



# LEARNING DRIVER PREFERENCES FOR FREEWAY MERGING USING MULTITASK IRL

Sanath Bhat

THINC Lab

Department of Computer Science, UGA



# OUTLINE

- Introduction
  - Problem Description and Motivation
  - Background
  - Contributions
  - Related Work
- Data and Setup
  - NGSIM
  - Metadata Description
  - MDP Definition
  - Trajectory Extraction

## OUTLINE (Contd...)

- Task Separation
  - Split-Merge Clustering
  - Results
- Likelihood Weighting Multitask IRL
  - Hierarchical Bayesian Multitask Model
  - Choice of priors
  - Likelihood Weighting Multitask IRL Algorithm
  - Results
- Conclusion and Future Work



# INTRODUCTION



# AUTONOMOUS CARS

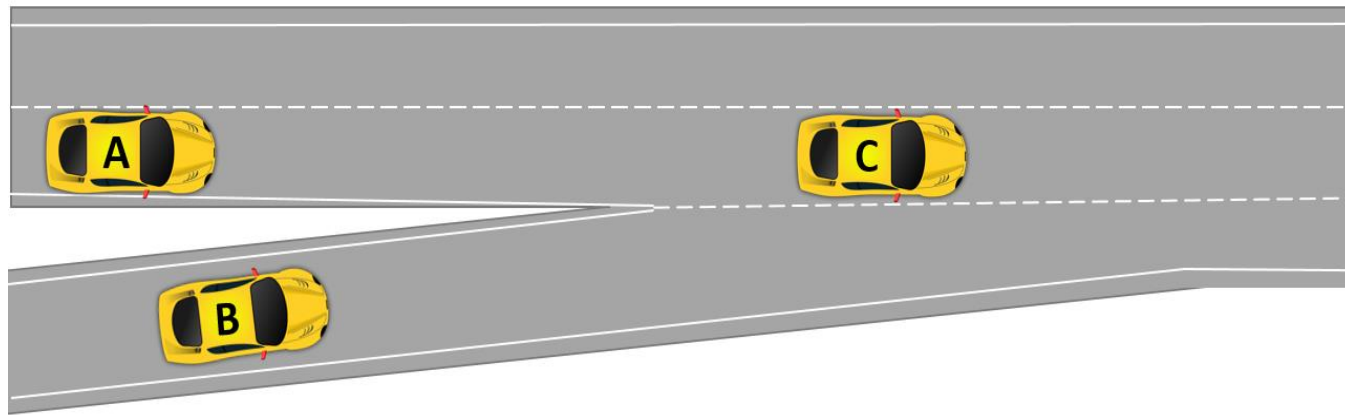


# PROBLEM DESCRIPTION AND MOTIVATION

- Sub-problems of autonomous car-decision making
  - Pathing
  - Collision detection
  - Following relevant signage
  - Parking
  - Passing
  - Lane-Merging ←



# PROBLEM DESCRIPTION AND MOTIVATION



- Car A accelerates
- Car A brakes
- Car A ??

# BACKGROUND : MARKOV DECISION PROCESS AND RELATED CONCEPTS

- MDP - 5-tuple  $\{S, A, T, R, \gamma\}$ :
  - State set,  $S : \{S_0, S_1, \dots, S_{|S|}\}$
  - Action set,  $A : \{A_0, A_1, \dots, A_{|A|}\}$
  - Transition Function,  $T(s' | s, a) : S \times A \times S \rightarrow [0, 1]$ ,  $\sum_{s'} T(s' | s, a) = 1$  for each  $(s, a)$
  - Reward Function,  $R(s, a) : S \times A \rightarrow \mathbb{R}$
  - Discount factor,  $\gamma \in [0, 1]$
- Other related:
  - Policy,  $\pi(s) : S \rightarrow A$  OR  $\pi(s, a) : S \times A \rightarrow [0, 1]$ ,  $\sum_a \pi(a | s) = 1$  for each  $s$
  - Value Function,  $V(s) : S \rightarrow \mathbb{R}$
  - Q-function,  $Q(s, a) : S \times A \rightarrow \mathbb{R}$



# BACKGROUND : MARKOV DECISION PROCESS & RELATED CONCEPTS (Contd...)

- Value function

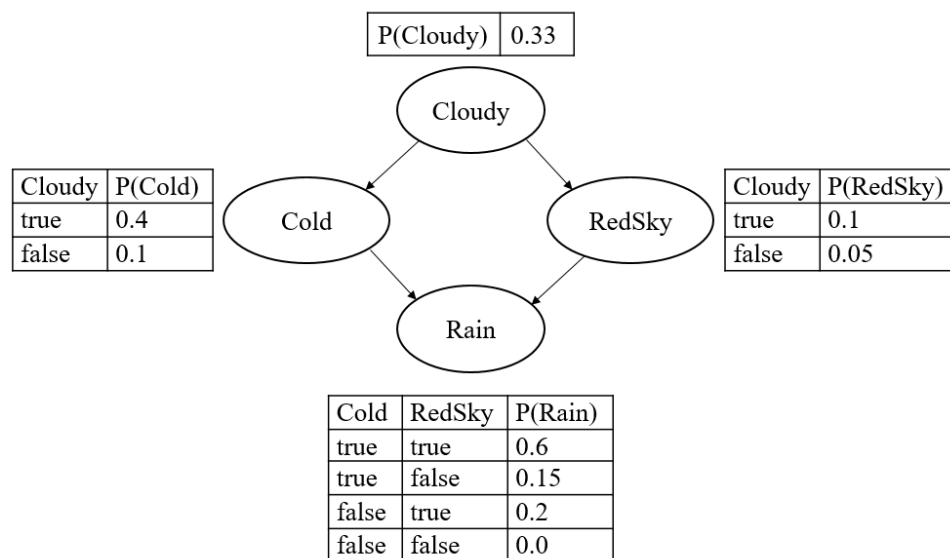
$$V(s) = \max_{a \in A} \left( R(s, a) + \gamma \sum_{s'} T(s, a, s') * V(s') \right)$$

- Value Iteration, Policy Iteration

- Q-function using converged Value function

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} T(s, a, s') * V(s')$$

# BACKGROUND : LIKELIHOOD WEIGHTING



$P(\text{RedSky} \mid \text{Cloudy} = \text{true}, \text{Rain} = \text{false}) = ?$

1. Sample weight  $w = 1$ .
2.  $\text{Cloudy} = \text{true}$  is evidence  $\Rightarrow w = w \times P(\text{Cloudy} = \text{true}) = 0.33$
3.  $\text{Cold} \sim P(\text{Cold} \mid \text{Cloudy} = \text{true})$ . Suppose we get true.
4.  $\text{RedSky} \sim P(\text{RedSky} \mid \text{Cloudy} = \text{true})$ . Suppose we get false.
5.  $\text{Rain} = \text{true}$  is evidence  $\Rightarrow w = w \times P(\text{Rain} = \text{false} \mid \text{Cold} = \text{true}, \text{RedSky} = \text{false}) = 0.33 \times 0.85 = 0.2805$
6. Tabulate as  $W[\text{RedSky} = \text{false}] += 0.2805$

**function** LIKELIHOOD-WEIGHTING( $X, \mathbf{e}, bn, N$ ) **returns** an estimate of  $\mathbf{P}(X \mid \mathbf{e})$

**inputs:**  $X$ , the query variable

$\mathbf{e}$ , observed values for variables  $\mathbf{E}$

$bn$ , a Bayesian network specifying joint distribution  $\mathbf{P}(X_1, \dots, X_n)$

$N$ , the total number of samples to be generated

**local variables:**  $\mathbf{W}$ , a vector of weighted counts for each value of  $X$ , initially zero

**for**  $j = 1$  to  $N$  **do**

$\mathbf{x}, w \leftarrow \text{WEIGHTED-SAMPLE}(bn, \mathbf{e})$

$\mathbf{W}[x] \leftarrow \mathbf{W}[x] + w$  where  $x$  is the value of  $X$  in  $\mathbf{x}$

**return** NORMALIZE( $\mathbf{W}$ )

**function** WEIGHTED-SAMPLE( $bn, \mathbf{e}$ ) **returns** an event and a weight

$w \leftarrow 1$ ;  $\mathbf{x} \leftarrow$  an event with  $n$  elements initialized from  $\mathbf{e}$

**foreach** variable  $X_i$  in  $X_1, \dots, X_n$  **do**

**if**  $X_i$  is an evidence variable with value  $x_i$  in  $\mathbf{e}$

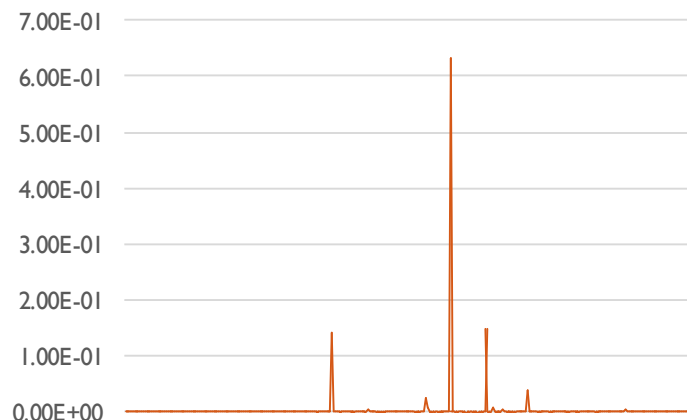
**then**  $w \leftarrow w \times P(X_i = x_i \mid \text{parents}(X_i))$

**else**  $\mathbf{x}[i] \leftarrow$  a random sample from  $\mathbf{P}(X_i \mid \text{parents}(X_i))$

**return**  $\mathbf{x}, w$

## BACKGROUND : LIKELIHOOD WEIGHTING (Contd...)

- Problems with Likelihood Weighting
  - Fares poorly with large number of evidence variables
  - Worsens if evidence occurs late in the network



# BACKGROUND : HIERARCHICAL BAYESIAN MODELING

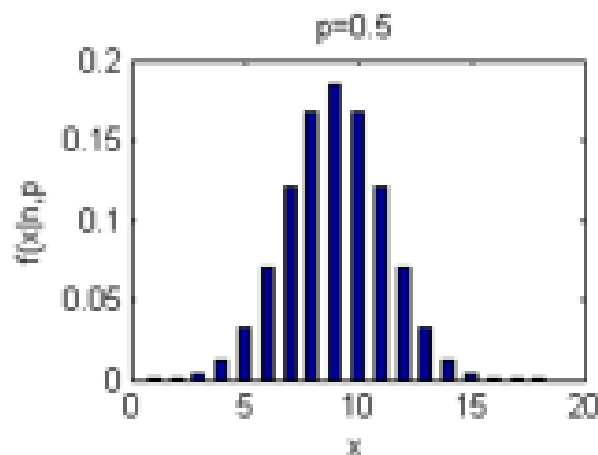
Bayesian Modeling the distribution of the number of heads when  $n$  coins are tossed

Problem: Determine  $p$  given various  $x$  obtained by using some coin

Model

- $X \sim \text{Bin}(n, p) = \text{Bin}(20, 0.5)$

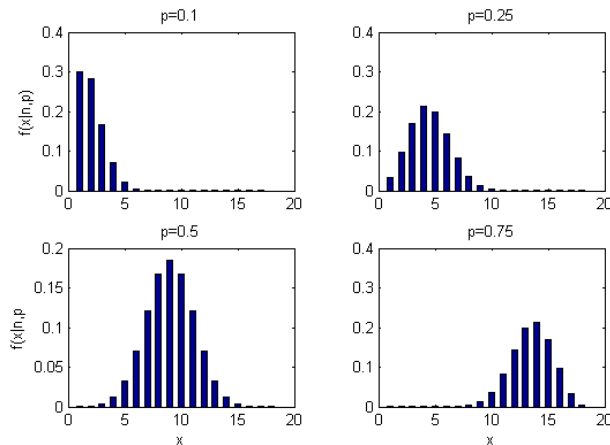
i.e.  $p=0.5$  (a fixed value)



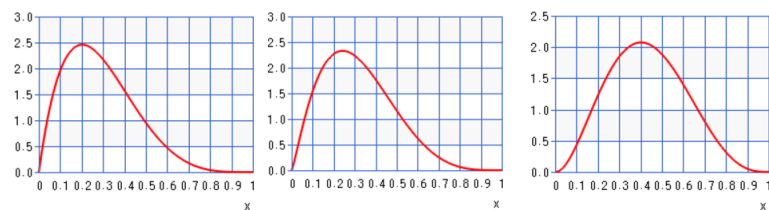
# BACKGROUND : HIERARCHICAL BAYESIAN MODELING

Hierarchical Bayesian Modeling the distribution of the number of heads when n coins are tossed

- $X \sim \text{Bin}(n, p) = \text{Bin}(20, p)$



$p \sim \text{Beta}(\alpha, \beta)$  (a parameter)



$\alpha \sim \mathcal{N}(2, 1)$ ,  $\beta \sim \mathcal{N}(5, 2)$  (hyperparameters)



# BACKGROUND : HIERARCHICAL BAYESIAN MODELING

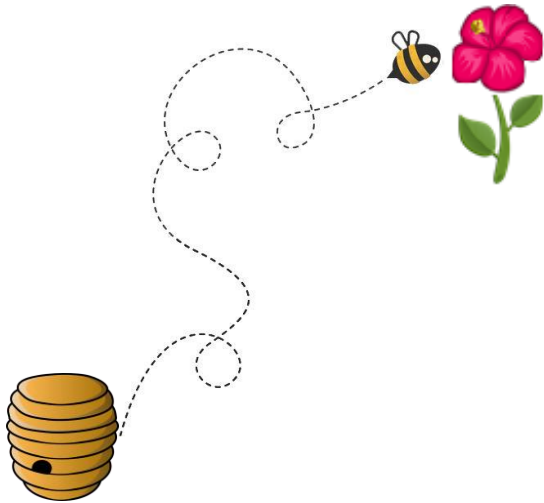
- Hierarchical Model

- Level 1-Evidence:  $X | p \sim \text{Bin}(100, p)$
- Level 2-Prior:  $p | \gamma \sim \text{Beta}(\alpha, \beta)$  ...  $\gamma = (\alpha, \beta)$
- Level 3-Hyperprior:  $\gamma \sim \mathcal{N}(2, 1) \times \mathcal{N}(5, 2)$
- (Can have more levels! E.g.. Prior over Gaussian parameters of hyperpriors)

- Joint posterior given by

$$P(\gamma, p | X) \propto P(X | p, \gamma) P(p, \gamma) P(\gamma)$$

# BACKGROUND : INVERSE REINFORCEMENT LEARNING



I thought the shortest  
distance from A to B is a  
straight line

Shut up and watch!  
The bee knows something  
you don't!



# BACKGROUND : INVERSE REINFORCEMENT LEARNING

- Noteworthy mentions
  - Maximum Entropy IRL, Ziebart et al.
  - Bayesian IRL – Ramachandran et al.
  - Non-parametric Bayesian IRL for multiple reward functions, Choi et al.
  - Bayesian Multitask IRL, Dimitrakakis et al.



# CONTRIBUTIONS

- Establish that a driver has different motivations when passing through different sections of a **merging zone**
- Hypothesize the existence of an ‘**average behavior**’ motivated by **multiple reward functions** for all drivers passing through the merging zone
- To split the merging task into multiple tasks each with its own reward functions and policies differing from that of others.
- To determine the **individual reward functions and policies** of the average driver in each task.



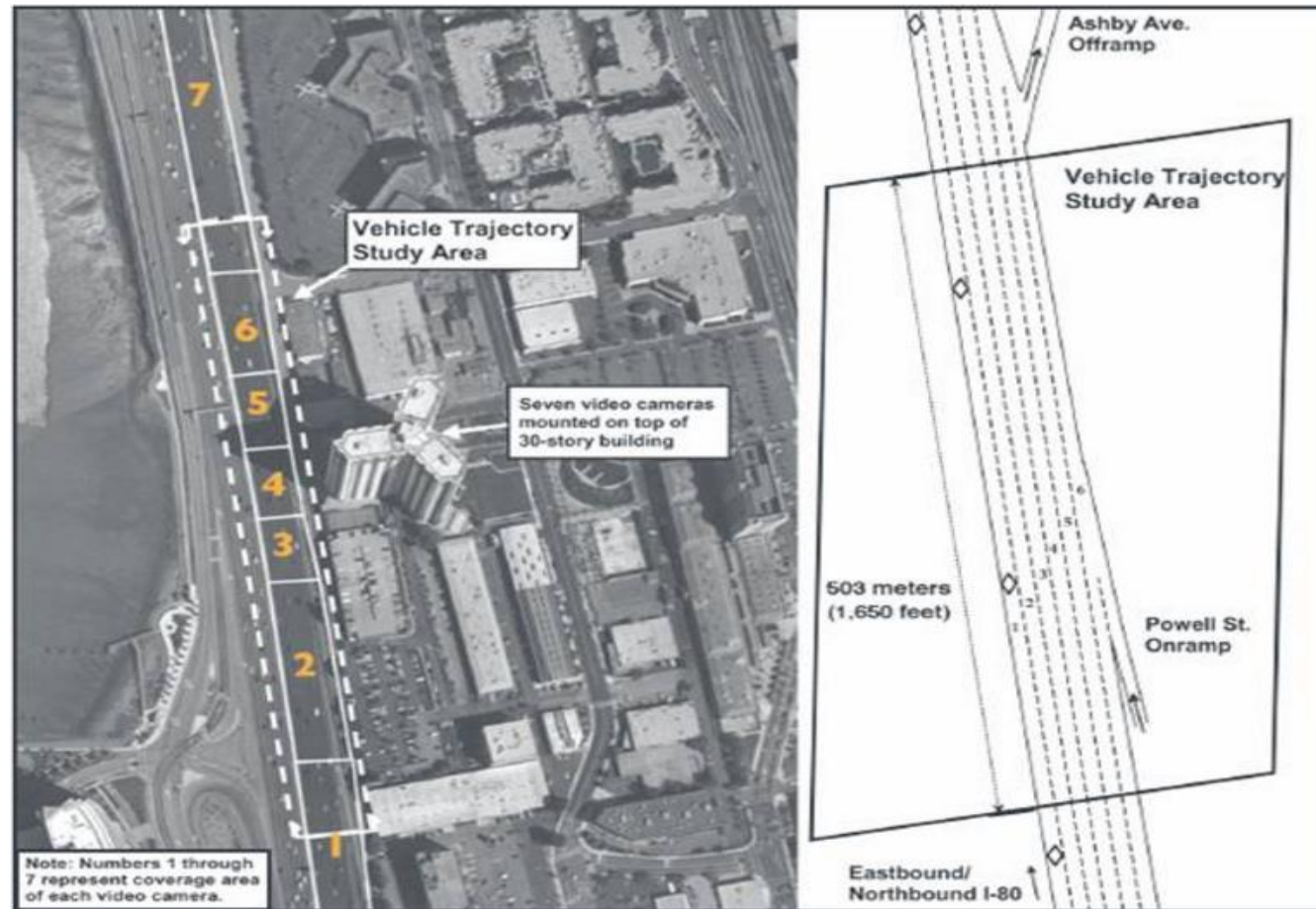
# DATA AND SETUP



# THE NGSIM PROGRAM

- Developed a core of open behavioral algorithms to support traffic simulation and microscopic modeling of vehicles and their interactions
- I80 Dataset – one of the datasets collected under this program. Dataset contains vehicle trajectory data.
  - Where: Eastbound I-80 in the SFO Bay Area in Emeryville, CA
  - When: April 13, 2005
  - Span: Over a length of 1650 feet including an acceleration ramp
  - How: Using 7 synchronized cameras mounted on a 30 story building
  - Duration: Three 15 minute intervals
    - 4.00 p.m. - 4.15 p.m.
    - 5.00 p.m. - 5.15 p.m.
    - 5.15 p.m. - 5.30 p.m.
  - Type of Data: Video + Transcribed csv format

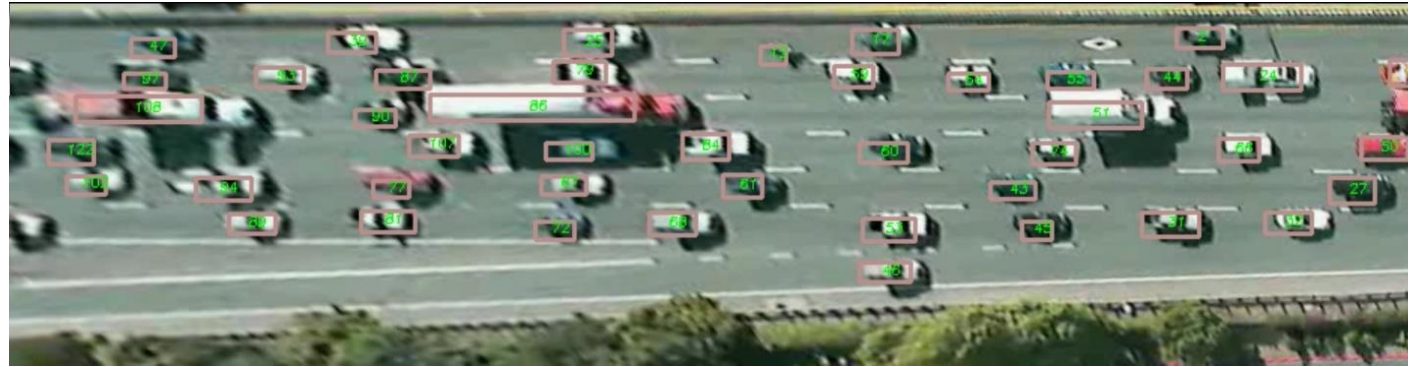
# THE NGSIM PROGRAM



# THE NGSIM PROGRAM



0

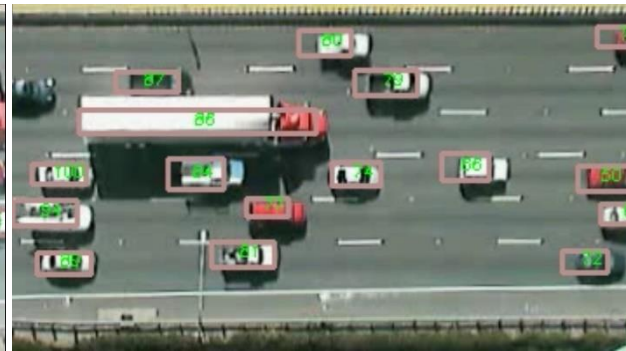


~250

~714



~715



~900

~1090

# METADATA DESCRIPTION : NGSIM DATASET

- A collection of 3 CSV files each representing the 3 time intervals
- Each row has instantaneous information about a vehicle in one frame represented in 18 columns
- The most important columns that are of our concern are as follows

Column Field	Description	Column Field	Description
Vehicle ID	unique ID of current vehicle	Vehicle acceleration	instantaneous acceleration(feet/s <sup>2</sup> )
Frame ID	frame numbers increment by one every 100ms	Lane identification	current lane # of the vehicle
Local Y	the longitudinal coordinate of the front center of the vehicle	Leading Vehicle ID	ID of the leading vehicle
Vehicle length	measured in feet	Following Vehicle ID	ID of the following vehicle
Vehicle velocity	instantaneous velocity (feet/sec)	Spacing	the distance to the front bumper of the preceding vehicle

# TRAJECTORY DATA EXTRACTION : OVERVIEW



$d_1$  \_\_\_\_\_  $((s_0, a_0), (s_1, a_1), \dots, (s_t, a_t))$   
 $d_2$  \_\_\_\_\_  $\cdot$   
\_\_\_\_\_  $\cdot$   
 $d_N$  \_\_\_\_\_  $\cdot$



$\cdot$   
 $\cdot$   
 $\cdot$   
 $\cdot$   
 $\cdot$   
 $\cdot$   
 $\cdot$



$\cdot$   
 $\cdot$   
 $\cdot$   
 $\cdot$   
 $\cdot$   
 $\cdot$   
 $\cdot$

# TRAJECTORY DATA EXTRACTION : MDP MODEL

- MDP model of car A
- State Features:
- Spacing ( $x_{AC}$ ): Distance between the front bumper of car A and the rear bumper of the leading car i.e. car C.

$$x_{AC} = \text{Spacing in dataset} - \text{length of leading car}$$

- Relative velocity ( $v_{AC}$ ): Augment the original dataset with another column - instantaneous velocity of the leading car. Then,

$$v_{AC} = \text{velocity(of car A)} - \text{velocity of leading car(augmented column)}$$



# TRAJECTORY DATA EXTRACTION : MDP MODEL

- MDP model of car A
- State features:
  - Spacing ( $x_{AC}$ ): Distance between the front bumper of car A and the rear bumper of the leading car i.e. car C.

$$x_{AC} = \text{Spacing in dataset} - \text{length of leading car}$$

- Relative velocity ( $v_{AC}$ ): Augment the original dataset with another column - instantaneous velocity of the leading car. Then,

$$v_{AC} = \text{velocity(of car A)} - \text{velocity of leading car(augmented column)}$$

- Action features:
  - Acceleration( $a$ )

$x_{AC}$ (feet)	$v_{AC}$ (feet/second)	$a$ (feet/s <sup>2</sup> )
< 0	> 45	< -9
0 – 14	35 – 45	-9 – -4.8
14 – 28	25 – 35	-4.8 – -0.6
28 – 42	15 – 25	-0.6 – 0.6
42 – 56	5 – 15	0.6 – 4.8
56 – 70	-5 – 5	4.8 – 9
70 – 84	-15 – -5	> 9
84 – 98	-25 – -15	
98 – 112	-35 – -25	
112 – 126	-45 – -35	
126 – 140	< -45	
140 – 168		
168 – 196		

# TRAJECTORY DATA EXTRACTION : MDP MODEL

$v_{AC}$	> 45	35 – 45	25 – 35	15 – 25	5 – 15	-5 – 5	-15 – -5	-25 – -15	-35 – -25	-45 – -35	< -45
$x_{AC}$											
< 0	0	1	2	3	4	5	6	7	8	9	10
0 – 14	11	12	13	14	15	16	17	18	19	20	21
14 – 28	22	23	24	25	26	27	28	29	30	31	32
28 – 42	33	34	35	36	37	38	39	40	41	42	43
42 – 56	44	45	46	47	48	49	50	51	52	53	54
56 – 70	55	56	57	58	59	60	61	62	63	64	65
70 – 84	66	67	68	69	70	71	72	73	74	75	76
84 – 98	77	78	79	80	81	82	83	84	85	86	87
98 – 112	88	89	90	91	92	93	94	95	96	97	98
112 – 126	99	100	101	102	103	104	105	106	107	108	109
126 – 140	110	111	112	113	114	115	116	117	118	119	120
140 – 168	121	122	123	124	125	126	127	128	129	130	131
168 – 196	132	133	134	135	136	137	138	139	140	141	142
> 196	143	144	145	146	147	148	149	150	151	152	153

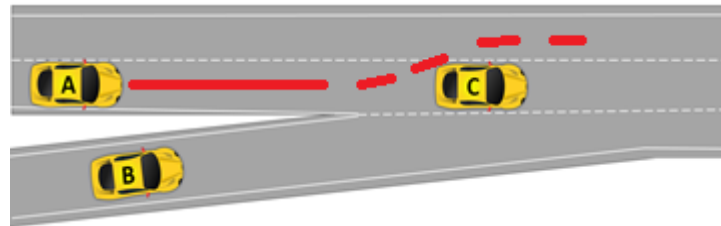
State Space

a (feet/s <sup>2</sup> )	Actions
< -9	0
-9 – -4.8	11
-4.8 – -0.6	22
-0.6 – 0.6	33
0.6 – 4.8	44
4.8 – 9	55
> 9	66

Action Space

# TRAJECTORY DATA EXTRACTION :TRAJECTORY EXTRACTION

- Extract lane #6 vehicles – About 450-550 vehicles in each time period
- Map features to position, state, action triplets
- Note: Trajectories may be ‘broken’ or incomplete



# TRAJECTORY DATA EXTRACTION : TRANSITION FUNCTION

- Based on sampling next state distribution for each action for a given state
- Frequency of  $(s, a, s')$  triplet given  $(s, a)$  is proportional to  $T(s' | s, a)$
- Motion model based update of car A's features based on action  $a$ .
- Car C assumed to be driving with constant velocity

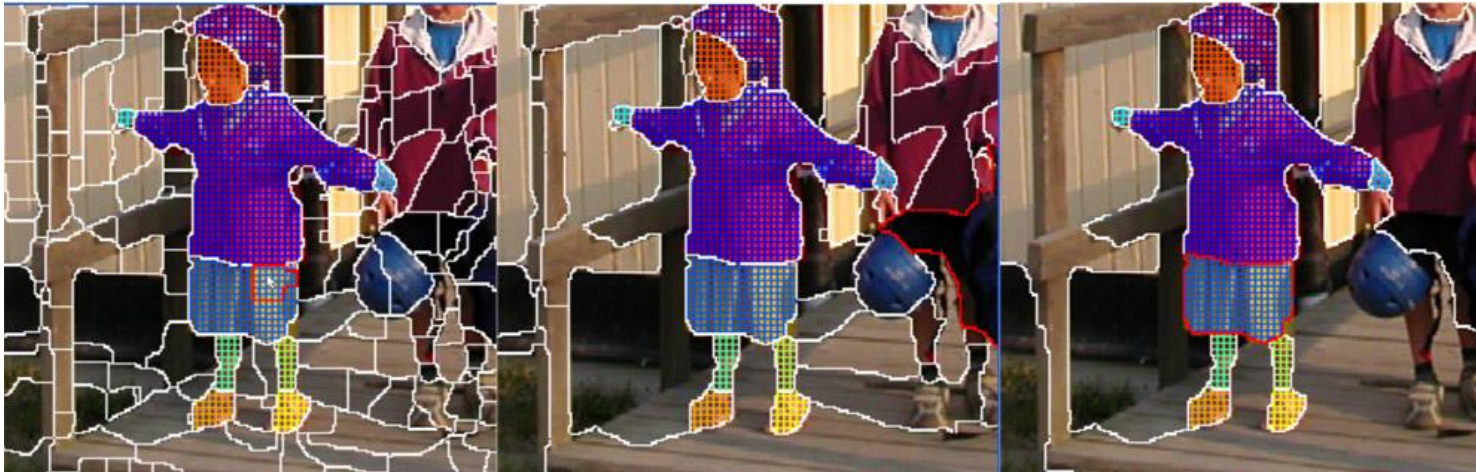


# TASK SEPARATION



# SPLIT-MERGE CLUSTERING : OVERVIEW

- To determine the number of different tasks being performed by the driver of car A, i.e., number of sections to divide the road into.
- Each section's behavior is 'homogenous' and heterogenous w.r.t. adjacent sections
- Analogous to segmentation in Image Processing



# SPLIT-MERGE CLUSTERING : OVERVIEW (contd...)

- Pixel intensity >> Section behavior →

0	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...
2	0.039801	0.084577	0.164179	0.338308	0.458021	0.21393	0.084577
3	0.006897	0.027586	0.117241	0.503448	0.276011	0.117241	0.110344
4	0.094118	0.035294	0.082353	0.517647	0.059059	0.058824	0.070588
5	0.117647	0.044118	0.102941	0.573529	0.412012	0.117647	0.029411
...	...	...	...	...	...	...	...
153	...	...	...	...	...	...	...

- Measure of similarity

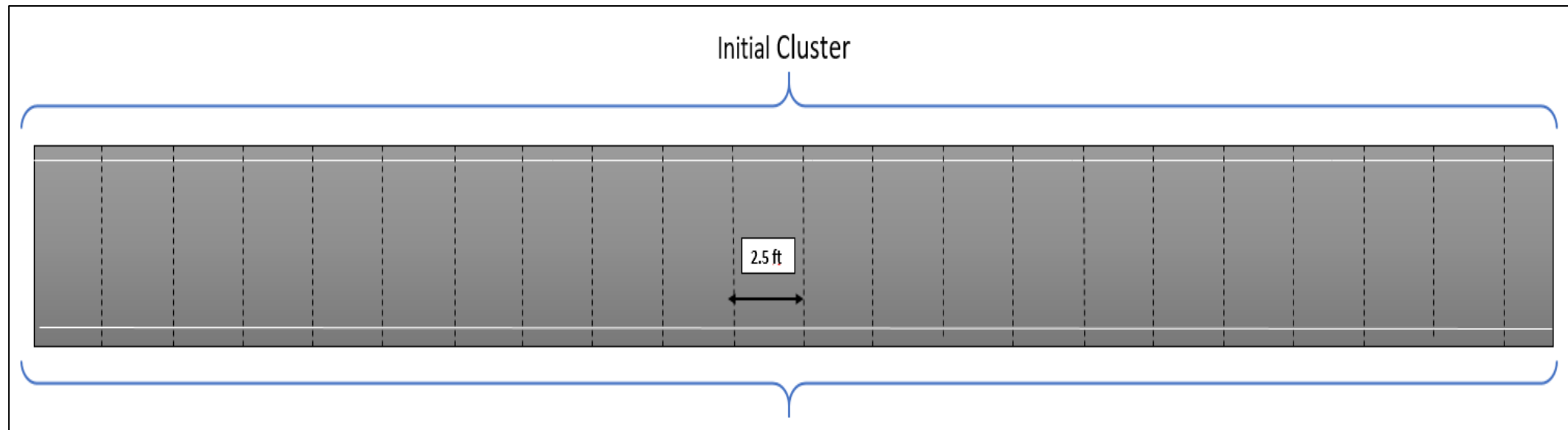
Average Hellinger distance

$$\text{Hellinger Distance : } H(P, Q) = \frac{1}{\sqrt{2}} \sqrt{\sum_{i=1}^k (\sqrt{p_i} - \sqrt{q_i})^2}$$

$$\text{Average Hellinger Distance : } H_{\text{avg}}(P, Q) = \frac{\sum_s \frac{1}{\sqrt{2}} \sqrt{\sum_{a=1}^{|A|} (\sqrt{p_{sa}} - \sqrt{q_{sa}})^2}}{|S|}$$

# SPLIT-MERGE CLUSTERING : INITIAL CLUSTER

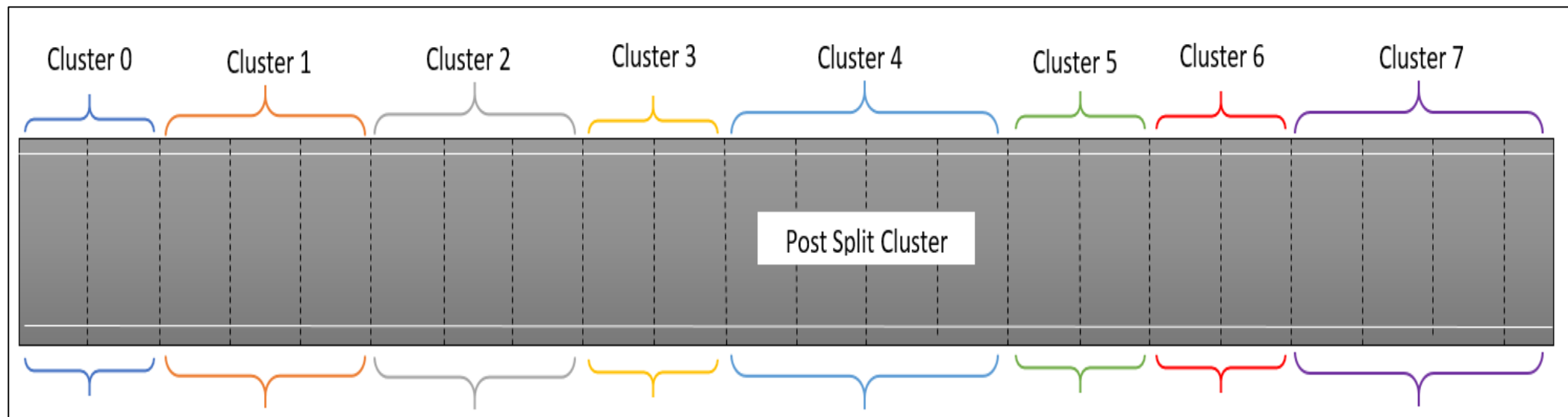
- Create initial cluster





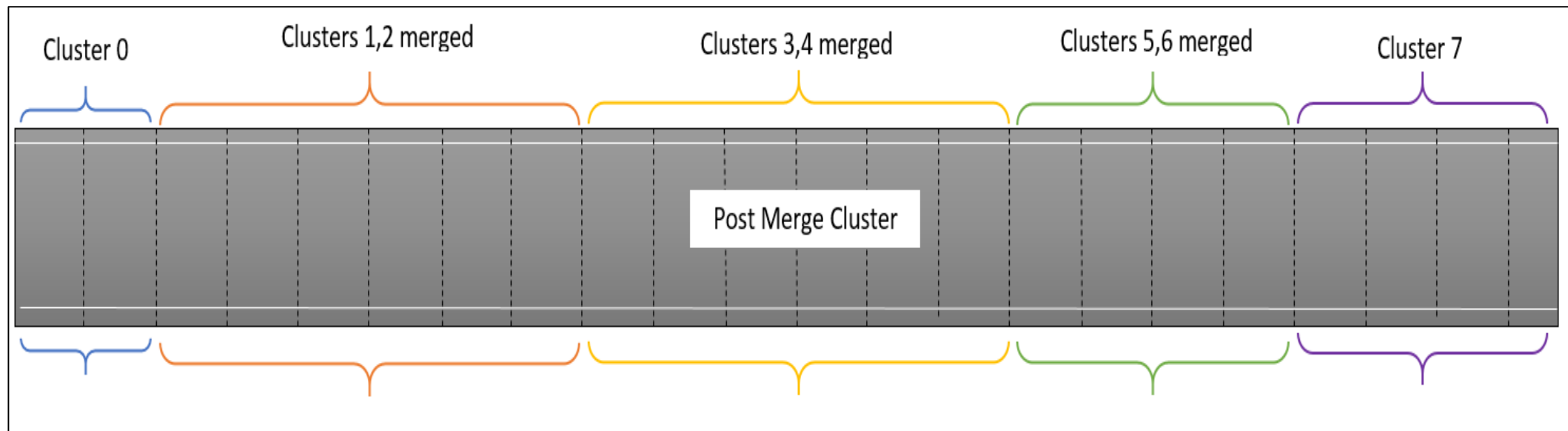
# SPLIT-MERGE CLUSTERING : POST-SPLIT CLUSTERS

- Split with threshold  $\epsilon$ , recursively, until you can



# SPLIT-MERGE CLUSTERING : POST-MERGE CLUSTERS

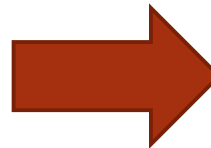
- Merge iteratively with threshold  $\delta > \epsilon$ , until no consecutive sections can merge



- Repeat the split-merge process until convergence

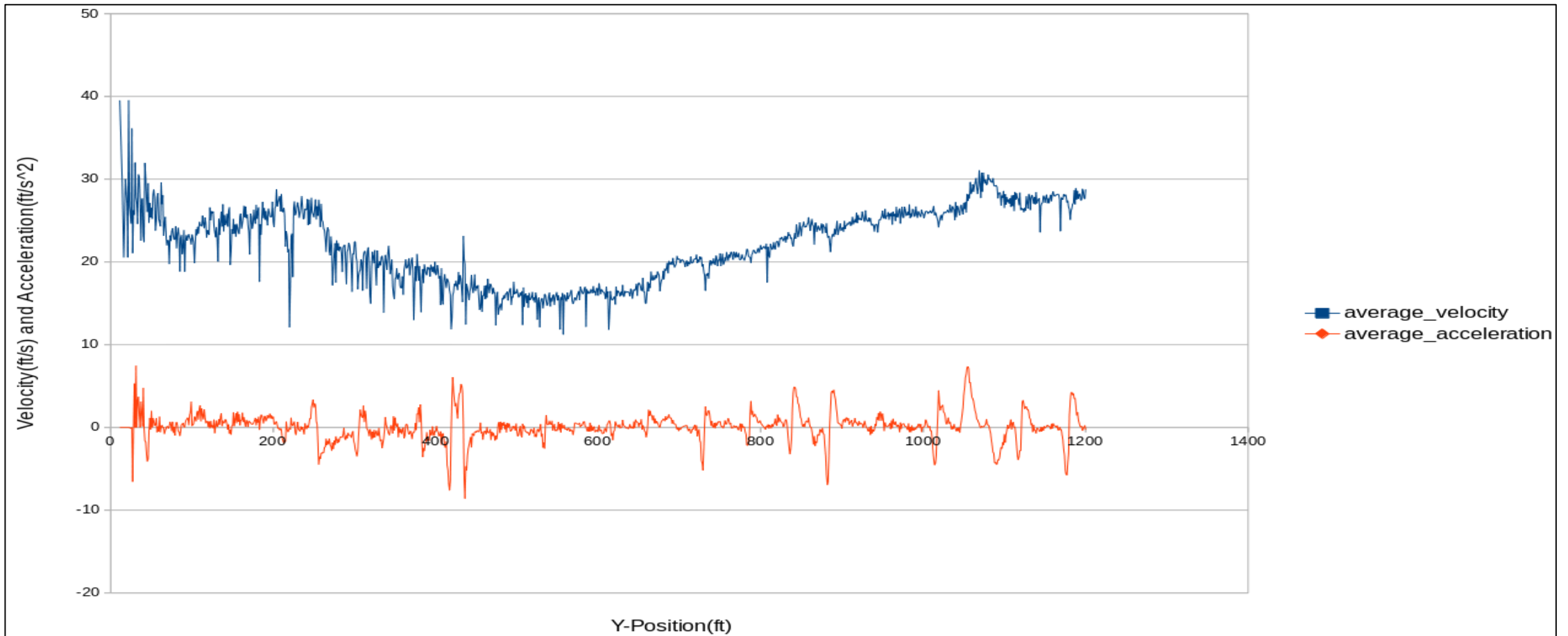
# TASK SEPARATION : RESULTS

Task section#	Bounds(y-position on freeway in feet)
0	0 – 67.5
1	67.5 – 437.5
2	437.5 – 440
3	440 – 830
4	830 – 840
5	840 – 1182.5
6	1182.5 – 1190
7	1190 – 1200



Task section#	Bounds(y-position on freeway in feet)
0	0 – 440
1	440 – 840
2	840 – 1200

# TASK SEPARATION : RESULTS COMPARISON





# LIKELIHOOD WEIGHTING MULTITASK IRL



# HIERARCHICAL BAYESIAN MULTITASK MODEL: BASICS AND NOTATIONS

- Trajectory:  $(s^t, a^t) \equiv \{(s_1, a_1), (s_2, a_2) \dots (s_t, a_t)\}$ 
  - Markovian assumption leads to
    - $s_{t+1} \sim T(S|s_t, a_t)$
    - $a_t \sim \pi(A|s_t)$
- Tasks  $m = 3$
- Number of trajectories in task  $m$  :  $N_m$
- $n^{\text{th}}$  trajectory of task  $m$ :  $d_{m,n} \equiv (s_{m,n}^{T_{m,n}}, a_{m,n}^{T_{m,n}})$  where  $T_{m,n}$  denotes the length of  $n^{\text{th}}$  trajectory of task  $m$
- Reward function of  $m^{\text{th}}$  task :  $\rho_m$
- Policy of  $m^{\text{th}}$  task :  $\pi_m$
- Controlled Markov Process:  $v = \text{MDP}/\rho$
- Problem: Find reward functions  $\rho_1, \rho_2, \rho_3$  and policies  $\pi_1, \pi_2, \pi_3$  for each of the 3 tasks

# HIERARCHICAL BAYESIAN MULTITASK MODEL: BASICS AND NOTATIONS

- The likelihood of a trajectory  $(s^t, a^t)$  given a MDP  $\mu$ , and policy  $\pi$

- $P((s^t, a^t) \mid \mu, \pi) = P((s_1, a_1), (s_2, a_2) \dots (s_t, a_t) \mid \mu, \pi)$

$$= T_\mu(s_1 | s_0, a_0) * \pi(a_1 | s_1) * T_\mu(s_2 | s_1, a_1) * \pi(a_2 | s_2) \dots * T_\mu(s_t | s_{t-1}, a_{t-1}) * \pi(a_t | s_t)$$

$$= \prod_{i=1}^t T_\mu(s_i | s_{i-1}, a_{i-1}) \pi(a_i | s_i)$$

where  $T_\mu(s_1 | s_0, a_0) = T_\mu(s_1)$  which is the initial state distribution

# HIERARCHICAL BAYESIAN MULTITASK MODEL: BASICS AND NOTATIONS

- Optimal Q-function for MDP  $\mu$  :  $Q_\mu^*(s, a)$ 
  - Can be computed using value iteration
- Need for a stochastic policy
- Stochastic policy needs to be parameterized so that prior can be assumed over it
- Soft-max function with inverse temperature parameter for policy

$$\pi(a_i | s_i, \mu, c) = \text{Softmax}(a_i | s_i, \mu, c) \quad \dots c \in \mathbb{R}^1$$

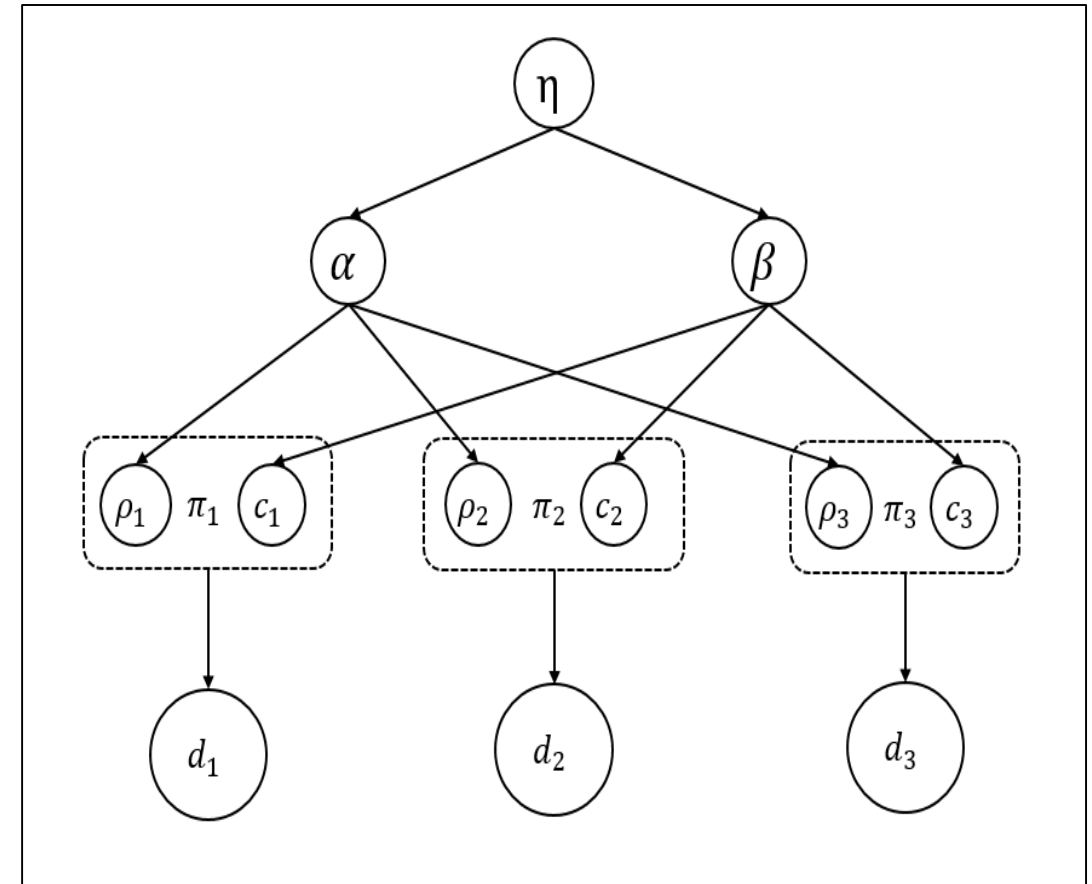
$$= \frac{e^{cQ_\mu^*(s_i, a_i)}}{\sum_{a \in A} e^{cQ_\mu^*(s_i, a)}}$$

- $\mu$  and  $\rho$  can be interchangeable since CMP  $v$  is constant



# HIERARCHICAL BAYESIAN MULTITASK MODEL : MODEL DESCRIPTION

- Space of reward functions for each task :  $\mathcal{R}$ .
  - Prior over  $\mathcal{R}$  with parameter  $\alpha$  :  $\psi(\rho) \equiv P(\mathcal{R} | \alpha)$
- Space of policies :  $\mathcal{P}$ 
  - Policy space parameterized by  $c$  given a reward function  $\rho$ ,  $\pi \equiv f(c, \rho) \in \mathcal{P}$
  - Prior over  $c$  with parameter  $\beta$  :  $\xi(c) \equiv P(\mathbb{R}^1 | \beta)$
- Space of  $\alpha$  :  $\mathcal{A}$  and Space of  $\beta$  :  $\mathcal{B}$ 
  - Product space over the reward and  $c$  parameter priors :  $\mathcal{A} \times \mathcal{B}$
  - Product hyperprior over  $\mathcal{A} \times \mathcal{B}$  :  $\eta \equiv (\eta_\alpha, \eta_\beta)$ 
    - $\eta(\alpha, \beta) = \eta_\alpha(\alpha) \cdot \eta_\beta(\beta)$
    - $\eta_\alpha(\alpha) \equiv P(\alpha | \eta_\alpha)$
    - $\eta_\beta(\beta) \equiv P(\beta | \eta_\beta)$



# CHOICE OF PRIORS : OVERVIEW

- Choice of priors for each of the 4 variables -  $\rho$ ,  $c$ ,  $\alpha$ ,  $\beta$ 
  - Prior over  $c$  : Soft-max Prior
  - Prior over the Soft-max  $c$ -prior : Soft-max hyperprior
  - Prior over  $\rho$  : Reward prior
  - Prior over the Reward prior : Reward hyperprior

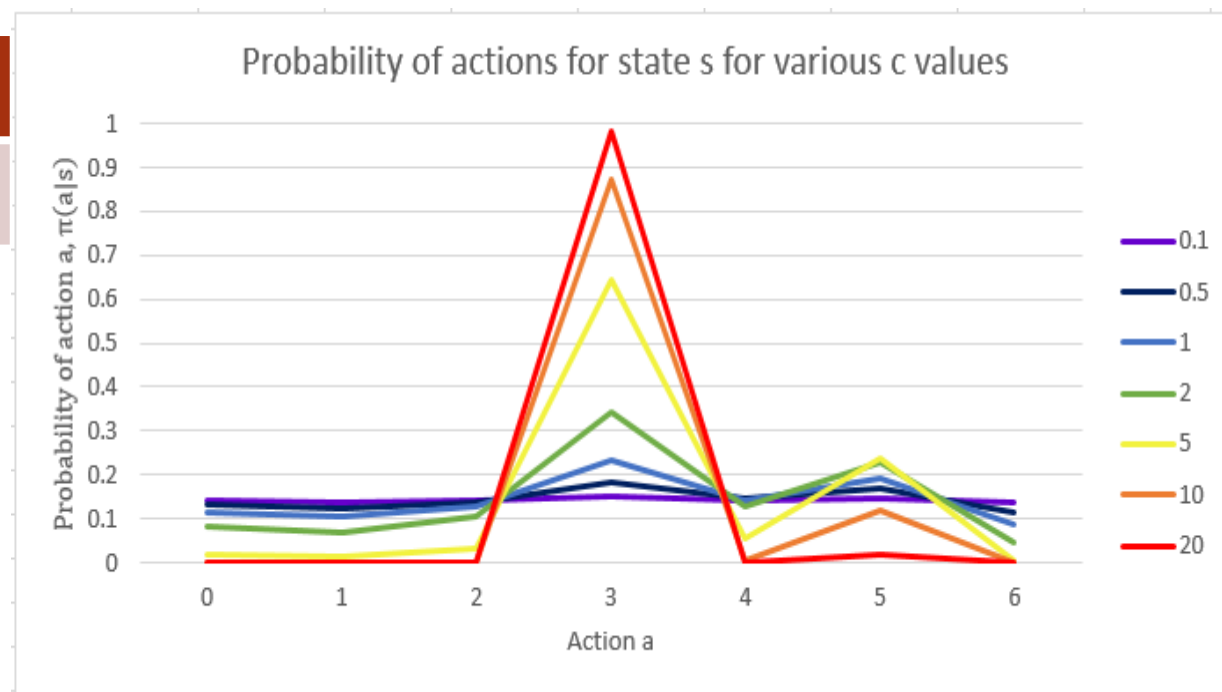
# CHOICE OF PRIORS : SOFT-MAX PRIOR

- Effect of  $c$  parameter on a policy given the reward  $\rho$

a	0	1	2	3	4	5	6
$Q^*(s,a)$	30.3	30.2	30.4	31	30.5	30.8	30

- Higher  $c$  value  $\Rightarrow$  peaky stochastic policies
- Lower the  $c$  value  $\Rightarrow$  flatter stochastic policies
- Too high  $c$  values  $\Rightarrow$  for action  $a$  corresponding to highest  $Q$ -value  $\Pr(a) \rightarrow 1$
- Choice of Prior :  $\text{Exp}(c \mid \beta)$**

If,  $\beta = 0.2$ , Expected value of this distribution =  $1/\beta = 5$

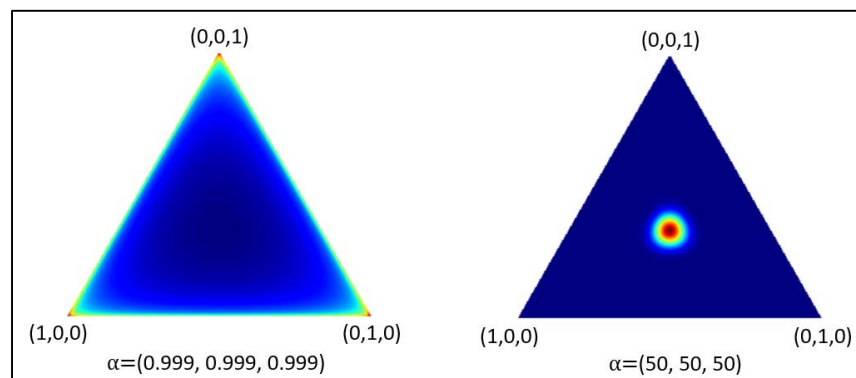


# CHOICE OF PRIORS : SOFT-MAX HYPERPRIOR

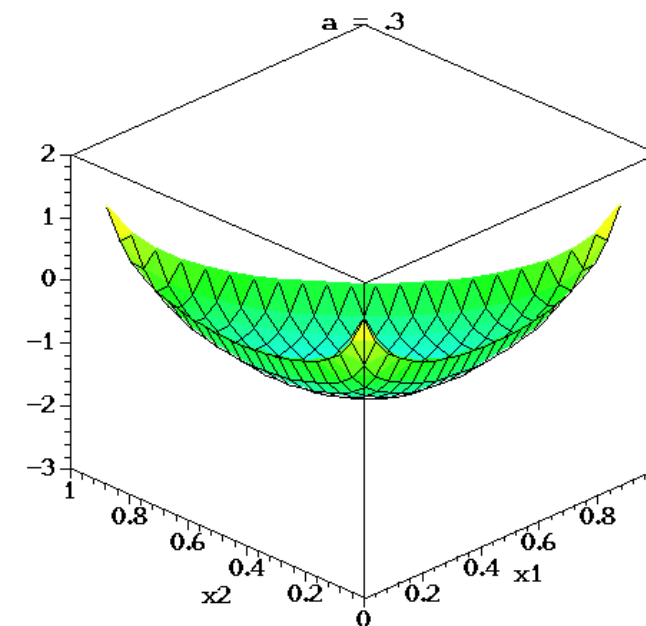
- Instead of  $\beta = 0.2$ , we use another Exponential distribution with expected value = 0.2  $\Rightarrow \eta_{\beta} = 5$
- **Choice of prior :  $\text{Exp}(\beta \mid 5)$**

# CHOICE OF PRIORS : REWARD PRIOR

- Dirichlet Distribution :  $\text{Dir}(\alpha) = \text{Dir}(\alpha_1, \alpha_2, \dots, \alpha_k)$ . Assume  $\alpha_1 = \alpha_2 = \dots = \alpha_k = \alpha$



- Every point  $x = (x_1, x_2, \dots, x_k)$  on the support is constrained by  $\sum_{i=1}^k x_i = 1$
- Rewards are parsimonious, i.e. for each state, high for few actions, low for most
- One alpha per state  $\Rightarrow |S|$   $\alpha$  parameters
- Choice of Prior : Product-Dirichlet( $\rho \mid \bar{\alpha}$ )** where  $\bar{\alpha} \in \mathbb{R}^{|S|}$



Pic courtesy: Wikipedia

# CHOICE OF PRIORS : REWARD HYPERPRIOR

- We need low  $\alpha$  values for good Dirichlet priors
- A sharp Exponential Distribution, say, with expected value = 0.1  $\Rightarrow \eta_{\alpha} = 5$
- **Choice of Prior :  $\text{Exp}(\alpha \mid 10)$**

## RESULTS : EXPECTED REWARDS

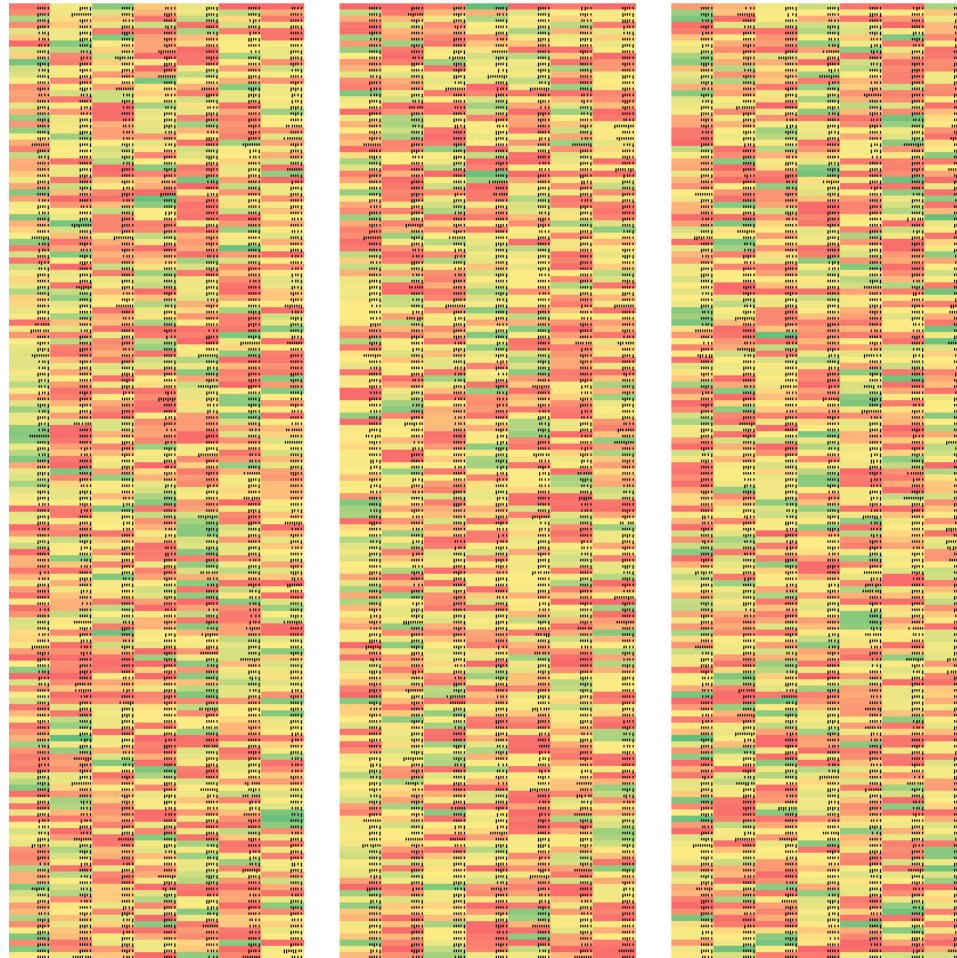
[illegible]

## RESULTS : EXPECTED REWARDS

[illegible]



# RESULTS : EXPECTED REWARDS



# RESULTS : EXPECTED POLICIES

2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431	2432	2433	2434	2435	2436	2437	2438	2439	2440	2441	2442	2443	2444	2445	2446	2447	2448	2449	2450	2451	2452	2453	2454	2455	2456	2457	2458	2459	2460	2461	2462	2463	2464	2465	2466	2467	2468	2469	2470	2471	2472	2473	2474	2475	2476	2477	2478	2479	2480	2481	2482	2483	2484	2485	2486	2487	2488	2489	2490	2491	2492	2493	2494	2495	2496	2497	2498	2499	2500	2501	2502	2503	2504	2505	2506	2507	2508	2509	2510	2511	2512	2513	2514	2515	2516	2517	2518	2519	2520	2521	2522	2523	2524	2525	2526	2527	2528	2529	2530	2531	2532	2533	2534	2535	2536	2537	2538	2539	2540	2541	2542	2543	2544	2545	2546	2547	2548	2549	2550	2551	2552	2553	2554	2555	2556	2557	2558	2559	2560	2561	2562	2563	2564	2565	2566	2567	2568	2569	2570	2571	2572	2573	2574	2575	2576	2577	2578	2579	2580	2581	2582	2583	2584	2585	2586	2587	2588	2589	2590	2591	2592	2593	2594	2595	2596	2597	2598	2599	2600	2601	2602	2603	2604	2605	2606	2607	2608	2609	2610	2611	2612	2613	2614	2615	2616	2617	2618	2619	2620	2621	2622	2623	2624	2625	2626	2627	2628	2629	2630	2631	2632	2633	2634	2635	2636	2637	2638	2639	2640	2641	2642	2643	2644	2645	2646	2647	2648	2649	2650	2651	2652	2653	2654	2655	2656	2657	2658	2659	2660	2661	2662	2663	2664	2665	2666	2667	2668	2669	2670	2671	2672	2673	2674	2675	2676	2677	2678	2679	2680	2681	2682	2683	2684	2685	2686	2687	2688	2689	2690	2691	2692	2693	2694	2695	2696	2697	2698	2699	2700	2701	2702	2703	2704	2705	2706	2707	2708	2709	2710	2711	2712	2713	2714	2715	2716	2717	2718	2719	2720	2721	2722	2723	2724	2725	2726	2727	2728	2729	2730	2731	2732	2733	2734	2735	2736	2737	2738	2739	2740	2741	2742	2743	2744	2745	2746	2747	2748	2749	2750	2751	2752	2753	2754	2755	2756	2757	2758	2759	2760	2761	2762	2763	2764	2765	2766	2767	2768	2769	2770	2771	2772	2773	2774	2775	2776	2777	2778	2779	2780	2781	2782	2783	2784	2785	2786	2787	2788	2789	2790	2791	2792	2793	2794	2795	2796	2797	2798	2799	2800	2801	2802	2803	2804	2805	2806	2807	2808	2809	2810	2811	2812	2813	2814	2815	2816	2817	2818	2819	2820	2821	2822	2823	2824	2825	2826	2827	2828	2829	2830	2831	2832	2833	2834	2835	2836	2837	2838	2839	2840	2841	2842	2843	2844	2845	2846	2847	2848	2849	2850	2851	2852	2853	2854	2855	2856	2857	2858	2859	2860	2861	2862	2863	2864	2865	2866	2867	2868	2869	2870	2871	2872	2873	2874	2875	2876	2877	2878	2879	2880	2881	2882	2883	2884	2885	2886	2887	2888	2889	2890	2891	2892	2893	2894	2895	2896	2897	2898	2899	2900	2901	2902	2903	2904	2905	2906	2907	2908	2909	2910	2911	2912	2913	2914	2915	2916	2917	2918	2919	2920	2921	2922	2923	2924	2925	2926	2927	2928	2929	2930	2931	2932	2933	2934	2935	2936	2937	2938	2939	2940	2941	2942	2943	2944	2945	2946	2947	2948	2949	2950	2951	2952	2953	2954	2955	2956	2957	2958	2959	2960	2961	2962	2963	2964	2965	2966	2967	2968	2969	2970	2971	2972	2973	2974	2975	2976	2977	2978	2979	2980	2981	2982	2983	2984	2985	2986	2987	2988	2989	2990	2991	2992	2993	2994	2995	2996	2997	2998	2999	3000	3001	3002	3003	3004	3005	3006	3007	3008	3009	3010	3011	3012	3013	3014	3015	3016	3017	3018	3019	3020	3021	3022	3023	3024	3025	3026	3027	3028	3029	3030	3031	3032	3033	3034	3035	3036	3037	3038	3039	3040	3041	3042	3043	3044	3045	3046	3047	3048	3049	3050	3051	3052	3053	3054	3055	3056	3057	3058	3059	3060	3061	3062	3063	3064	3065	3066	3067	3068	3069	3070	3071	3072	3073	3074	3075	3076	3077	3078	3079	3080	3081	3082	3083	3084	3085	3086	3087	3088	3089	3090	3091	3092	3093	3094	3095	3096	3097	3098	3099	3100	3101	3102	3103	3104	3105	3106	3107	3108	3109	3110	3111	3112	3113	3114	3115	3116	3117	3118	3119	3120	3121	3122	3123	3124	3125	3126	3127	3128	3129	3130	3131	3132	3133	3134	3135	3136	3137	3138	3139	3140	3141	3142	3143	3144	3145	3146	3147	3148	3149	3150	3151	3152	3153	3154	3155	3156	3157	3158	3159	3160	3161	3162	3163	3164	3165	3166	3167	3168	3169	3170	3171	3172	3173	3174	3175	3176	3177	3178	3179	3180	3181	3182	3183	3184	3185	3186	3187	3188	3189	3190	3191	3192	3193	3194	3195	3196	3197	3198	3199	3200	3201	3202	3203	3204	3205	3206	3207	3208	3209	3210	3211	3212	3213	3214	3215	3216	3217	3218	3219	3220	3221	3222	3223	3224	3225	3226	3227	3228	3229	3230	3231	3232	3233	3234	3235	3236	3237	3238	3239	3240	3241	3242	3243	3244	3245	3246	3247	3248	3249	3250	3251	3252	3253	3254	3255	3256	3257	3258	3259	3260	3261	3262	3263	3264	3265	3266	3267	3268	3269	3270	3271	3272	3273	3274	3275	3276	3277	3278	3279	3280	3281	3282	3283	3284	3285	3286	3287	3288	3289	3290	3291	3292	3293	3294	3295	3296	3297	3298	3299	3300	3301	3302	3303	3304	3305	3306	3307	3308	3309	3310	3311	3312	3313	3314	3315	3316	3317	3318	3319	3320	3321	3322	3323	3324	3325	3326	3327	3328	3329	3330	3331	3332	3333	3334	3335	3336	3337	3338	3339	3340	3341	3342	3343	3344	3345	3346	3347	3348	3349	3350	3351	3352	3353	3354	3355	3356	3357	3358	3359	3360	3361	3362	3363	3364	3365	3366	3367	3368	3369	3370	3371	3372	3373	3374	3375	3376	3377	3378	3379	3380	3381	3382	3383	3384
------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------

## RESULTS : EXPECTED POLICIES

[illegible]



## RESULTS : EXPECTED POLICIES

[illegible]

## RESULTS : EXPECTED POLICIES

[illegible]



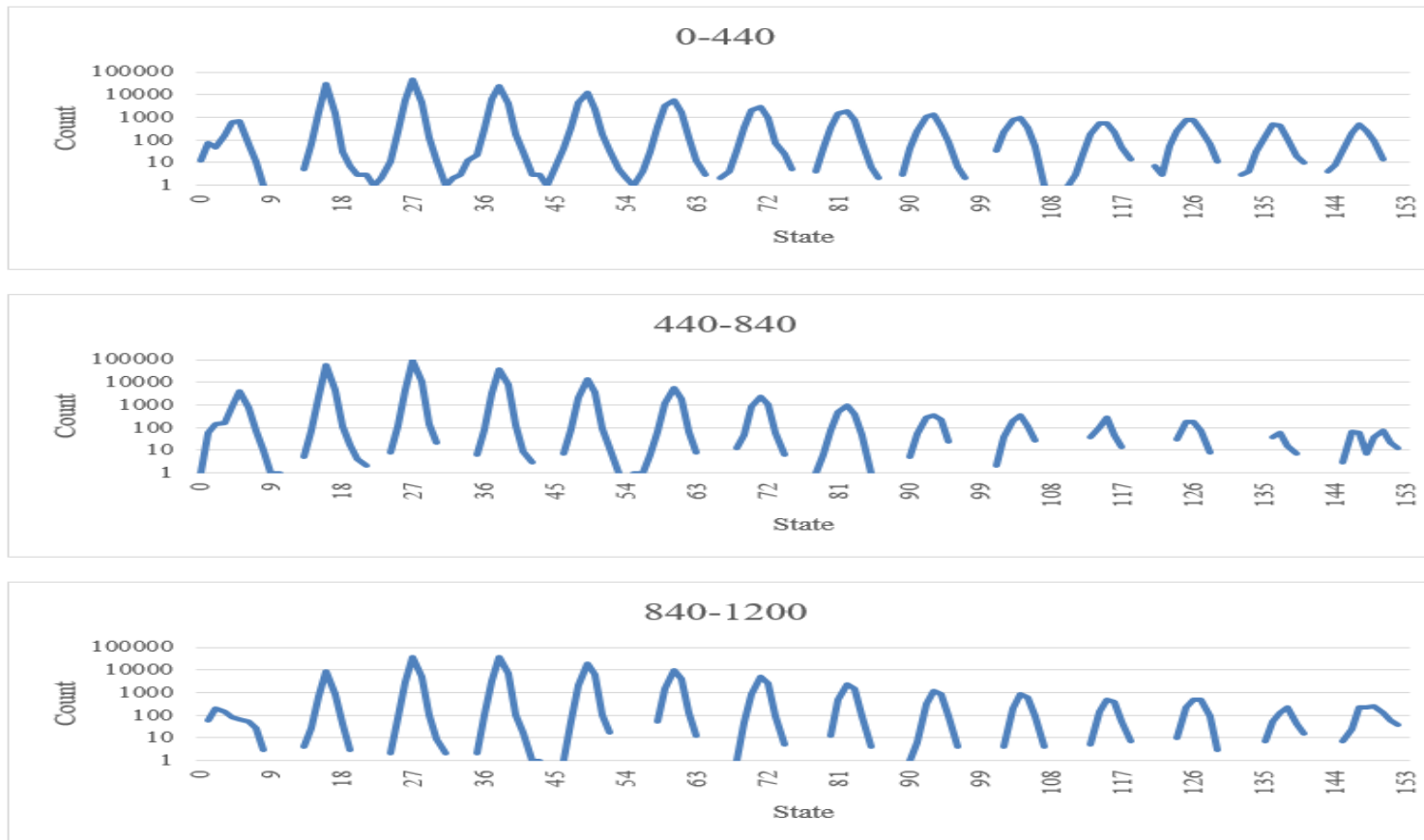
# CONCLUSION AND FUTURE WORK



# CONCLUSION

- High variance in expected rewards from different runs
- No discernible pattern suggesting difference in reward functions for the tasks
- Policies follow a trend
- Task 2 policy favors braking/acceleration actions more than constant velocity
- Peakier policies for task 2 compared to task 1 and task 3
- Suspected main cause: Not enough information about all states in the trajectories!

# CONCLUSION

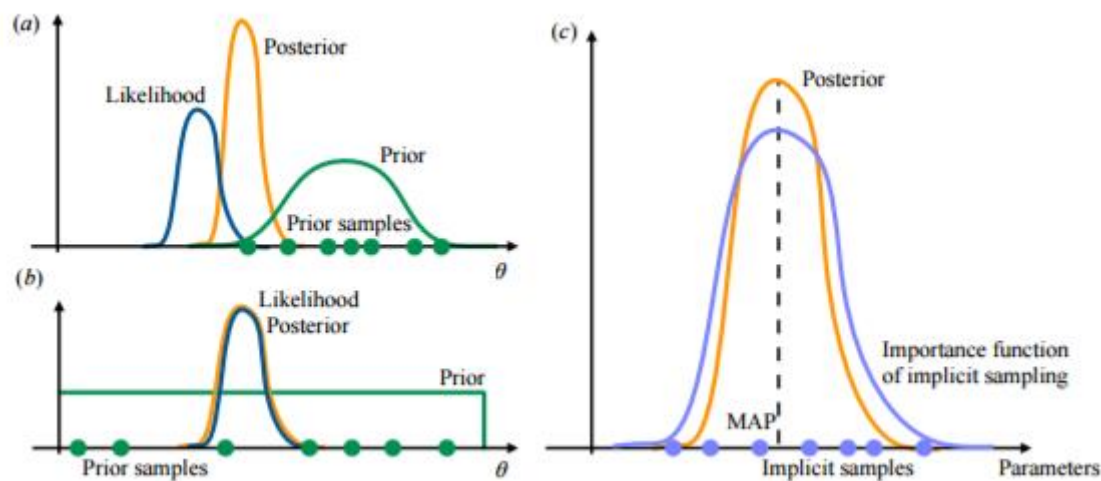


State counts in all trajectories



# FUTUREWORK

- Implicit sampling
  - Generating implicit samples that receive large posterior probability instead of generating a large number of prior samples which rarely fall in the high posterior region.



# FUTUREWORK

- Eliminate c parameter, reduce network complexity
  - Attempt to learn rewards of the expert than trying to best it
  - Computing stochastic policies as

$$\pi^*(a | s) \propto \sum_{s'} T(s, a, s') * V(s')$$
$$\pi^*(a | s) = \frac{\sum_{s'} T(s, a, s') * V(s')}{\sum_{a'} \sum_{s'} T(s, a', s') * V(s')}$$

# FUTUREWORK

- Better state space
  - Non-uniform discretization
  - Highlight minute differences between nearby states which may have been fused inadvertently
  - Reduce states which are missing in trajectories by merging them together

# REFERENCES

- [1] Jochem, T., Pomerleau, D., Kumar, B. and Armstrong, J., 1995, September. PANS: A portable navigation platform. In *Intelligent Vehicles'95 Symposium., Proceedings of the* (pp. 107-112). IEEE.
- [2] Das, I., 2016, August. Inverse reinforcement learning of risk-sensitive utility. (Masters' Thesis, University of Georgia Athens)
- [3] Russell, S.J., Norvig, P. & Davis, E., 2010. *Artificial intelligence: a modern approach* 3rd ed.(Vol 2), Boston: Pearson.
- [4] Russell, S., 1998, July. Learning agents for uncertain environments. In *Proceedings of the eleventh annual conference on Computational learning theory* (pp. 101-103). ACM.
- [5] Ng, A.Y. and Russell, S.J., 2000, June. Algorithms for inverse reinforcement learning. In *Icml* (pp. 663-670).
- [6] Ziebart, B.D., Maas, A.L., Bagnell, J.A. and Dey, A.K., 2008, July. Maximum Entropy Inverse Reinforcement Learning. In *AAAI* (Vol. 8, pp. 1433-1438).
- [7] Ramachandran, D. and Amir, E., 2007. Bayesian inverse reinforcement learning. *Urbana*, 51(61801), pp.1-4.
- [8] Rothkopf, C. and Dimitrakakis, C., 2011. Preference elicitation and inverse reinforcement learning. *Machine Learning and Knowledge Discovery in Databases*, pp.34-48.

## REFERENCES (Contd...)

- [9] Dimitrakakis, C. and Rothkopf, C.A., 2011, September. Bayesian multitask inverse reinforcement learning. In *European Workshop on Reinforcement Learning* (pp. 273-284). Springer Berlin Heidelberg.
- [10] Babes, M., Marivate, V., Subramanian, K. and Littman, M.L., 2011. Apprenticeship learning about multiple intentions. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (pp. 897-904).
- [11] Choi, J. and Kim, K.E., 2012. Nonparametric Bayesian inverse reinforcement learning for multiple reward functions. In *Advances in Neural Information Processing Systems* (pp. 305-313).
- [12] Wilson, A., Fern, A., Ray, S. and Tadepalli, P., 2007, June. Multi-task reinforcement learning: a hierarchical Bayesian approach. In *Proceedings of the 24th international conference on Machine learning* (pp. 1015-1022). ACM.
- [13] US Department of Transportation, 2006, December. Fact Sheet-Interstate 80 Freeway Dataset, *Federal Highway Administration (FHWA)*.
- [14] Geweke, J., 1989. Bayesian inference in econometric models using Monte Carlo integration. *Econometrica: Journal of the Econometric Society*, pp.1317-1339.
- [15] Thrun, S., Burgard, W. and Fox, D., 2005. *Probabilistic robotics*. MIT press.
- [16] Morzfeld, M., Tu, X., Wilkening, J. and Chorin, A., 2015. Parameter estimation by implicit sampling. *Communications in Applied Mathematics and Computational Science*, 10(2), pp.205-225.