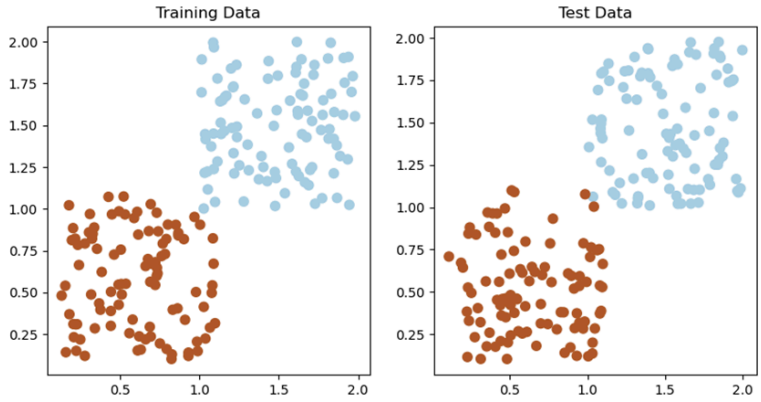
Advanced Artificial Intelligence, 2425

Workshop Task: Quiz on Gaussian Bayes Nets

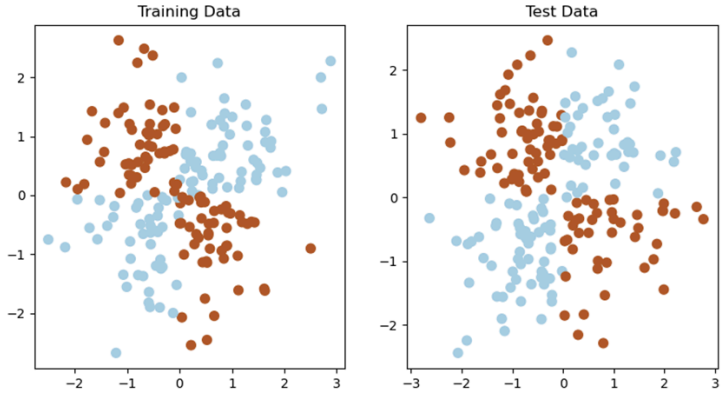
Question 1

30 points

Linearly separable data:



Non-linearly separable data:



Using the Toy Data discussed during this week's lecture, shown above and available in your workshop materials, train the following Gaussian Bayesian networks (GBNs):

1. GBN using linearly separable data with RIDGE regression.
2. GBN using linearly separable data with LASSO regression.
3. GBN using non-linearly separable data with RIDGE regression.
4. GBN using non-linearly separable data with LASSO regression.
5. GBN using non-linearly separable data with KERNEL RIDGE regression.

You can use the following programs to train those GBNs and evaluate their performance:

- **`python PDF_Generator.py .\config\config-toydata-pdf.txt .\data\data-linearlyseparable-train.csv`**
- **`python ModelEvaluator.py .\config\config-toydata-pdf.txt .\data\data-linearlyseparable-test.csv`**

You will need to edit `PDF_Generator.py` to change the type of regressor. Note that when you run this program, it will store the corresponding regression model in a PKL file with the same name as the config file. Such a PKL file will be loaded by `ModuleEvaluator.py` (and associated files) for probabilistic inference using linear Gaussians. In your own time, you should have a look a those programs in more detail for further understanding.

What is the performance of GBN 1? Balanced Accuracy=[Blank 1], KL-Divergence=[Blank 2]
What is the performance of GBN 2? Balanced Accuracy=[Blank 3], KL-Divergence=[Blank 4]
What is the performance of GBN 3? Balanced Accuracy=[Blank 5], KL-Divergence=[Blank 6]
What is the performance of GBN 4? Balanced Accuracy=[Blank 7], KL-Divergence=[Blank 8]
What is the performance of GBN 5? Balanced Accuracy=[Blank 9], KL-Divergence=[Blank 10]

Please provide your answers with up to two decimals and without rounding, and make sure you save your changes to programs and also that you use the appropriate training and test files.

Now, your config file above assumes a simple (Naïve Bayes) structure. Change it as follows:
P(target);P(x1|target);P(x2|target,x1)

Save the newly edited config file and re-train and re-evaluate your GBNs.

What is the performance of GBN 3 w/new structure? Balanced Accuracy=[Blank 11], KL-Divergence=[Blank 12]
What is the performance of GBN 4 w/new structure? Balanced Accuracy=[Blank 13], KL-Divergence=[Blank 14]
What is the performance of GBN 5 w/new structure? Balanced Accuracy=[Blank 15], KL-Divergence=[Blank 16]

Overall, did the less naïve structure improve the performance of GBNs? Answer(Y/N)=[Blank 17]

Blank 1

Blank 2

Blank 3

Blank 4

Blank 5

Blank 6

Blank 7

Blank 8

Blank 9

Blank 10

Blank 11

Blank 12

Blank 13

Blank 14

Blank 15

Blank 16

Blank 17

Question 2

30 points

Snapshot of the Bank Note Authentication data (from the UCI Machine Learning repository (https://archive.ics.uci.edu/dataset/267/banknote+authentication)):

	A	B	C	D	E
1	x1	x2	x3	x4	target
2	3.6216	8.6661	-2.8073	-0.44699	0
3	4.5459	8.1674	-2.4586	-1.4621	0
4	3.866	-2.6383	1.9242	0.10645	0
5	3.4566	9.5228	-4.0112	-3.5944	0
6	0.32924	-4.4552	4.5718	-0.9888	0
625	-1.3971	3.3191	-1.3927	-1.9948	1
626	-1.6677	-7.1535	7.8929	0.96765	1
627	-3.8483	-12.8047	15.6824	-1.281	1
628	-3.5681	-8.213	10.083	0.96765	1
629	-2.2804	-0.30626	1.3347	1.3763	1

Using the continuous data above (available in your workshop materials), train the following Gaussian Bayesian networks (GBNs):

1. GBN using non-linearly separable data with RIDGE regression.
2. GBN using non-linearly separable data with LASSO regression.
3. GBN using non-linearly separable data with KERNEL RIDGE regression.

Similarly as before, you can use the following programs to train and evaluate those GBNs:

- **`python PDF_Generator.py .\config\config-banknote-pdf.txt .\data\data_banknote_authentication-train.csv`**
- **`python ModelEvaluator.py .\config\config-banknote-pdf.txt .\data\data_banknote_authentication-test.csv`**

You will need to edit `PDF_Generator.py` to change the type of regressor.

What is the performance of GBN 1? Balanced Accuracy=[Blank 1], KL-Divergence=[Blank 2]
What is the performance of GBN 2? Balanced Accuracy=[Blank 3], KL-Divergence=[Blank 4]
What is the performance of GBN 3? Balanced Accuracy=[Blank 5], KL-Divergence=[Blank 6]

Please provide your answers with up to two decimals and without rounding, and don't forget to save your changes to programs and also that you use the appropriate training and test files.

Since your config file above assumes a simple (Naive Bayes) structure, you should use your knowledge on structure learning to induce the structure from training data. To do that, you can use **`bnlearn_StructureLearning-bankdata.py`**. Update the config file with the learnt structure and re-train and re-evaluate your GBNs.

What is the performance of GBN 4 w/learnt structure? Balanced Accuracy=[Blank 7], KL-Divergence=[Blank 8]
What is the performance of GBN 5 w/learnt structure? Balanced Accuracy=[Blank 9], KL-Divergence=[Blank 10]
What is the performance of GBN 6 w/learnt structure? Balanced Accuracy=[Blank 11], KL-Divergence=[Blank 12]

Did the learnt structure improve the performance of GBNs? Answer(Y/N)=[Blank 13]

Blank 1

Blank 2

Blank 3

Blank 4

Blank 5

Blank 6

Blank 7

Blank 8

Blank 9

Blank 10

Blank 11

Blank 12

Blank 13

Question 3

40 points

Snapshot of the Cardiovascular disease data (from a Kaggle repository (https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset)):

	A	B	C	D	E	F	G	H	I	J	K	L
1	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	target
2	18393	2	168	62	110	80	1	1	0	0	1	0
3	20228	1	156	85	140	90	3	1	0	0	1	1
4	18857	1	165	64	130	70	3	1	0	0	0	1
5	21914	1	151	67	120	80	2	2	0	0	0	0
6	22113	1	157	93	130	80	3	1	0	0	1	0
7	22584	2	178	95	130	90	3	3	0	0	1	1
8	17668	1	158	71	110	70	1	1	0	0	1	0
9	19834	1	164	68	110	60	1	1	0	0	0	0
10	22530	1	169	80	120	80	1	1	0	0	1	0

This is a more challenging dataset than the previous two due to having much more training and test data points. Using the data above (available in your workshop materials), train the following Gaussian Bayesian networks (GBNs):

1. GBN using non-linearly separable data with RIDGE regression.
2. GBN using non-linearly separable data with LASSO regression.
3. GBN using non-linearly separable data with KERNEL RIDGE regression.

Similarly as before, you can use the following programs to train and evaluate those GBNs:

- **`python PDF_Generator.py .\config\config-cardiovascular-pdf.txt .\data\cardiovascular_data-original-train.csv`**
- **`python ModelEvaluator.py .\config\config-cardiovascular-pdf.txt .\data\cardiovascular_data-original-test.csv`**

Recall that you will need to edit `PDF_Generator.py` to change the type of regressor. Please make sure you do the targeted changes, save your edits, and that you use the appropriate training and test files.

What is the performance of GBN 1? Balanced Accuracy=[Blank 1], KL-Divergence=[Blank 2]
What is the performance of GBN 2? Balanced Accuracy=[Blank 3], KL-Divergence=[Blank 4]
What is the performance of GBN 3? Balanced Accuracy=[Blank 5], KL-Divergence=[Blank 6]

Please provide your answers with up to two decimals and without rounding.

Your config file above assumes a simple (Naïve Bayes) structure. You are required to use your knowledge on structure learning to induce the structure from training data. This program should be written by yourself, which requires small changes to the corresponding example program from last week. Update the config file with the learnt structure obtained and proceed re-training and re-evaluating your GBNs.

Please be aware of the data restriction in PDF_Generator.py when training GBNs with the KernelRidge choice, which only makes use of 2K training examples for that particular regressor instead of the whole training data. This is due to higher computational requirements than the other two choices of regressor (Ridge and Lasso).

What is the performance of GBN 4 w/learnt structure? Balanced Accuracy=[Blank 7], KL-Divergence=[Blank 8]
What is the performance of GBN 5 w/learnt structure? Balanced Accuracy=[Blank 9], KL-Divergence=[Blank 10]
What is the performance of GBN 6 w/learnt structure? Balanced Accuracy=[Blank 11], KL-Divergence=[Blank 12]

Did the learnt structure improved the performance of GBNs? Answer(Y/N)=[Blank 13]

Blank 1

Blank 2

Blank 3

Blank 4

Blank 5

Blank 6

Blank 7

Blank 8

Blank 9

Blank 10

Blank 11

Blank 12

Blank 13