

BN Toolbox Help for Assignment #3

<http://bnt.googlecode.com/svn/trunk/docs/usage.html>

Selected Functions

BNT functions:

Initialize:

mk_bnet – create a bayesian network,

tabular_CPD – randomly initialize its CPDs (CPTs in our case),

Learn:

learn_params – use MLE to find the parameter values,

learn_params_em – do the same for incomplete data (more time consuming than previous),

learn_struct_K2 – apply K2 to get a network structure, a variable sort must be given,

learn_struct_mcmc -- apply MCMC to get a set of network structures, the best is to be found yet,

Inference:

jtree_inf_engine – one of the inference engines (an overview:

http://bnt.googlecode.com/svn/trunk/docs/usage.html#engine_summary)

enter_evidence – add the evidence to the engine,

marginal_nodes – run the inference engine, get probabilities,

Score graphs:

log_lik_complete – compute log likelihood for a completely observed data set,

enter_evidence -- can be used to get log likelihood for an incomplete data set too,

score_dags – evaluate one or more DAGs in terms of BIC or Bayesian score,

Visualize:

draw_graph – draw a network,

dispcpt – show CPT (see show_CPT in family_learn.m for easier utilization).

Matlab functions:

knnsearch -- find k -nearest neighbors using data,

cell2num – convert a cell array into a numeric matrix for easier manipulation (identification of NaNs etc.),

External Functions:

initCPTs – call tabular_CPD for all the network nodes,

show_CPT – show CPT of a network node (uses dispcpt), see family_learn.m

top_sort_dag -- rearrange the incidence matrix such that the variables appear in topological sort,

Get Function Details:

doc function_name

Runtime error reasons

The node CPTs were not (randomly) initialized

tabular_CPD or initCPTs were not called before calling e.g., learn_params,

Network node sizes and sample node sizes do not match
the ordering of nodes in the network does not agree with the variable ordering in the sample,
if the node sizes do not match a runtime error occurs, if the ordering is wrong keeping the node sizes
the network is misleading “only”

Networks with poor performance

Parameters were not learned and remained random

The variable sort is not topological
the model occasionally works, nevertheless, inference engines may fail and give wrong probabilities

Improper network structure
not enough samples used for training, improper starting sort or structure etc.