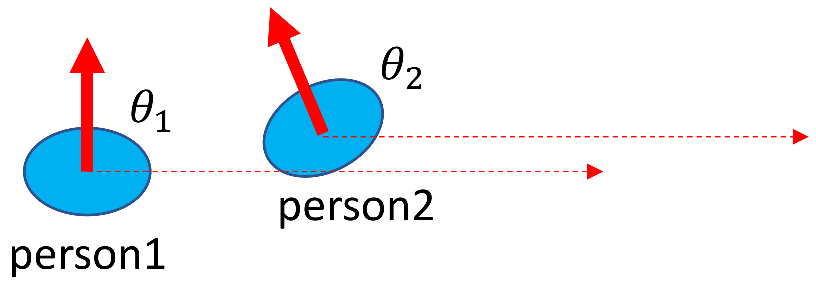
Motivation

- In a small indoor area such as classroom or office, human behaviors are most driven by social interactions (i.e., communication, gestures, eye contacts)
- Those behaviors can be described theoretically if we can come up with a model that can well capture and quantify those abstract social interactions
- The model should be developed based on real observed data
- The best parameters that fit the real data might be obtained if we can implement inverse statistical method (Machine Learning) in some descent manner

Definition of related quantities in the model

The basic unit is a dyad configuration.
Each pair contains 4 quantities

1. Person1 location r_1 *(x1, y1)*
2. Person1 location r_2 *(x2, y2)*
3. Person1 orientation θ_1
4. Person2 orientation θ_2



We propose the interaction model between two pairwise individuals as

Here, K, a, and r_0 are parameters that controls the social interaction

$$V(r_1, r_2, \theta_1, \theta_2) = \mathrm{K}(e^{-2a(|r_1-r_2|-r_0)} - 2e^{-a(|r_1-r_2|-r_0)}) \cos(\theta_1 - \theta_2)$$

*derivation of this expression is bit tedious and irrelevant to ML topic, One can look at end of the slides to find more details if interested

Equilibrium probability distribution of observations

Particle system (refers to individual human in this problem) can be represented by the *Hamiltonian of such system.* Here is just the summation of interaction overall all pairwise configurations

$$H = -\sum_{i,j}^{N} \mathrm{K}\left(e^{-2a\left(|r_i - r_j| - r_0\right)} - 2e^{-a\left(|r_i - r_j| - r_0\right)}\right)\cos(\theta_i - \theta_j)\,,$$

Boltzmann equilibrium probability distribution of observations (locations, orientations denoted by X in general) is given by

$$p(X) = \frac{1}{Z} e^{-H}$$

where Z is a normalizing factor (called partition function in physics or statistics) (defined as $Z = \sum_s e^{-H_s}$, summing e^{-H} over all possible equilibrium states, just ignor physics here)

Maximum likelihood

The equilibrium probability distribution of above system is conditional on parameter set $\Theta \sim (D, a, r_0)$

$$p(X|\Theta) = \frac{1}{Z(\Theta)} e^{-H(X,\Theta)}$$

Z should not be dependent on a specific observation X

The posterior distribution given by Bayes theory

$$p(\Theta|X) = \frac{p(\Theta, X)}{p(X)} = \frac{p(X|\Theta)p(\Theta)}{p(X)}$$

Since we have no prior knowledge of the parameter value, Θ is uniformly distributed.

$$p(\Theta|X) \propto p(X|\Theta)$$

The maximum likelihood estimator will find the parameters maximizing $p(\Theta|X)$

Now our purpose is to maximize $p(X|\Theta) = \frac{1}{Z(\Theta)} e^{-H(X,\Theta)}$

Taking more convenient form

$$L^D(X,\Theta) = -\log\left(p(X|\Theta)\right)$$
$$= H^D(\mathrm{X},\Theta) + \ln\left(Z(\Theta)\right)$$

Minimize L^D instead now

$$\frac{\partial L^D(X,\Theta)}{\partial\Theta} = < \frac{\partial H^D(X,\Theta)}{\partial\Theta} > - < \frac{\partial H(\Theta)}{\partial\Theta} >$$

Second term has no D superscript, meaning this is general system expression only parametrized by (Θ)

At the minimum of the log-likelihood those derivatives are zero; the maximum-likelihood estimate of the parameters is reached when the expectation values of observation under the Boltzmann statistics match their sample averages

Note: superscript D denotes data, $\ln(Z(\Theta))$ here serves as an average generator, for instance,

$$-\frac{\partial \ln(Z(\Theta))}{\partial\Theta} = < \frac{\partial H}{\partial\Theta} >$$

Boltzmann machine learning

The log-likelihood turns out to be a convex function of the model parameters. It can be minimized by a convex optimization algorithm, here we can use gradient-descent algorithm

The update for parameters are according to

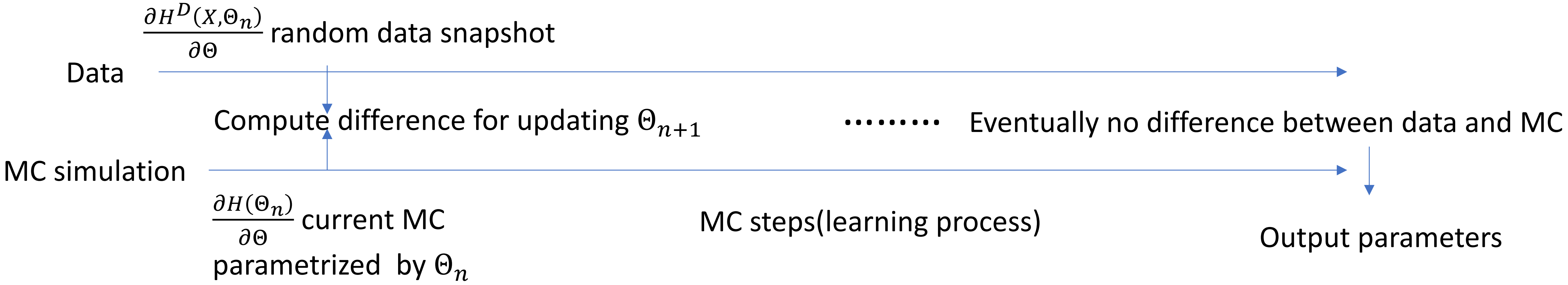
$$\Theta_{n+1} = \Theta_n - \alpha \frac{\partial L^D(X, \Theta_n)}{\partial \Theta} = \Theta_n - \alpha\left(< \frac{\partial H^D(X, \Theta_n)}{\partial \Theta} > - < \frac{\partial H(\Theta_n)}{\partial \Theta} >\right)$$

NOTE: the first expectation term on the RHS takes average of finite real data set, while second term is true expectation value of the distribution parametrized by Θ_n. It is infeasible in practice because it requires parametrized ensemble average for each update. Even through we could implement Monte Carlo simulation to get some averages, it is still too computationally expensive.

Stochastic gradient-descent

$$\Theta_{n+1} = \Theta_n - \alpha(< \frac{\partial H^D(X, \Theta_n)}{\partial \Theta} > - < \frac{\partial H(\Theta_n)}{\partial \Theta} >)$$

To compute first expectation term, we instead pick only one snapshot of the system out of the whole data set.
Besides, we could simultaneously run a Monte Carlo simulation taking current Θ_n as parameters, and using current snapshot of running MC to compute second expectation term



Learning results

First, use MC simulation to generate a set of “fake” data, the parameters of such system are chosen by ourself, meaning they are known

$\frac{\partial H^D(X,\Theta_n)}{\partial \Theta}$ random data snapshot

”fake” Data $\longrightarrow$

Compute difference for updating Θ_{n+1} ……… Eventually no difference between data and MC

MC simulation $\longrightarrow$

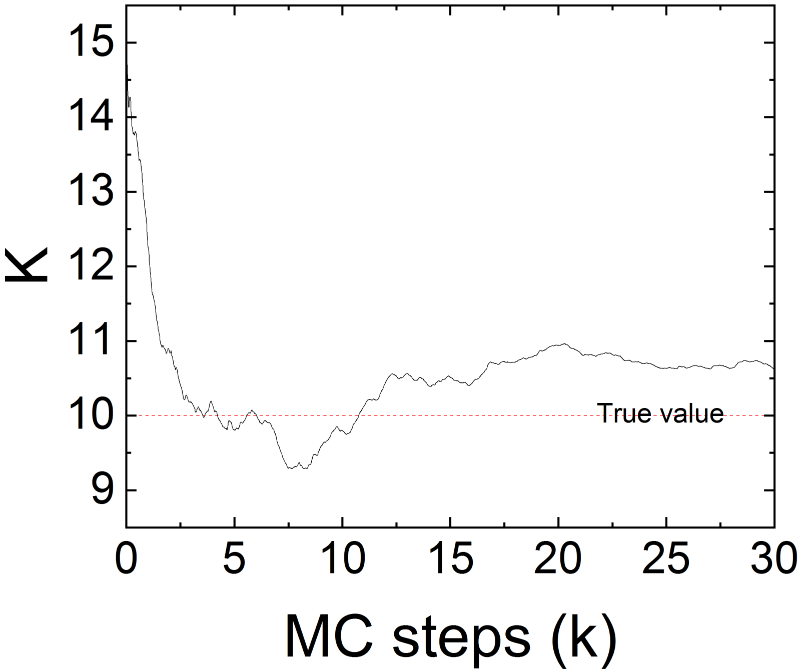
$\frac{\partial H(\Theta_n)}{\partial \Theta}$ current MC parametrized by Θ_n

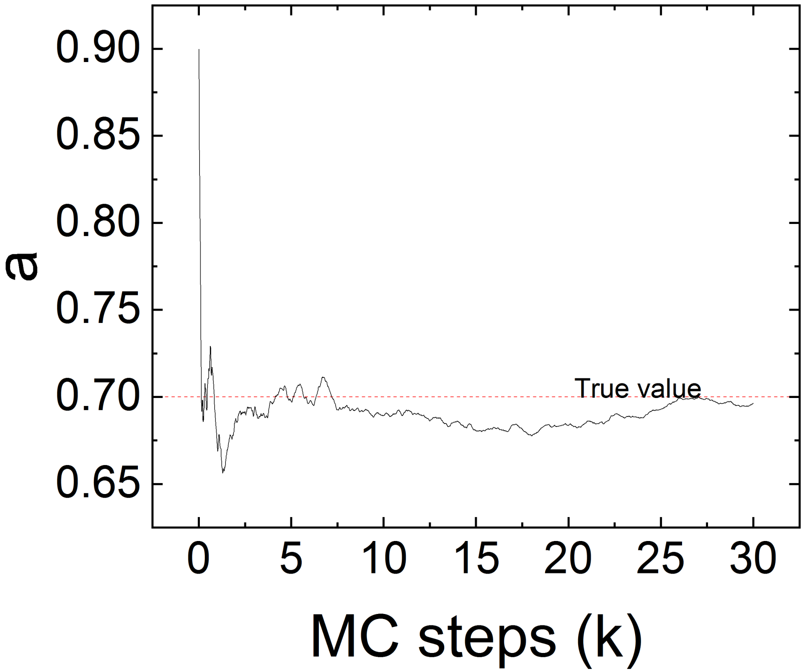
MC steps(learning process)

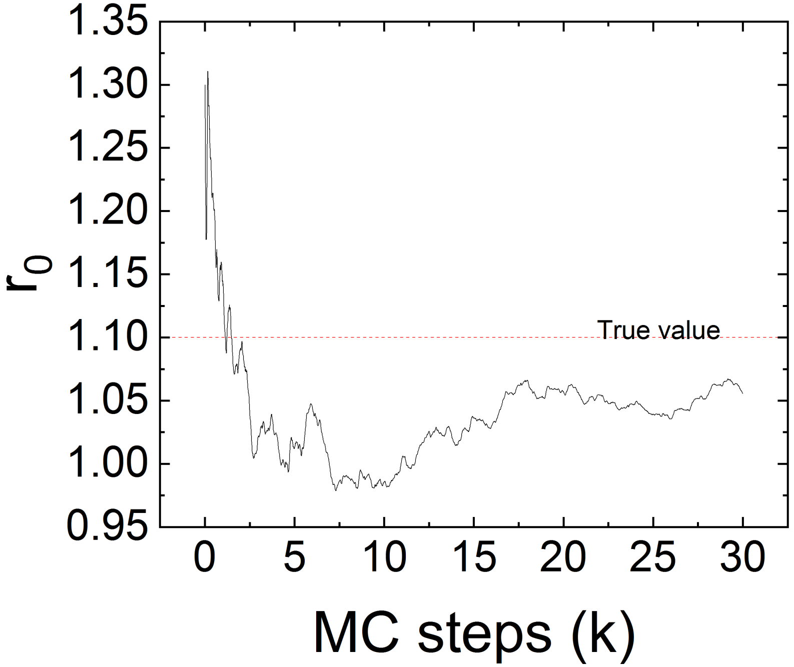
Output parameters and compare with the true value

NOTE: in practice, we need to consider the real physics, human can't be 0 distance, so we treat particles as soft ball (soft sphere potential, see http://www.sklogwiki.org/SklogWiki/index.php/Soft_sphere_potential), simply avoid the scenario that two particles occupy the same location

”Fake” data learning

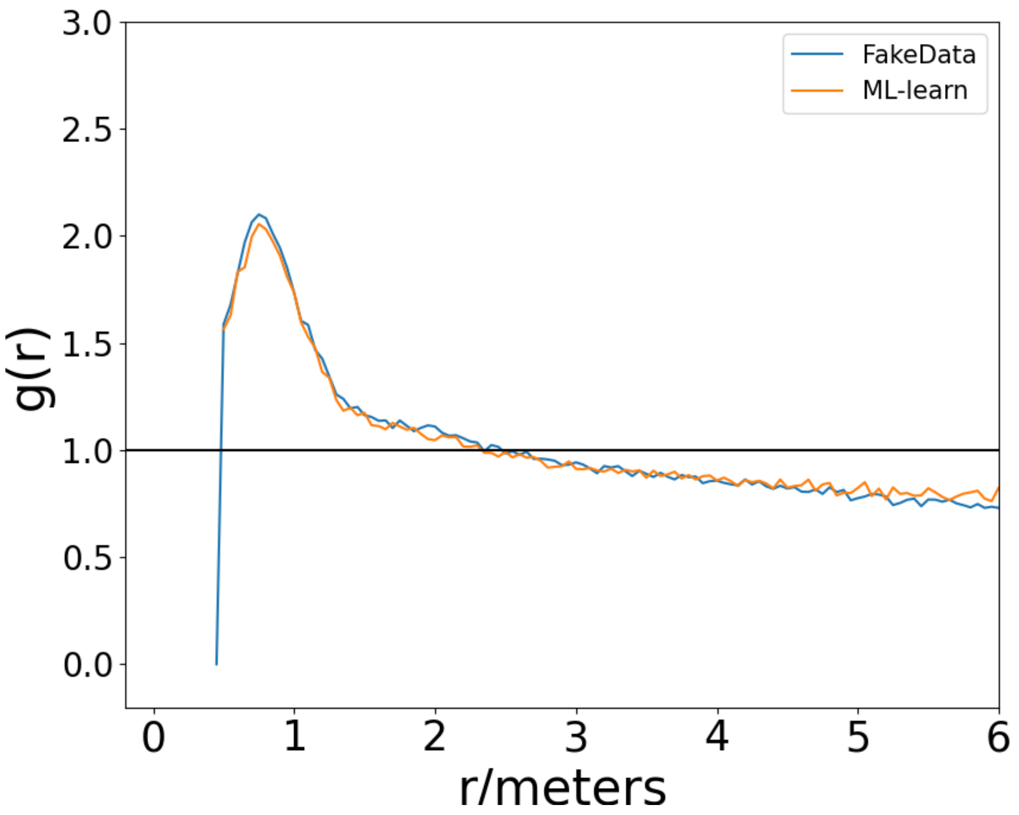






An example

Measurement of radial distribution function g(r)*



In ”fake” data learning, the machine has successfully learned this feather from “fake” data

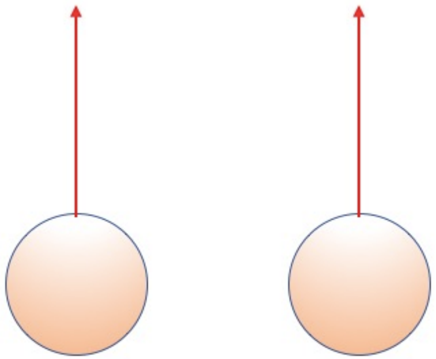
** g(r) measures the probability of distance r roughly, see more* https://en.wikipedia.org/wiki/Radial_distribution_function

Following slides are rough derivations of interaction, might NOT be relevant to ML topic

Angular correlation vs. social radii

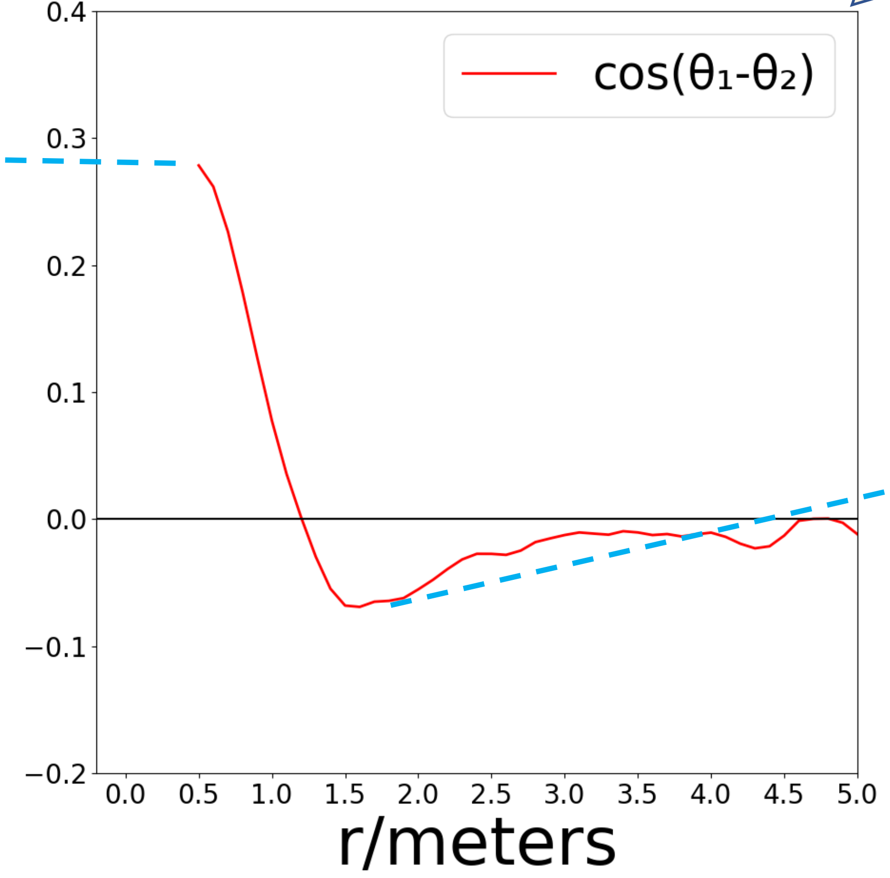
$$Parallel \xrightarrow{as\ r\ increases} Face\ to\ face$$

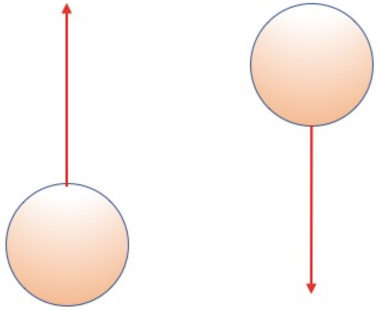
- Plot $\cos(\theta_1 - \theta_2)\ vs. r$



$\theta_1 - \theta_2 = 0\ (parallel)$

$\cos(\theta_1 - \theta_2) = 1$





$\theta_1 - \theta_2 = \pm\pi\ (face\ to\ face)$

$\cos(\theta_1 - \theta_2) = -1$

Guess for interaction expression

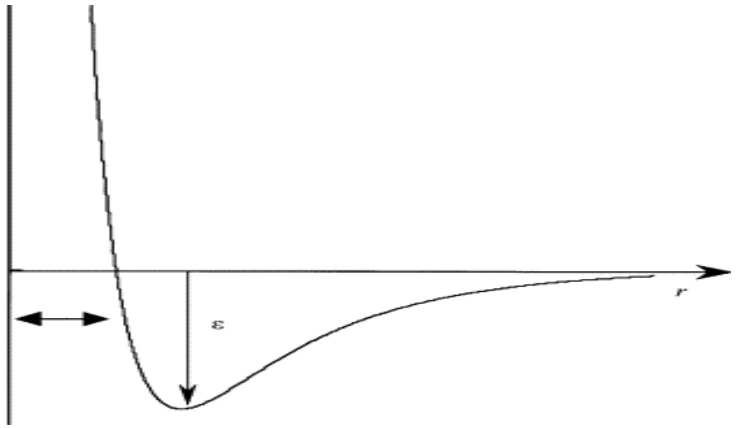
- Summarize what we've learned so for…
  - ✓At a short r range, children tend to be parallel(synchronized)

    → large positive K (~ferromagnetic)

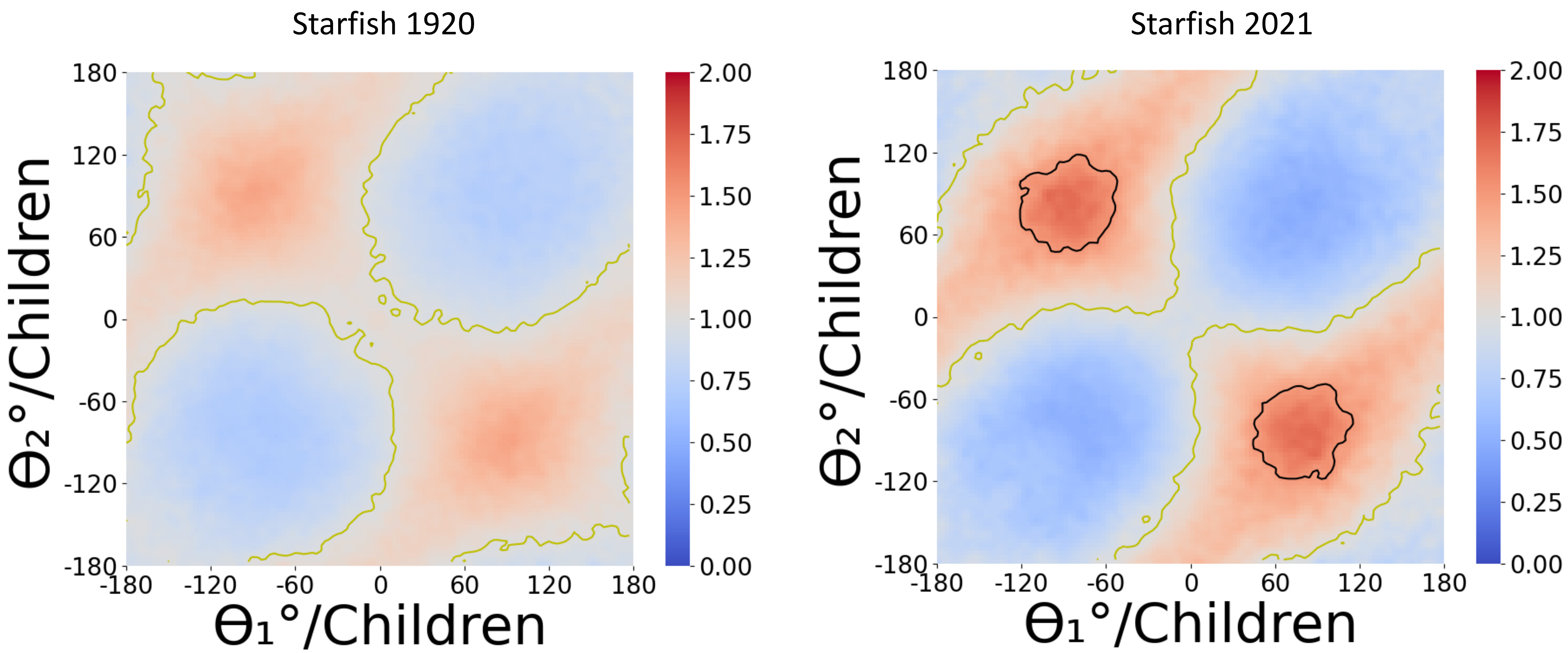
  - ✓As r increasing, children have tendency of turning to face to face

    → negative K (~antiferromagnetic)

Leonard-Jones potential? Or Morse potential?

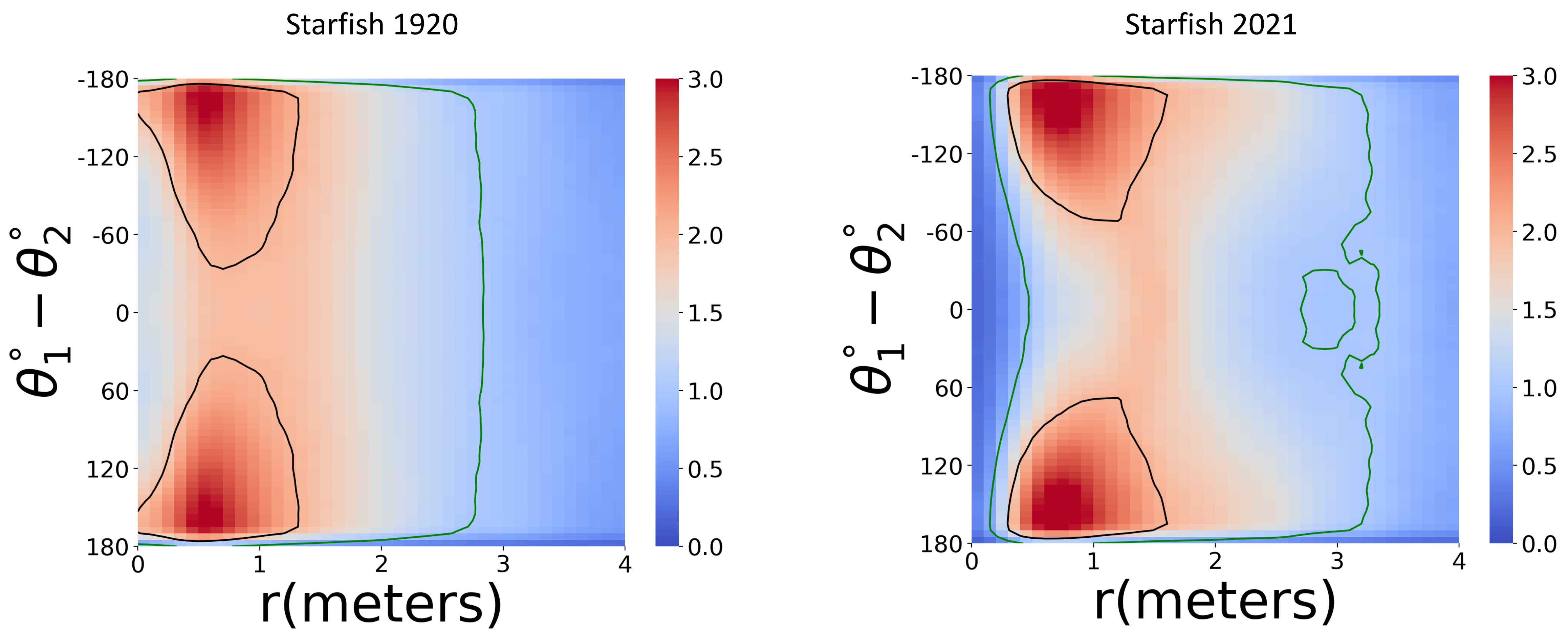


heatmap observations before and after pandemic



1. Kids tend to have more shoulder-to-shoulder interaction
2. Kids tend to have less face-to-face interactions

g(r, theta) observations before and after pandemic



Kids tend to interact shoulder-to-shoulder at a relative larger distance