Project: Kincet image data visualization

Andrew

June 25, 2021

Summary This short article is to explain the

Milestone 1: Kincet image data visualization - Exercise 2

The following deliverables are requested:

Deliverables will be 2nd exercise conversion from the repo to python and concept explanation

https://github.com/mcharrak/probabilistic-graphical-models-PGM-learning-coursera-daphne-koller/tree/master/PA_ex2-Learning-Tree-Structured-Networks-Release

Submissions

In the attachment along with this document, a python project (Python 3.x) will the following sources file is provided.

The folder structure of the python project (Python 3.x) is shown in the left in the following figure; correspondingly the folder structure of the matlab project is shown in the right.

When converting from matlab to python, the first challenge is the indexing. In matlab, the indexing is starting from 1; while in the python the indexing is starting from 0.

Test Cases

Conclusion: From the accuracy from the test case o2/o3/o4, the python project is giving the identical performance as the matlab project.

idea .idea		
pycache		
doc		
ClassifyDataset.py		
ComputeLogLikelihood.py		
ConvertAtoG.py		
FitGaussianParameters.py		
itLinearGaussianParameters.py		
func_DrawLine.py	Current Folder	
GaussianMutualInformation.py	B lib	
	class_idxs_amine.mat	
LearnCPDsGivenGraph.py	class pred amine.mat	
LearnGraphAndCPDs.py	dass_probs_amine.mat	
LearnGraphStructure.py	class_probs_gt.mat ClassifyDataset.asv	
LearnGraphStructure.py	ClassifyDataset.m	
lognormpdf.py	ComputeLogLikelihood.m	
MaxSpanningTree.py	ConvertAtoG.m	
MaxSpanning Free.py	FitLinearGaussianParameters.m	
PA8Data	func_DrawLine.m	
	GaussianMutualInformation.m LearnCPDsGivenGraph.asv	
PA8SampleCases	EearnCPDsGivenGraph.m	
PA9SampleCases	LearnGraphAndCPDs.m	
	LearnGraphStructure.m	
Sample Gaussian.py		
SampleMultinomial.py	octave-workspace	
	PA8Data.mat PA8SampleCases.mat	
SamplePose.py	PA-Learning-Tree-Structured-Networks.pd	
ShowPose.py	SampleGaussian.m	
	SampleMultinomial.m SamplePose.m	
itest01.py	ShowPose.m	
test02.py	submit.m	
	submit_input.mat test01.m	
test03.py	test01.m	
test04.py	test02.m	
	test03.asv	
test05.py	test04.asv	
☐ VisualizeDataset.py	test04.m	
	test05.m VisualizeDataset.m	
☐ VisualizeModels.py	VisualizeDataset.m VisualizeModels.m	

(a) python project

(b) matlab project

name	description	output from Matlab	output from Python
testo1	visualize the data set		50 - 50 -
testo2	Learning with the naive	log likelihood: -	log likelihood: -
102	bayes model	46680.632740	46680.632739531786
	bayes model	Accuracy: 0.7875	Accuracy: 0.7875
			11000111091 01/0/3
testo3	Learning with known	log likelihood: -	log likelihood: -
	skeletal structure	27716.908524	27716.90852437852
		Accuracy: 0.8375	Accuracy: 0.8375
testo4	Learning the graph	log likelihood: -	log likelihood: -
	structures and then	26640.584282	26640.584282426134
	learning the parameters	Accuracy: 0.9175	Accuracy: 0.9175
	for each graph structure		0 50 50 50 - 100 - 130 - 130 - 200 - 250 - 250 - 150 250 0 150 250