Course Code	18AIC304J	Course Name	Reinforcement Learning Techniques	Course Category	C	*Professional Core Courses*	L	T	P	C
							2	0	2	3

Pre-requisite Courses	*Nil*	Co-requisite Courses	*Nil*	Progressive Courses	*Nil*
Course Offering Department	*Artificial Intelligence*	**Data Book / Codes/Standards**	*Nil*		

Course Objectives	*The purpose of learning this course is to:*
1	*Introduce a range of topics related to Reinforcement Learning and probability concepts*
2	*Gain knowledge on Markov Decision Process*
3	*The dynamic programming methods of Reinforcement L*
4	*The Monte Carlo Prediction and Time Difference Learning*
5	*Function Approximation methods and Q-learning*

Course Outcomes (CO):	*At the end of this course, learners will be able to:*	Level of Thinking (Bloom)	Expected Proficiency (%)	Expected Attainment (%)	Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment &	Ethics	Individual & Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
		1	2	3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CO-1:	*Device solutions to small problems using reinforcement Learning and Concepts of Probability*	*3*	*80*	*75*	*2*	*2*	*2*	-	-	-	-	-	-	-	-	*3*	-	-	-
CO-2:	*Express proficiency in utilizing Markov Decision Process and Bellman Equation for reinforcement learning*	*3*	*75*	*70*	*3*	*3*	*2*	-	-	-	-	-	-	-	-	*3*	-	-	-
CO-3:	*Apply dynamic programming techniques on Markov decision process and Monte Carlo methods*	*3*	*80*	*70*	*3*	*3*	*3*	-	-	-	-	-	-	-	-	*3*	-	-	-
CO-4:	*Implement Time difference Learning for real world problems*	*4*	*80*	*70*	*3*	*3*	*3*	-	-	-	-	-	-	-	-	*3*	-	-	-
CO-5:	*Apply Approximation methods of learning and Q-Learning Techniques to solve a problem*	*3*	*75*	*70*	*3*	*3*	*3*	-	-	-	-	-	-	-	-	*3*	-	-	-

Duration (hour)		***15***	**15**	***15***	***15***	***15***
S-1	**SLO-1**	*Introduction to Reinforcement Learning*	*Markov Decision Process*	*Overview of dynamic programming forMDP*	*Monte Carlo Prediction*	*Getting started with the function approximation methods*
	SLO-2	*Examples*	*The Agent–Environment Interface*		*Monte Carlo Estimation of Action Values*	
S-2	**SLO-1**	*Elements of Reinforcement Learning - Limitations and Scope*	*Goals and Rewards– Returns*	*Definition and formulation of planning in MDPs*	*Monte Carlo Control*	*Revisiting risk minimization*
	SLO-2		*Unified Notation for Episodic and Continuing Tasks*			*gradient descent from Machine Learning*

S-3	**SLO-1**	*Tic-Tac-Toe example*	*The Markov Property*	*principle of optimality*	*Off-policy Prediction via Importance Sampling*	*Gradient MC and Semi-gradient TD(0) algorithms*
	SLO-2	*History of Reinforcement Learning*	*Markov Decision Processes*	*Policy Evaluation*	*Incremental Implementation*	
S 4-5 (LAB)	**SLO-1**	*LAB 1: Installation of Code Standards and Libraries used in RL (Python/Keras/Tensorflow)*	*LAB 4: Dynamic programming algorithms for solving MDPs.*	*Lab 7: Monte Carlo Prediction*	*Lab 10: Q-Learning (Off Policy TD Learning)*	*Lab 13: Policy Gradient: REINFORCE with Baseline*
	SLO-2					
S-6	**SLO-1**	*Probability concepts*	*Value Functions*	*Policy Improvement*	*Off-Policy Monte Carlo Control*	*Linear Methods*
	SLO-2	*Axioms of probability*				
S-7	**SLO-1**	*Concepts of random variables*	*Optimal Value Functions*	*Policy Iteration*	*Temporal-Difference Learning: TD Prediction*	*Eligibility trace for function approximation*
	SLO-2					
S-8	**SLO-1**	*PMF, PDFs, CDFs, Expectation*	*Optimality and Approximation*	*Value Iteration*	*Advantages of TD Prediction Methods*	*Control with function approximation*
	SLO-2				*Optimality of TD(0)*	
S 9-10 (LAB)	**SLO-1**	*Lab 2: Implement Tic-tac-toe problem*	*Lab 5: Dynamic Programming: Policy Evaluation and Policy Iteration*	*Lab 8: Monte Carlo Off-Policy Control with Importance Sampling*	*Lab 11:Q-Learning with Linear Function Approximation*	*Lab 14: Policy Gradient: Actor Critic with Baseline*
	SLO-2					
S-11	**SLO-1**	*Concepts of joint and multiple random variables*	*Bellman expectation equations*	*Generalized Policy Iteration*	*TD(1)*	*Least squares, Experience replay in deep Q-Networks*
	SLO-2			*Efficiency of Dynamic Programming*	*TD(λ)*	
S-12	**SLO-1**	*joint, conditional and marginal distributions*	*Bellman optimality equations*	*Banach fixed point theorem*	*Sarsa: On-Policy TD Control*	*Naive REINFORCE algorithm*
	SLO-2	*Correlation and independence*				*Bias and variance in Reinforcement Learning*
S-13	**SLO-1**	*An-Armed Bandit Problem*	*Markov Reward Process*	*proof of convergence of policy evaluation and value iteration algorithms*	*Q-Learning: Off-Policy TD Control*	*Actor–Critic Methods*
	SLO-2	*Action-Value Methods.*			*unified view of DP, MC and TD evaluation methods*	
S 14-15 (LAB)	**SLO-1**	*Lab 3: Implement Armed Bandit Problem*	*Lab 6:Dynamic Programming: Policy Improvement and Value Iteration*	*Lab 9: SARSA (On Policy TD Learning)*	*Lab 12: Deep Q-Learning for Atari Games*	*Lab 15: Policy Gradient: Actor Critic with Baseline for Continuous Action Spaces*
	SLO-2					

Learning Resources	1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MIT Press, 2019 2. Algorithms for Reinforcement learning, by Csaba Szepesvari, Morgan & Claypool Publishers, 2010. 3. Probability, Statistics, and Random Processes for Electrical Engineering, 3rd Edition, Alberto Leon-Garcia, 2009 4. "Machine Learning: A Probabilistic Perspective", Kevin P. Murphy, 2012

Learning Assessment											
	Bloom's Level of Thinking	Continuous Learning Assessment (50% weightage)								Final Examination (50% weightage)	
		CLA – 1 (10%)		CLA – 2 (15%)		CLA – 3 (15%)		CLA – 4 (10%)			
		Theory (5%)	Practice (5%)	Theory (7.5%)	Practice (7.5%)	Theory (7.5%)	Practice (7.5%)	Theory (5%)	Practice (5%)	Theory (25%)	Practice (25%)
Level 1	Remember	20%		15%		15%				15%	
Level 2	Understand	20%		25%		25%				20%	
Level 3	Apply	45%	30%	40%	35%	40%	40%	45%	20%	45%	30%
Level 4	Analyze	15%	40%	20%	35%	20%	30%	35%	50%	20%	35%
Level 5	Evaluate		30%		30%		30%	20%	30%		35%
Level 6	Create	20%		15%		15%				15%	
	Total	100 %	100 %	100 %	100 %	100 %	100 %	100%	100%	100%	100%

CLA – 4 can be from any combination of these: Assignments, Seminars, Tech Talks, Mini-Projects, Case-Studies, Self-Study, MOOCs, Certifications, Conf. Paper etc.,

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr. Soundararajan Dhakshinamoorthy, Principle Software Engineer, Technology Leadership, Optum Insights India Pvt Ltd., Chennai.	*Dr. Udendran, Dept. of CSE., Bharathidasan University, Tiruchirappalli*	Mr. S. Joseph James, SRMIST