Probabilistic Graphical Models Bayesian Network and Markov Network Construction for Children's Handwriting Analysis

Implemented in R

Harshith Kumar Ramadev
The State University of New York at Buffalo, USA
harshith@buffalo.edu

Abstract

Probabilistic Graphical Models represent a set of random variables and their conditional dependencies. We construct PGM's to deduce inference on Children's Handwriting Analysis. Dataset is a collection of both cursive and handprint data of Children's handwriting. Bayesian and Markov networks are implemented in this paper.

Keywords

Probabilistic Graphical Model, Bayesian, Markov, CPD, Entropy, Mean, Log Loss, KL Divergence, Glasso graphical package, Probabilistic mass functions

Introduction

Probabilistic Graphical Model is a construct that combines probability and logical structure to represent real world problems. The primary point of PGM is to proficiently indicate the dependencies in the dataset rather than an arrangement of random variables. In this paper, two kinds of Probabilistic Graphical Models are Bayesian and Markov Networks on Children's handwriting dataset. Bayesian Networks are directed graphical models and Markov Networks are undirected graphical models. The goal of this paper is to implement different algorithms for construction of PGM's and evaluate the models through inference on Children's handwriting dataset. The handling of missing values is also mentioned in the paper with Missing value implementation algorithms to deduce precise inference by replacing the missing values with the most likely available value.

Dataset

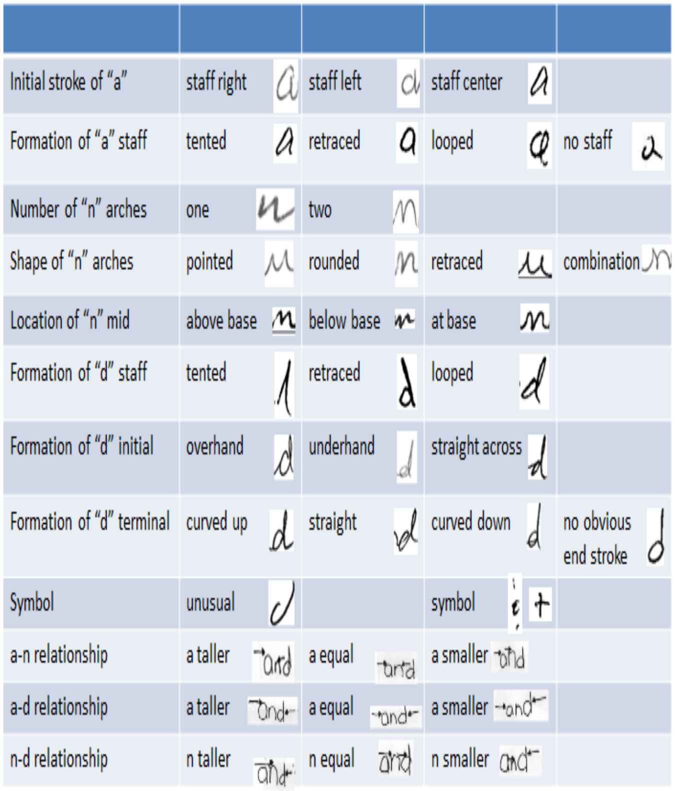
The dataset is a collection of handwriting samples of students for two styles of writing - Cursive style samples of students from Grade 3, Grade 4 and Grade 5 classes and Hand-print style samples of students from Grade 1, Grade 2, Grade 3, Grade 4 and Grade 5 classes respectively. The word 'and' was extracted from the paragraph written by the students and its characteristics were analyzed. The word 'and' was represented by 12 features each taking a range of discrete values captured with the help of a Truthing tool.

The handwriting tests are from an extensive number of understudies as they are taking in (second grade) and also have recently taken in (third and fourth grade) on how to create cursive and printed composition (2012). This is proceeded through the understudy's essential instructive vocation (2013, 2014). The gathering happens yearly, every spring for 3 years as the student's writing and hand printing aptitudes keep on developing over the period of time.

The dataset consists of 12 different features comprising of values ranging from 0 to 4,with an additional unnecessary/extra values such as 99 and -1. The features comprises of "Strokes","Shapes", "Formation" and so on.

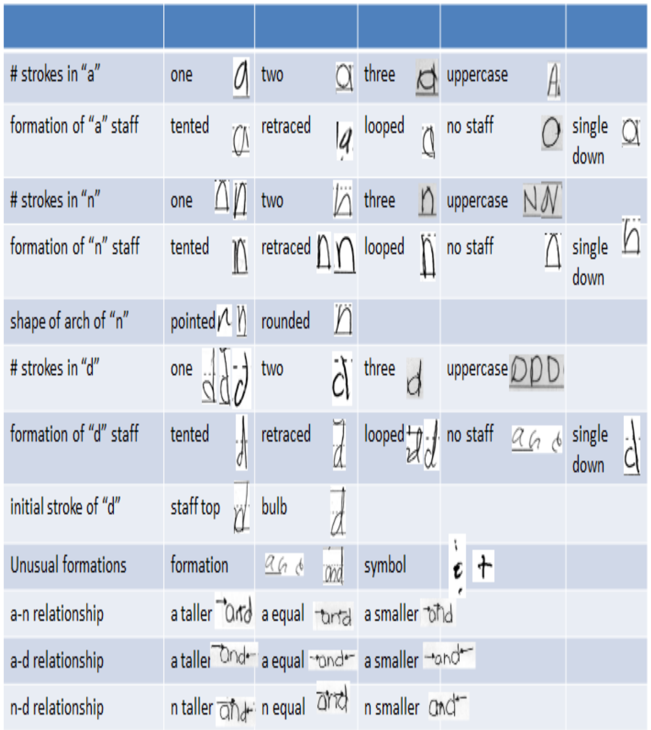
The cursive and hand print datasets have set of files containing "and" extracted data in them.

Snapshots of the feature vectors and the corresponding values are given in the next page.



Initial stroke of "a"	staff right	staff left	staff center	
Formation of "a" staff	tented	retraced	looped	no staff
Number of "n" arches	one	two		
Shape of "n" arches	pointed	rounded	retraced	combination
Location of "n" mid	above base	below base	at base	
Formation of "d" staff	tented	retraced	looped	
Formation of "d" initial	overhand	underhand	straight across	
Formation of "d" terminal	curved up	straight	curved down	no obvious end stroke
Symbol	unusual		symbol	
a-n relationship	a taller	a equal	a smaller	
a-d relationship	a taller	a equal	a smaller	
n-d relationship	n taller	n equal	n smaller	

Cursive data – 12 feature vectors (Column 1) and their corresponding values (Column 2 to Column 5)

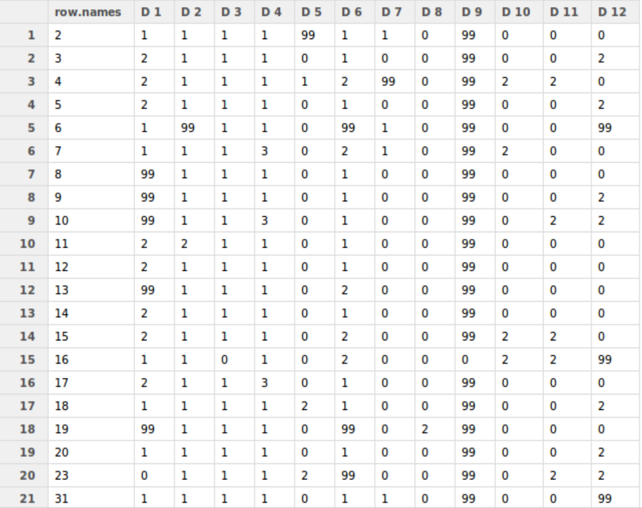


# strokes in "a"	one	two	three	uppercase	
formation of "a" staff	tented	retraced	looped	no staff	single down
# strokes in "n"	one	two	three	uppercase	
formation of "n" staff	tented	retraced	looped	no staff	single down
shape of arch of "n"	pointed	rounded			
# strokes in "d"	one	two	three	uppercase	
formation of "d" staff	tented	retraced	looped	no staff	single down
initial stroke of "d"	staff top	bulb			
Unusual formations	formation		symbol		
a-n relationship	a taller	a equal	a smaller		
a-d relationship	a taller	a equal	a smaller		
n-d relationship	n taller	n equal	n smaller		

Handprint data – 12 feature vectors (Column 1) and their corresponding values (Column 2 to Column 5)

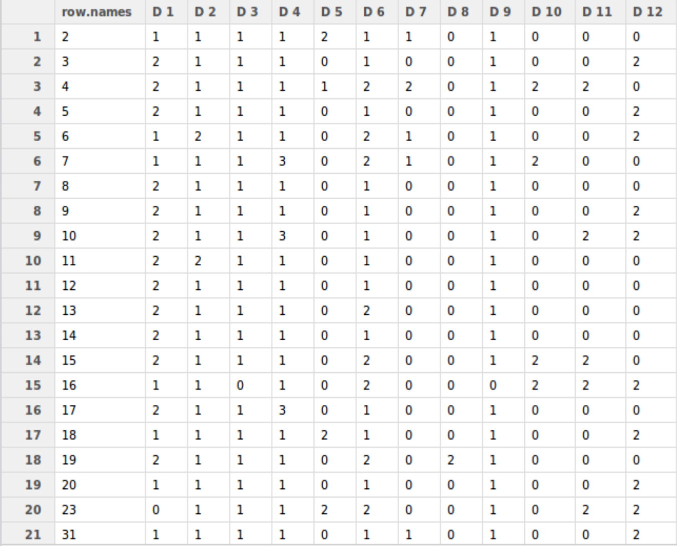
Data Cleaning

We notice that there are certain inconsistencies in the data provided. For certain entries we have insufficient data. For other entries we notice '99' which denotes the feature is inconsistent and '-1' for data not assigned. The data cleaning was made in all 12 features (ranging from D1 to D12) in such a way that these invalid values were replaced with legitimate ones in the range from 0 to 5. To achieve this functionality, we scan through the data values in each feature to identify these invalid values and then we find the maximum value (ranging from 0 to 5) present for this feature. Now the incorrect values are replaced by a value generated by computing (maximum value present for this feature) + 1. The same procedure is repeated for all the 12 features. By doing so, entire dataset contains values from 0 to 5.



	row.names	D 1	D 2	D 3	D 4	D 5	D 6	D 7	D 8	D 9	D 10	D 11	D 12
1	2	1	1	1	1	99	1	1	0	99	0	0	0
2	3	2	1	1	1	0	1	0	0	99	0	0	2
3	4	2	1	1	1	1	2	99	0	99	2	2	0
4	5	2	1	1	1	0	1	0	0	99	0	0	2
5	6	1	99	1	1	0	99	1	0	99	0	0	99
6	7	1	1	1	3	0	2	1	0	99	2	0	0
7	8	99	1	1	1	0	1	0	0	99	0	0	0
8	9	99	1	1	1	0	1	0	0	99	0	0	2
9	10	99	1	1	3	0	1	0	0	99	0	2	2
10	11	2	2	1	1	0	1	0	0	99	0	0	0
11	12	2	1	1	1	0	1	0	0	99	0	0	0
12	13	99	1	1	1	0	2	0	0	99	0	0	0
13	14	2	1	1	1	0	1	0	0	99	0	0	0
14	15	2	1	1	1	0	2	0	0	99	2	2	0
15	16	1	1	0	1	0	2	0	0	0	2	2	99
16	17	2	1	1	3	0	1	0	0	99	0	0	0
17	18	1	1	1	1	2	1	0	0	99	0	0	2
18	19	99	1	1	1	0	99	0	2	99	0	0	0
19	20	1	1	1	1	0	1	0	0	99	0	0	2
20	23	0	1	1	1	2	99	0	0	99	0	2	2
21	31	1	1	1	1	0	1	1	0	99	0	0	99

Sample data taken from Cursive writing dataset (Note the presence of 99 values in few rows)

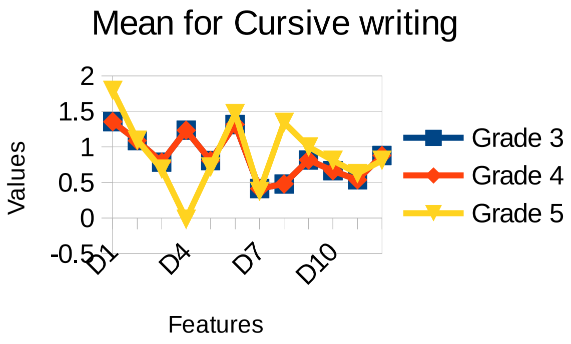


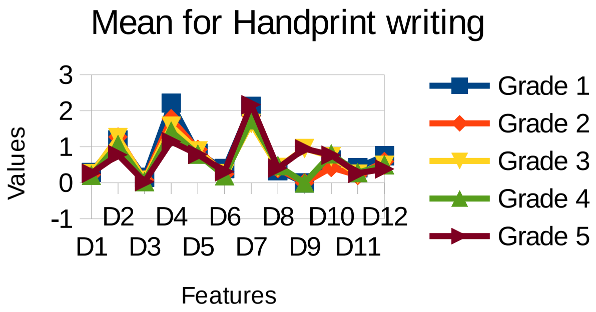
	row.names	D 1	D 2	D 3	D 4	D 5	D 6	D 7	D 8	D 9	D 10	D 11	D 12
1	2	1	1	1	1	2	1	1	0	1	0	0	0
2	3	2	1	1	1	0	1	0	0	1	0	0	2
3	4	2	1	1	1	1	2	2	0	1	2	2	0
4	5	2	1	1	1	0	1	0	0	1	0	0	2
5	6	1	2	1	1	0	2	1	0	1	0	0	2
6	7	1	1	1	3	0	2	1	0	1	2	0	0
7	8	2	1	1	1	0	1	0	0	1	0	0	0
8	9	2	1	1	1	0	1	0	0	1	0	0	2
9	10	2	1	1	3	0	1	0	0	1	0	2	2
10	11	2	2	1	1	0	1	0	0	1	0	0	0
11	12	2	1	1	1	0	1	0	0	1	0	0	0
12	13	2	1	1	1	0	2	0	0	1	0	0	0
13	14	2	1	1	1	0	1	0	0	1	0	0	0
14	15	2	1	1	1	0	2	0	0	1	2	2	0
15	16	1	1	0	1	0	2	0	0	0	2	2	2
16	17	2	1	1	3	0	1	0	0	1	0	0	0
17	18	1	1	1	1	2	1	0	0	1	0	0	2
18	19	2	1	1	1	0	2	0	2	1	0	0	0
19	20	1	1	1	1	0	1	0	0	1	0	0	2
20	23	0	1	1	1	2	2	0	0	1	0	2	2
21	31	1	1	1	1	0	1	1	0	1	0	0	2

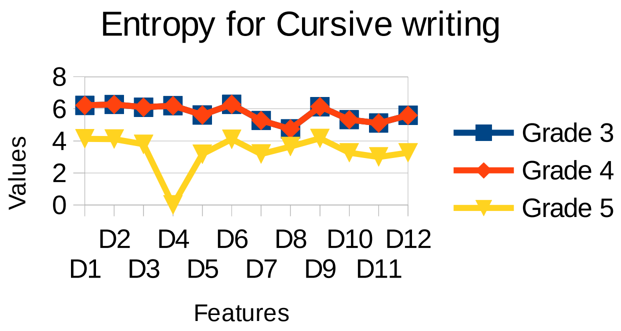
Cleaned data taken from Cursive writing dataset (Note that the value 99 is no longer present in any of the rows)

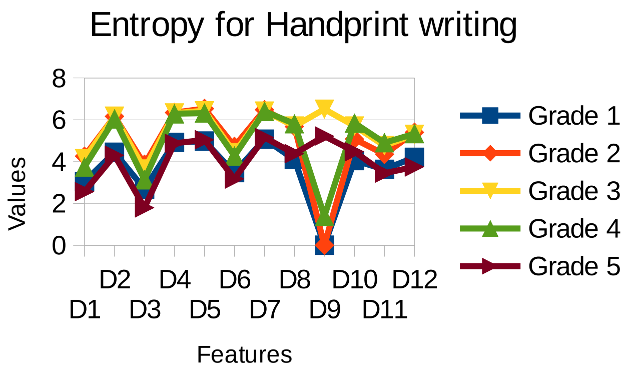
Mean and Entropy

Mean and Entropy are calculated for the cleaned dataset which are used for inference purpose from the constructed graphical models. Mean and entropy are calculated for each feature in both Cursive and handprint datasets.


Mean for Cursive writing


Mean for Handprint writing


Entropy for Cursive writing


Entropy for Handprint writing

Mean gives the average value for each feature of a dataset. We can observe from the results that the mean for a particular feature is shifting for each year.

The entropy measures the amount of information contained or the uncertainty in a random variable. From the observation of entropy samples of cursive and hand-printing data, we can infer that there is a slight variation in the handwriting styles over the years. Writing styles remain consistent from one year to the other.

Chi-Square Test

Pearson's Chi Square Test is applied on the datasets to find out the feature pairs to be considered for construction of Bayesian Network. The Chi square value is calculated for every combination of features in the dataset. This would give a 12*12 matrix as 12 features are present. The top 10 valued combinations are chosen and these will be used to construct the Bayesian Network. Chi Square test results are shown in the next page.

Bayesian Network Construction

Bayesian Network is a Probabilistic Graphical Model that speaks to the probabilistic connection between random variables. It is represented as a directed acyclic graph, where the random variables signify the nodes and the conditional dependency between these nodes are determined by the edges between them (which includes the conditional probability tables). The heading of the edge means the causality connoting the parent child relationship.

Bayesian Network is characterized by the blend of the DAG and an arrangement of CPDs. The diagram encodes the freedom suspicions, i.e every random variable is free of its non-descendant nodes provided its parents. The joint distribution is given by the formula below:

$$P_B(X_1,\ldots,X_n)=\prod_{i=1}^{n}P(X_i\mid \mathbf{Pa}_{X_i}).$$

Assumptions

- A node cannot have more than two parents, as it is computationally difficult to consider all possible combinations.
- The prior knowledge of the features can be used to obtain the co-relation between them. The mathematical representation of this dependency is given by Chi-square value and the goodness of the model is determined by the log-likelihood score.

Algorithm

Step-1: Apply Pearson's Chi-square test on the features (pairwise) to obtain a dependency matrix between the features

Step-2: Sort the matrix in descending order and pick the top 10 pairs, indicating pairs which have the highest dependency between them.

Step-3: Log-loss approach on the identified pairs to determine the directionality of dependency (parent and child relationship) which is detailed below:

Step-3.1: Find the frequency of each combination of values that a pair of features takes in the cleaned dataset. To find the frequency of a pair, count the number of times each combination of values are present in dataset and divide this figure with total sum of count of all possible combinations.
Eg: If features (D3,D4) are selected from Chi-square resultant matrix, find the frequency of (D3,D4)={(0,0),(0,1),(0,2),(0,3),(0,4),(1,0),...., (5,5)}, select only those value combinations that are present in cleaned dataset for this pair.

Step-3.2: Find the frequency of the second feature in the identified pair taking the value as taken by it in the pair.
Eg: If (D3,D4) pair takes a value (0,1), then calculate the frequency for D4 taking the value 1.

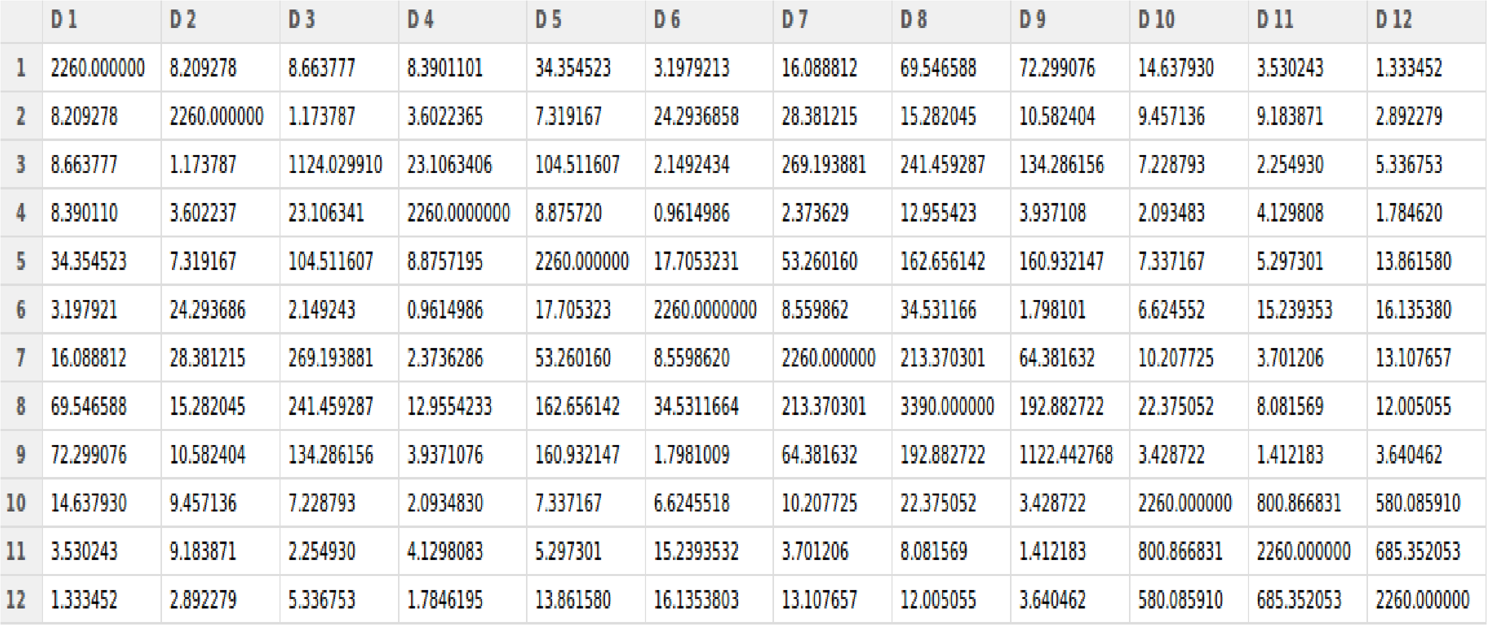
Step-3.3: Divide frequency obtained in Step 3.1 and Step 3.2 to obtain the Conditional probability distribution (CPD) for the identified feature pair

Step-3.4: Repeat steps 3.1, 3.2 and 3.3 until it covers all possible value combinations for an identified feature pair.

Step-3.5: Calculate the log-loss for the pair by taking the sum of logarithm to base 10 outcome of all the different values obtained in 3.3 to yield Log-Loss1.

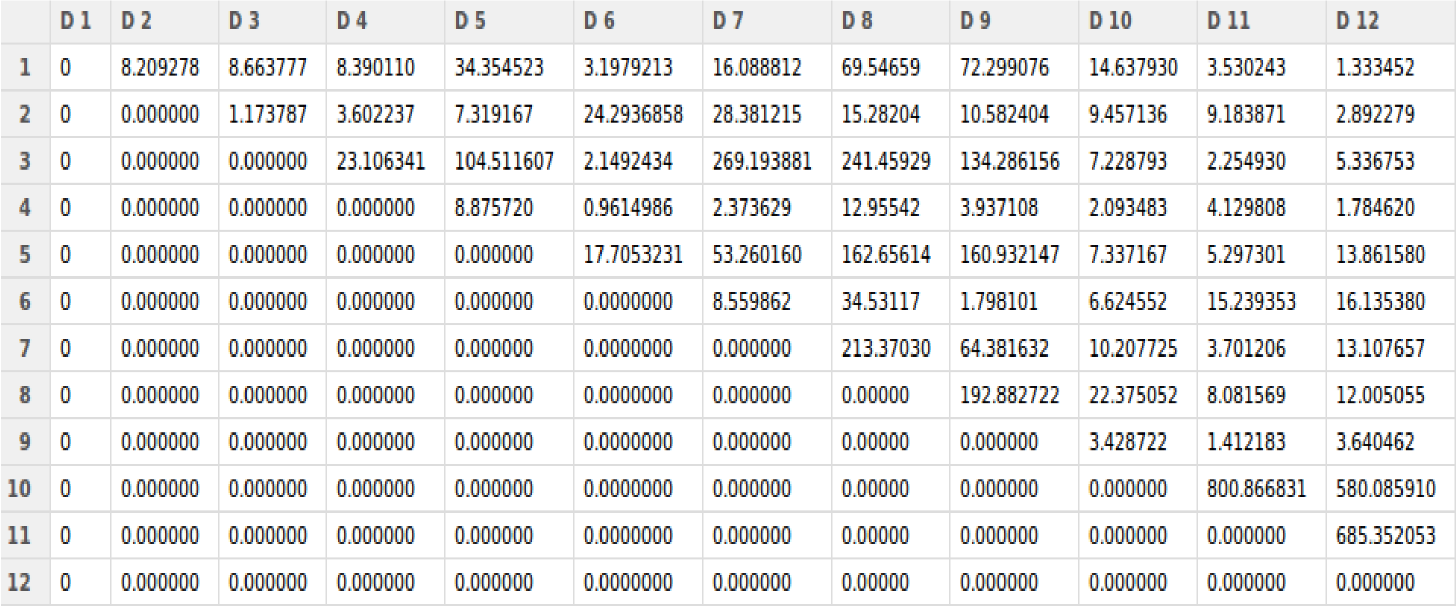
Step 3.6: Swap the features in the identified pair, (Eg: say (D3,D4) becomes (D4,D3) and second feature becomes D3 instead of D4) and repeat the procedure from Step 3.1 to Step 3.5 to yield Log-Loss2.

Step-3.7: If Log-Loss1<Log-Loss2, then feature 2 is the child of feature 1(Eg: D3->D4) Else, feature 1 is the child of feature 2
(Eg:D4->D3).



	D 1	D 2	D 3	D 4	D 5	D 6	D 7	D 8	D 9	D 10	D 11	D 12
1	2260.000000	8.209278	8.663777	8.3901101	34.354523	3.1979213	16.088812	69.546588	72.299076	14.637930	3.530243	1.333452
2	8.209278	2260.000000	1.173787	3.6022365	7.319167	24.2936858	28.381215	15.282045	10.582404	9.457136	9.183871	2.892279
3	8.663777	1.173787	1124.029910	23.1063406	104.511607	2.1492434	269.193881	241.459287	134.286156	7.228793	2.254930	5.336753
4	8.390110	3.602237	23.106341	2260.0000000	8.875720	0.9614986	2.373629	12.955423	3.937108	2.093483	4.129808	1.784620
5	34.354523	7.319167	104.511607	8.8757195	2260.000000	17.7053231	53.260160	162.656142	160.932147	7.337167	5.297301	13.861580
6	3.197921	24.293686	2.149243	0.9614986	17.705323	2260.0000000	8.559862	34.531166	1.798101	6.624552	15.239353	16.135380
7	16.088812	28.381215	269.193881	2.3736286	53.260160	8.5598620	2260.000000	213.370301	64.381632	10.207725	3.701206	13.107657
8	69.546588	15.282045	241.459287	12.9554233	162.656142	34.5311664	213.370301	3390.000000	192.882722	22.375052	8.081569	12.005055
9	72.299076	10.582404	134.286156	3.9371076	160.932147	1.7981009	64.381632	192.882722	1122.442768	3.428722	1.412183	3.640462
10	14.637930	9.457136	7.228793	2.0934830	7.337167	6.6245518	10.207725	22.375052	3.428722	2260.000000	800.866831	580.085910
11	3.530243	9.183871	2.254930	4.1298083	5.297301	15.2393532	3.701206	8.081569	1.412183	800.866831	2260.000000	685.352053
12	1.333452	2.892279	5.336753	1.7846195	13.861580	16.1353803	13.107657	12.005055	3.640462	580.085910	685.352053	2260.000000

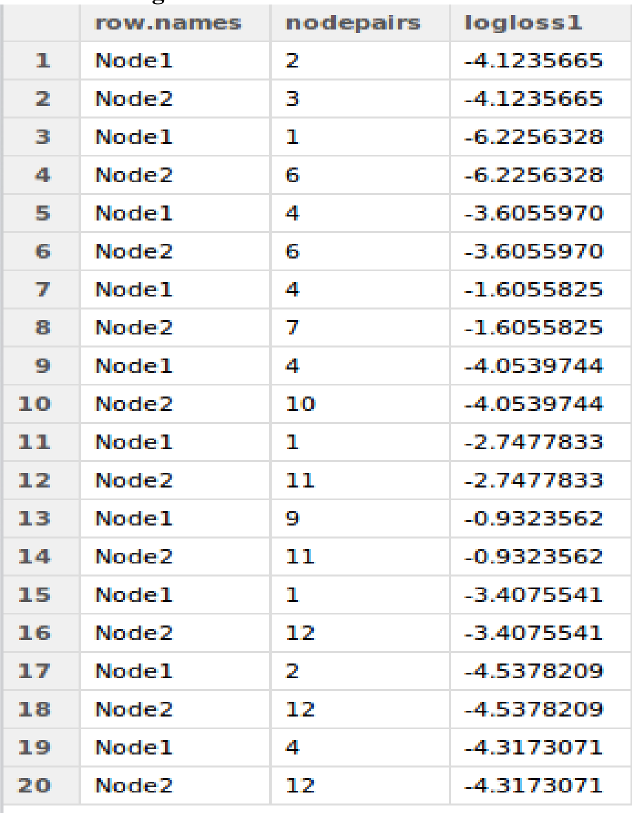
Chi Squared matrix for features of Cursive dataset



	D 1	D 2	D 3	D 4	D 5	D 6	D 7	D 8	D 9	D 10	D 11	D 12
1	0	8.209278	8.663777	8.390110	34.354523	3.1979213	16.088812	69.54659	72.299076	14.637930	3.530243	1.333452
2	0	0.000000	1.173787	3.602237	7.319167	24.2936858	28.381215	15.28204	10.582404	9.457136	9.183871	2.892279
3	0	0.000000	0.000000	23.106341	104.511607	2.1492434	269.193881	241.45929	134.286156	7.228793	2.254930	5.336753
4	0	0.000000	0.000000	0.000000	8.875720	0.9614986	2.373629	12.95542	3.937108	2.093483	4.129808	1.784620
5	0	0.000000	0.000000	0.000000	0.000000	17.7053231	53.260160	162.65614	160.932147	7.337167	5.297301	13.861580
6	0	0.000000	0.000000	0.000000	0.000000	0.0000000	8.559862	34.53117	1.798101	6.624552	15.239353	16.135380
7	0	0.000000	0.000000	0.000000	0.000000	0.0000000	0.000000	213.37030	64.381632	10.207725	3.701206	13.107657
8	0	0.000000	0.000000	0.000000	0.000000	0.0000000	0.000000	0.00000	192.882722	22.375052	8.081569	12.005055
9	0	0.000000	0.000000	0.000000	0.000000	0.0000000	0.000000	0.00000	0.000000	3.428722	1.412183	3.640462
10	0	0.000000	0.000000	0.000000	0.000000	0.0000000	0.000000	0.00000	0.000000	0.000000	800.866831	580.085910
11	0	0.000000	0.000000	0.000000	0.000000	0.0000000	0.000000	0.00000	0.000000	0.000000	0.000000	685.352053
12	0	0.000000	0.000000	0.000000	0.000000	0.0000000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000

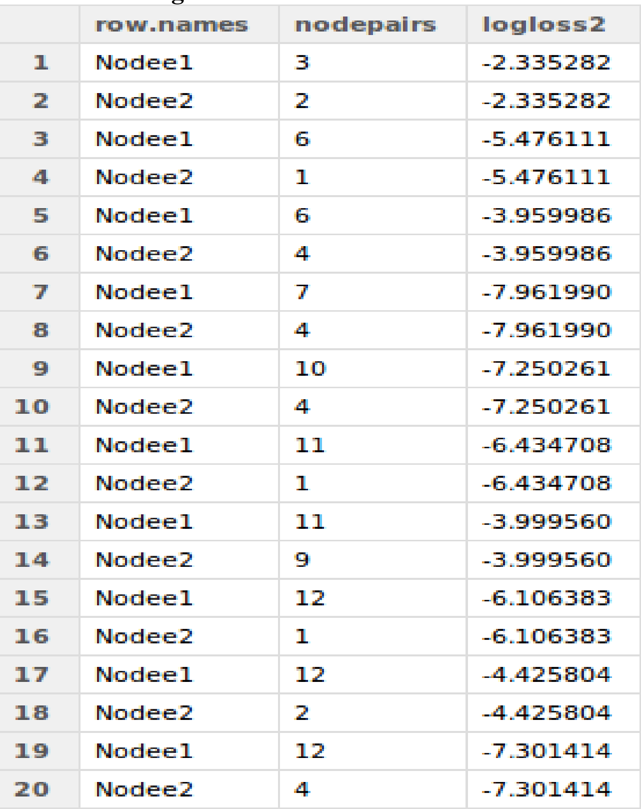
Chi Squared matrix for features of Cursive after obtaining top 10 combination values.

Log Loss 1 result for cursive dataset



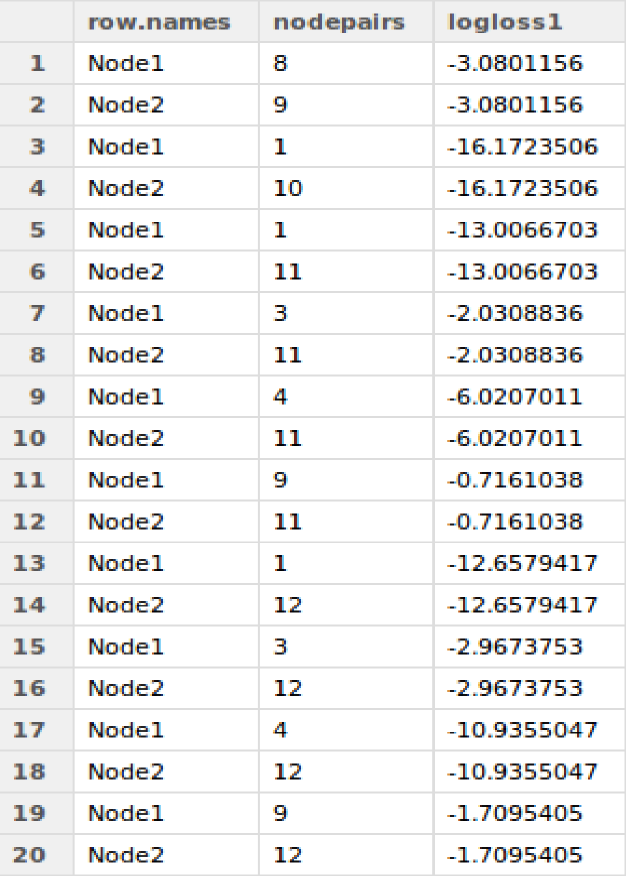
	row.names	nodepairs	logloss1
1	Node1	2	-4.1235665
2	Node2	3	-4.1235665
3	Node1	1	-6.2256328
4	Node2	6	-6.2256328
5	Node1	4	-3.6055970
6	Node2	6	-3.6055970
7	Node1	4	-1.6055825
8	Node2	7	-1.6055825
9	Node1	4	-4.0539744
10	Node2	10	-4.0539744
11	Node1	1	-2.7477833
12	Node2	11	-2.7477833
13	Node1	9	-0.9323562
14	Node2	11	-0.9323562
15	Node1	1	-3.4075541
16	Node2	12	-3.4075541
17	Node1	2	-4.5378209
18	Node2	12	-4.5378209
19	Node1	4	-4.3173071
20	Node2	12	-4.3173071

Log Loss 2 results for cursive dataset



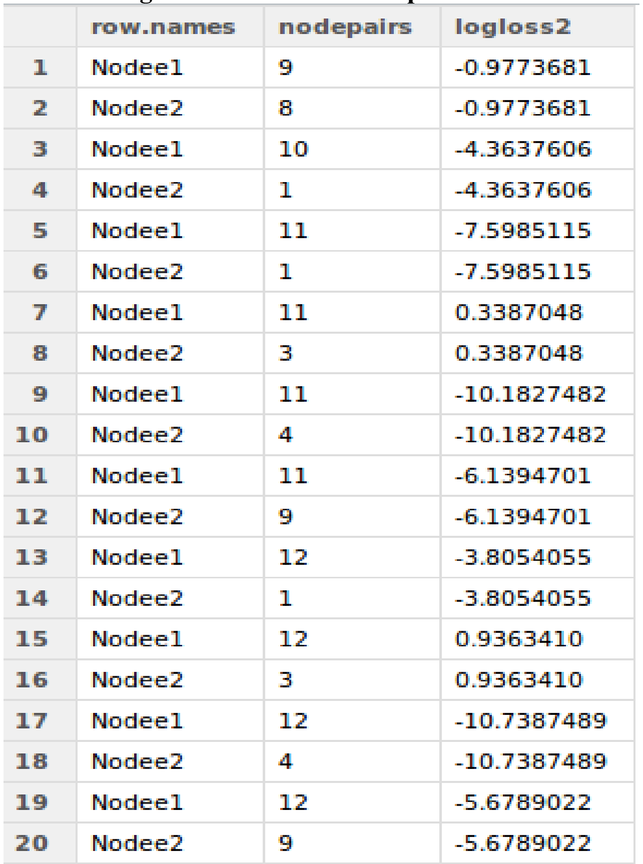
	row.names	nodepairs	logloss2
1	Nodee1	3	-2.335282
2	Nodee2	2	-2.335282
3	Nodee1	6	-5.476111
4	Nodee2	1	-5.476111
5	Nodee1	6	-3.959986
6	Nodee2	4	-3.959986
7	Nodee1	7	-7.961990
8	Nodee2	4	-7.961990
9	Nodee1	10	-7.250261
10	Nodee2	4	-7.250261
11	Nodee1	11	-6.434708
12	Nodee2	1	-6.434708
13	Nodee1	11	-3.999560
14	Nodee2	9	-3.999560
15	Nodee1	12	-6.106383
16	Nodee2	1	-6.106383
17	Nodee1	12	-4.425804
18	Nodee2	2	-4.425804
19	Nodee1	12	-7.301414
20	Nodee2	4	-7.301414

Log Loss 1 result for handprint dataset



	row.names	nodepairs	logloss1
1	Node1	8	-3.0801156
2	Node2	9	-3.0801156
3	Node1	1	-16.1723506
4	Node2	10	-16.1723506
5	Node1	1	-13.0066703
6	Node2	11	-13.0066703
7	Node1	3	-2.0308836
8	Node2	11	-2.0308836
9	Node1	4	-6.0207011
10	Node2	11	-6.0207011
11	Node1	9	-0.7161038
12	Node2	11	-0.7161038
13	Node1	1	-12.6579417
14	Node2	12	-12.6579417
15	Node1	3	-2.9673753
16	Node2	12	-2.9673753
17	Node1	4	-10.9355047
18	Node2	12	-10.9355047
19	Node1	9	-1.7095405
20	Node2	12	-1.7095405

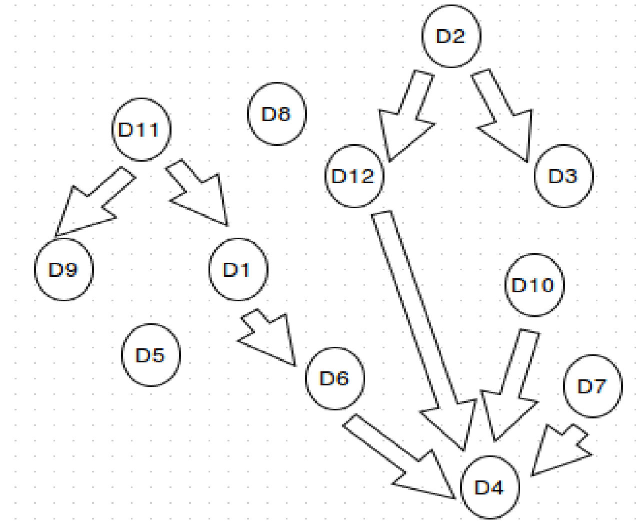
Log Loss 2 result for handprint dataset



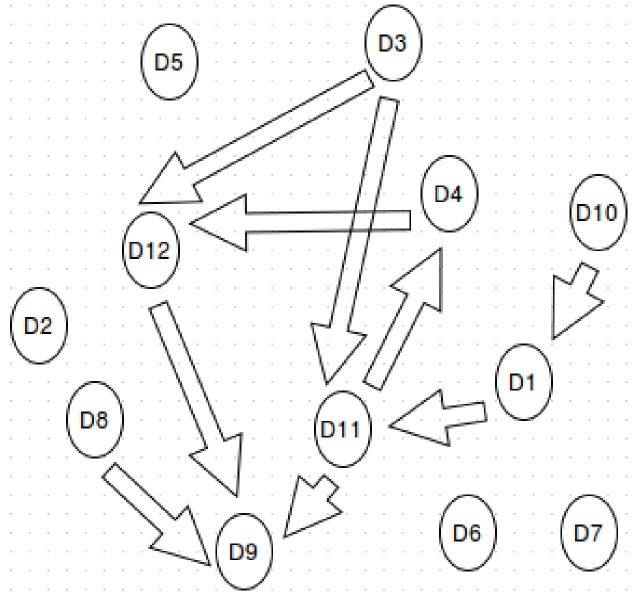
	row.names	nodepairs	logloss2
1	Nodee1	9	-0.9773681
2	Nodee2	8	-0.9773681
3	Nodee1	10	-4.3637606
4	Nodee2	1	-4.3637606
5	Nodee1	11	-7.5985115
6	Nodee2	1	-7.5985115
7	Nodee1	11	0.3387048
8	Nodee2	3	0.3387048
9	Nodee1	11	-10.1827482
10	Nodee2	4	-10.1827482
11	Nodee1	11	-6.1394701
12	Nodee2	9	-6.1394701
13	Nodee1	12	-3.8054055
14	Nodee2	1	-3.8054055
15	Nodee1	12	0.9363410
16	Nodee2	3	0.9363410
17	Nodee1	12	-10.7387489
18	Nodee2	4	-10.7387489
19	Nodee1	12	-5.6789022
20	Nodee2	9	-5.6789022

Note that, the Log-loss tables contain top 10 feature pairs which have the maximum dependency in the Chi-square output.

The Bayesian network constructed for Cursive dataset based on the above algorithm is shown below:



The Bayesian network constructed for hand-print dataset based on the above algorithm is shown below:



Cursive writing Bayesian network (Grade 3+Grade 4+Grade 5) Inference:

We can observe the direct dependency between feature 12 (n-d relationship) and feature 4 (Shape of “n” arches) and another direct dependency between feature 12 and feature 1 (Initial stroke of “a”). Also, there exists a dependency between feature 1 to feature 6 (Formation of “d” staff) and from feature 6 to feature 4. Hence, the relationships between these 4 features constitute that important specifications of each “a”, “n” and “d” letters are important to analyze the whole word “and”.

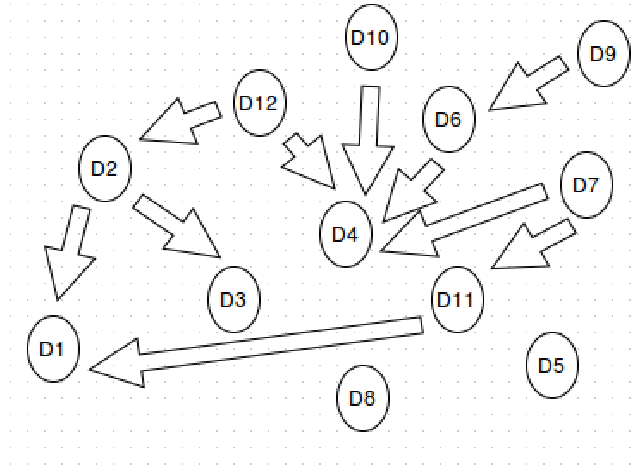
On a contrary, less significant features like feature 5(Location of “n” mid) and feature 8(Formation of “d” terminal) are independent.

Hand-print writing Bayesian network (Grade 1+ Grade 2+ Grade 3+Grade 4+Grade 5) Inference:

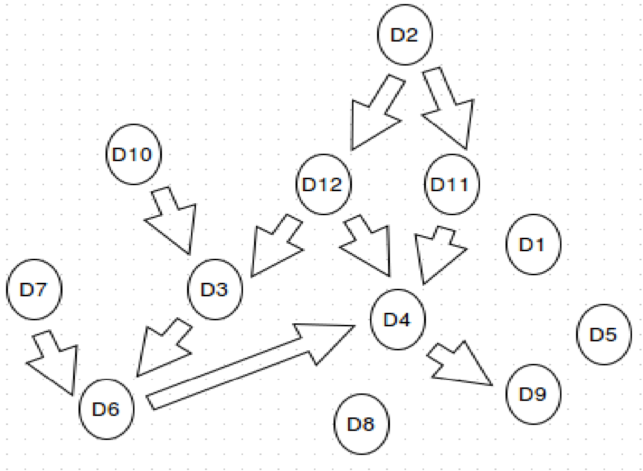
- It is observant that the feature 11(a-d relationship) is dependent on feature 4(formation of “n” staff) which is in turn dependent on feature 12(n-d relationship). Hence, a-d relationship is determined through n-d relationship

Dynamic Bayesian Network

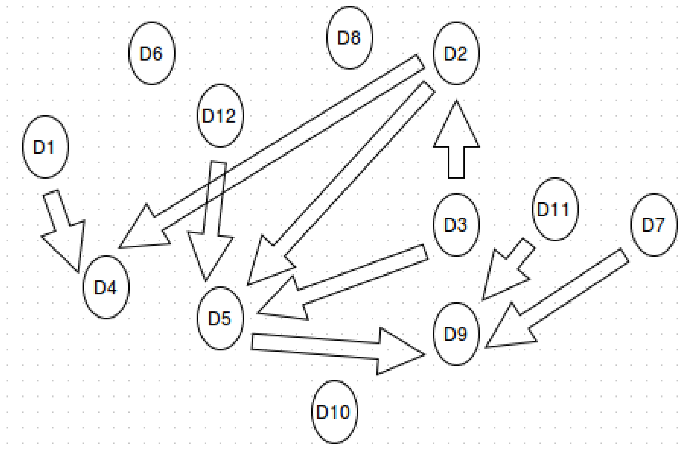
A Dynamic Bayesian network is a Bayesian network that relates the variables to each other in a progressive time step manner. As the dataset is progressive over the years, we can construct the Bayesian networks for Grade 3, Grade 4 with respect to Cursive writing and for Grade 3, Grade 4 and Grade 5 with respect to Hand-print writing. The transition in dependency between the features can be noted from these figures.



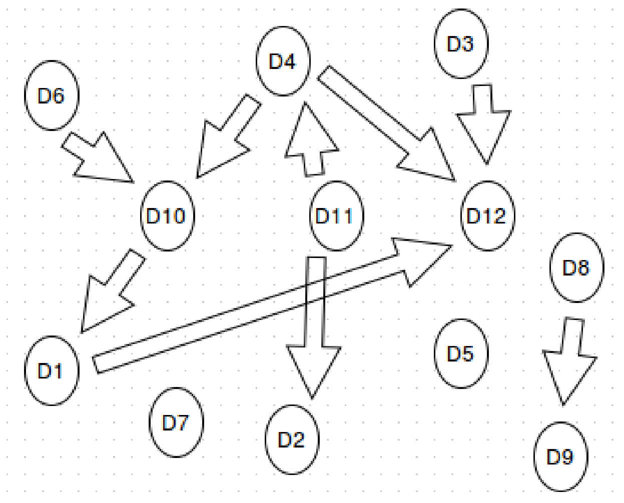
Cursive Grade 3



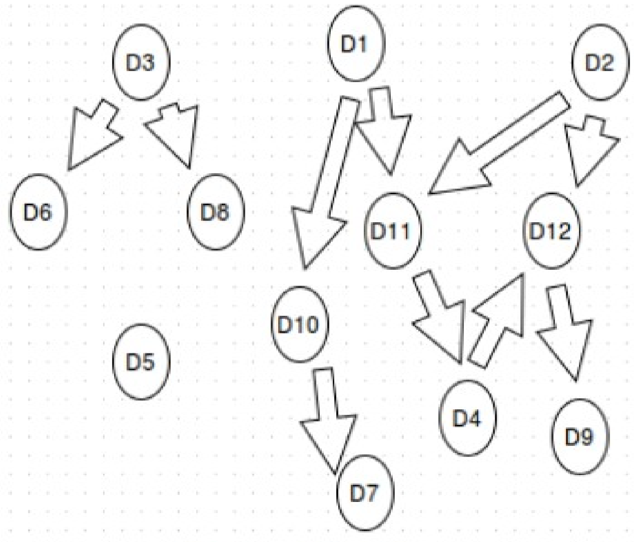
Cursive Grade 4



Handprint Grade 5



Handprint Grade 3



Handprint Grade 4

Markov Network Construction

The network construction method aims at constructing a Markov distribution with the highest entropy. The Kullback-Leibler cross entropy is a measure of divergence between two probability distributions.

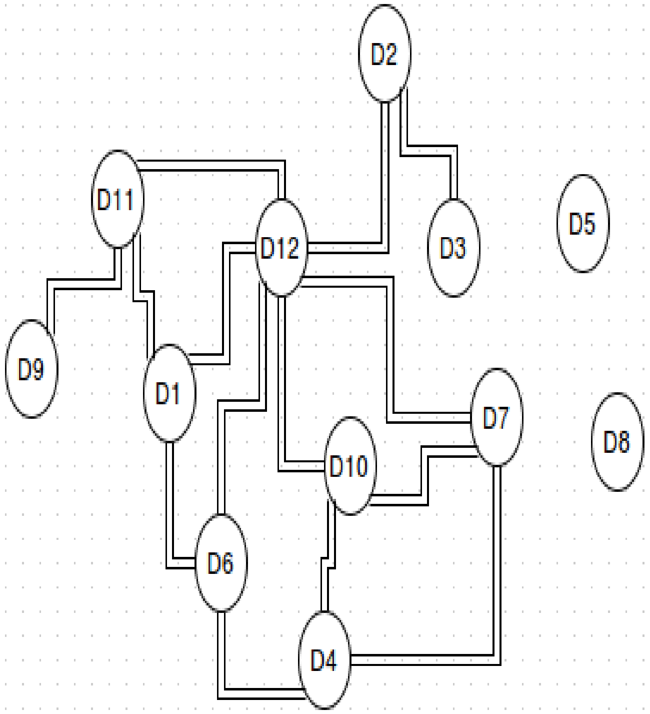
$$I(p,p') = \sum_{\mathbf{c}} p(\mathbf{c}) \log \frac{p(\mathbf{c})}{p'(\mathbf{c})}$$

where c is the configuration set of variables
From the above equation, minimizing the closeness metric(cross entropy, I(p,p’)) is equivalent to minimizing the entropy H(p’).

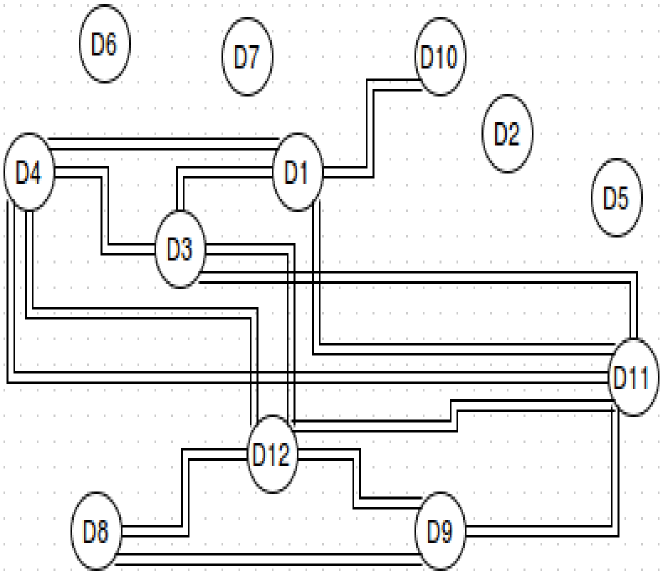
Moralization

A simple method to construct a Markov network is by moralizing the Bayesian Network, by maintaining the same structure and eliminating the directionality and adding extra edge between nodes having common child.
The existent Bayesian network is transformed into a Markov network except for the fact that directed edges are converted to non-directed edges and an extra edge is added between those pair of nodes which are having a common child.

The below graphs demonstrate the Markov networks constructed by applying the Moralization technique to Cursive Bayesian and Hand-print Bayesian networks:



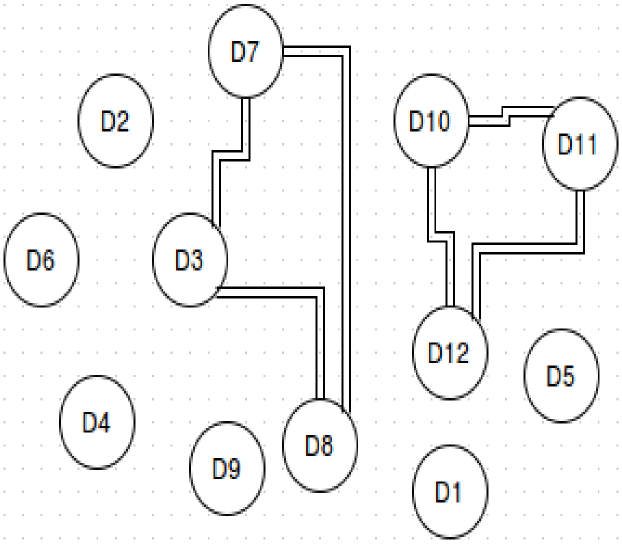
Cursive Markov network



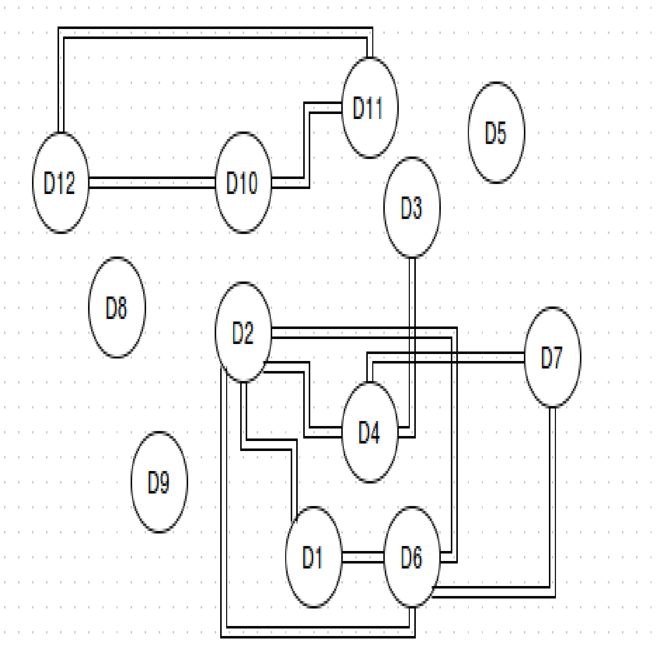
Handprint Markov network

Glasso Threshold approach to construct Markov Network

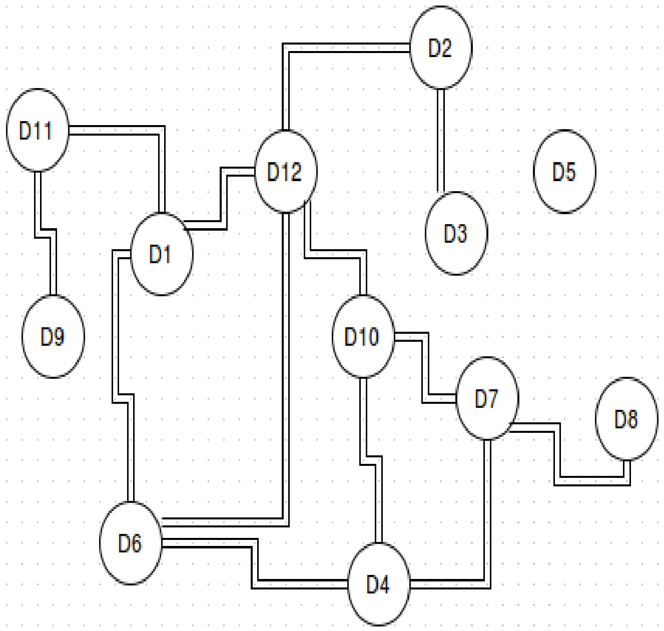
Initially, we assume all the variables to be independent, i.e. there are no edges between any nodes. We make use of graphical packages such as Glasso in association with the dependency(Co-variance) matrix to construct the Markov network. Here, we take the inverse of the Co-variance matrix to get the matrix containing values that indicate the divergence between the features. Now we set a threshold value in the range 0.01 to 0.04 and then consider all the feature pairs which are having a value within this threshold range for Markov network construction. Setting the threshold is important and generally set according to the size of the dataset. Smaller the dataset larger the threshold, this is to handle the erroneous samples in the small dataset, so using a higher threshold can suppress false dependencies.



Cursive Markov network using glasso threshold approach



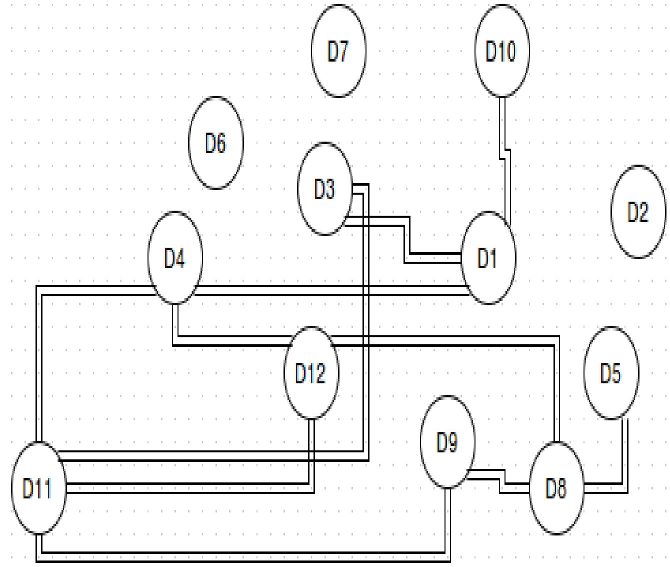
Handprint Markov network using glasso threshold approach



Cursive Markov network using KL divergence approach

KL Divergence approach to construct the Markov network

In this approach, we compute the Kullback-Leiber divergence for the given dataset as a pairwise comparison of features. We calculate the Probability Mass functions for the random variables and pass these frequencies as arguments to KL divergence computation. Note that, the mass functions are computed by taking the count of random variables taking different values over the the sum of all possibilities. The pairwise computation of probability mass functions and KL divergence provides us with the connection between nodes in the resultant Markov network.



Handprint Markov network using KL divergence approach

References

- Efficient and Accurate Learning of Bayesian Networks using Chi-Squared Independence Tests, Yi Tang and Sargur N. Srihari, SUNY Buffalo
- Bayesian Network Structure Learning and Inference Methods for Handwriting, Mukta Puri, Sargur N.Srihari and Yi Tang, SUNY Buffalo
- Bayesian Network Structure Learning Using Causality, Zhen Xu and Sargur N. Srihari, SUNY Buffalo
- Missing Value Imputation: With Application to Handwriting Data, Zhen Xu and Sargur N.Srihari, SUNY Buffalo