1 SVM vs. Neural Networks

1.1 Data A

From Data A [2], I choose datasets named "a1a", "a2a", "a3a", "a4a", "a5a", "a6a", "a7a", "a9a" (without "a8a") since they are all from the same source as well as have monotonically increasing training size and monotonically decreasing testing size from "a1a" to "a9a".

1.1.1 Comparison between SVM and MLP

SVM is conducted through functions from **libsvm** [1] and MLP through functions from **sklearn**. Here SVM and MLP are performed on all above datasets with following configurations:

- SVM type: C-SVC.
- Kernel function: linear.
- Degree in kernel function: 3.
- Hidden layer sizes: two layers with 5 and 2 units respectively.

The results are in Table 1. We can see that, with the increase of training size and decrease of testing size, the performances of SVM and MLP improve. The improvement of SVM's performance is not as obvious as that of MLP's.

Table 1: Comparison of SVM and MLP

dataset	training size	testing size	SVM	MLP
a1a	1605	30956	83.81%	62.46%
a2a	2265	30296	84.28%	62.89%
a3a	3185	29376	84.32%	63.26%
a4a	4781	27780	84.45%	66.24%
a5a	6414	26147	84.40%	67.36%
a6a	11220	21341	84.72%	68.46%
a7a	16100	16461	84.85%	69.14%
a8a	32561	16281	84.98%	69.90%

1.1.2 Comparison over different SVM kernel types

SVM is conducted through functions from **libsvm**. Here SVM is performed on all above datasets under different kernel types (linear, polynomial, radial basis function, sigmoid) with following configurations:

- SVM type: C-SVC.
- Degree in kernel function: 3.
- Cost value (parameter C): 1.

The results are in Table 2. The performances of them show an increasing trend with the increase of training size and decrease of testing size. Among these kernel functions, polynomial function has the worst performance. Three others don't show an obvious difference with current settings.

Table 2: Comparison over different SVM kernel types

dataset	training size	testing size	linear	polynomial	radial basis function	sigmoid
a1a	1605	30956	83.81%	75.95%	83.59%	82.12%
a2a	2265	30296	84.28%	76.01%	83.98%	83.32%
a3a	3185	29376	84.32%	75.94%	83.84%	83.39%
a4a	4781	27780	84.45%	76.05%	83.96%	83.72%
a5a	6414	26147	84.40%	76.01%	84.17%	84.04%
a6a	11220	21341	84.72%	75.87%	84.17%	84.03%
a7a	16100	16461	84.85%	76.17%	84.58%	84.49%
a8a	32561	16281	84.98%	76.38%	84.82%	84.91%

1.1.3 Comparison over different cost value in SVM

SVM is conducted through functions from **libsvm**. Here SVM is performed on all above datasets under different cost value (0.001, 0.01, 0.1, 1, 10) with following configurations:

- SVM type: C-SVC.
- Kernel function
- Degree in kernel function: 3.

The results are in Table 3. The performances of them show a slightly increasing trend with the increase of training size and decrease of testing size. The performance doesn't improve with increasing cost value (parameter C). When $C = 0.1$ under current configurations, the performance is the best among them.

1.2 Data B

From Data B, I choose MNIST dataset [3]. SVM is conducted by **sklearn** with following settings:

Table 3: Comparison over different cost value in SVM

dataset	training size	testing size	0.001	0.01	0.1	1	10
a1a	1605	30956	75.95%	82.84%	84.31%	83.81%	83.76%
a2a	2265	30296	76.01%	83.55%	84.60%	84.28%	84.03%
a3a	3185	29376	75.94%	83.74%	84.50%	84.32%	84.07%
a4a	4781	27780	76.05%	83.86%	84.59%	84.45%	84.48%
a5a	6414	26147	76.19%	84.11%	84.48%	84.40%	84.48%
a6a	11220	21341	80.46%	84.17%	84.71%	84.72%	84.73%
a7a	16100	16461	83.01%	84.65%	84.81%	84.85%	84.89%
a8a	32561	16281	84.03%	84.92%	85.02%	84.98%	84.97%

- SVM type: C-SVC.
- Cost value (parameter C): 1.
- Kernel function: RBF.

1.2.1 Comparison between SVM and Deep learning methods

SVM has a great performance on MNIST dataset and achieves accuracy of 97.92%. The comparison results are in Table 4. We can see from the table that many works applying deep learning on MNIST have achieved high accuracy and I can also find some papers applying well-designed SVM on this dataset and accuracy of them can be more than 99% as well.

Table 4: Accuracy of different classifiers

Classifier	Test accuracy	Reference
SVM (in this work)	97.92%	
Convolutional net LeNet-4	98.90%	[3]
Convolutional net LeNet-5 [no distortions]	99.05%	[3]
Convolutional net LeNet-5 [high distortions]	99.15%	[3]
Convolutional net LeNet-5 [distortions]	99.20%	[3]
Convolutional net Boosted LeNet-4 [distortions]	99.30%	[3]
Trainable feature extractor + SVMs [no distortions]	99.17%	[4]
Trainable feature extractor + SVMs [elastic distortions]	99.44%	[4]
Trainable feature extractor + SVMs [affine distortions]	99.46%	[4]
large/deep conv. net, 1-20-40-60-80-100-120-120-10 [elastic distortions]	99.65%	[5]
committee of 7 conv. net, 1-20-P-40-P-150-10 [elastic distortions]	99.73%	[6]
committee of 35 conv. net, 1-20-P-40-P-150-10 [elastic distortions]	99.77%	[7]

1.2.2 Discussion of SVM on big datasets

Strengths

- Effective in high dimensional spaces.
- Still effective in cases where number of dimensions is greater than the number of samples.
- Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
- Versatile: different Kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels.

Weaknesses

- If the number of features is much greater than the number of samples, avoid over-fitting in choosing Kernel functions and regularization term is crucial.
- SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation.

2 Causal discovery algorithms

2.1 Problem Introduction

Happiness of people has always been a key problem of countries. Governments or some organizations usually study which factors influence people's happiness (national-level differences in life evaluations) and what are the correlations among them. Through these, corresponding steps can be taken to improve the life evaluations of countries.

Dystopia is an imaginary country that has the worlds least-happy people. The purpose in establishing Dystopia is to have a benchmark against which all countries can be favorably compared (no country performs more poorly than Dystopia) in terms of each of the key variables. The lowest scores observed for the key variables, therefore, characterize Dystopia. Since life would be very unpleasant in a country with the worlds lowest incomes, lowest life expectancy, lowest generosity, most corruption, least freedom, and least social support, it is referred to as Dystopia, in contrast to Utopia. Here we should study all these key factors as well as their relationships.

2.2 Data Collection

The dataset I collected is from the World Happiness Report, which has been released by SDSN and extracted by PromptCloud's custom web crawling solution.

The World Happiness Report is a landmark survey of the state of global happiness that ranks 156 countries by how happy their citizens perceive themselves to be. This years World Happiness Report focuses on happiness and the community: how happiness has evolved over the past dozen years, with a focus on the technologies, social norms, conflicts and government policies that have driven those changes.

2.3 Algorithms

Here I conduct four algorithms through **Tetrad**: BPC, FAS, FCI, and PC. The results are in the following graphs.

BPC adds two latent variables which are the causes of several factors and do not explore the relations among these factors. Thus, BPC works not very well. FAS can only get that some factors are related and is also not very well. PC can find some relations among factors, and FCI can find specific relations among factors, including all 4 edge types in Tetrad.

In these algorithms, FCI is the best and PC is the second-best. Ladder is a ranking of national-level happiness evaluation, which is determined by all other factors no matter directly or indirectly. From the experimental results, the relations between other factors can be complex and we can see from the results that ladder is closely influenced by positive affect, healthy life expectancy, and social support. Social support and healthy life expectancy are causes of log of GDP per capita. Corruption, positive affect, generosity have causal relationships with freedom.

References

[1] Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1–27:27, 2011.

[2] John C. Platt. Fast training of support vector machines using sequential minimal optimization. In Bernhard Schlkopf, Christopher J. C. Burges, and Alexander J. Smola, editors, Advances in Kernel Methods - Support Vector Learning, Cambridge, MA, 1998. MIT Press.

[3] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition." Proceedings of the IEEE, 86(11):2278-2324, November 1998.

[4] Fabien Lauer, Ching Y. Suen, Grard Bloch. A trainable feature extractor for handwritten digit recognition. Pattern Recognition, Elsevier, 2007, 40 (6), pp.1816-1824.

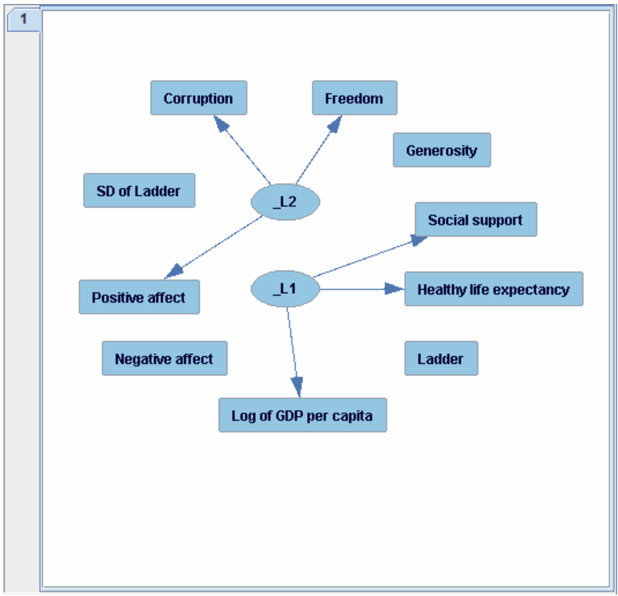

Figure 1: BPC

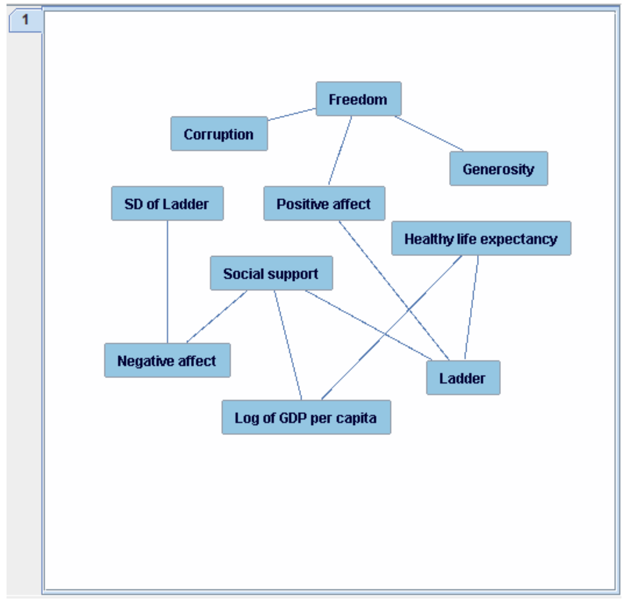

Figure 2: FAS

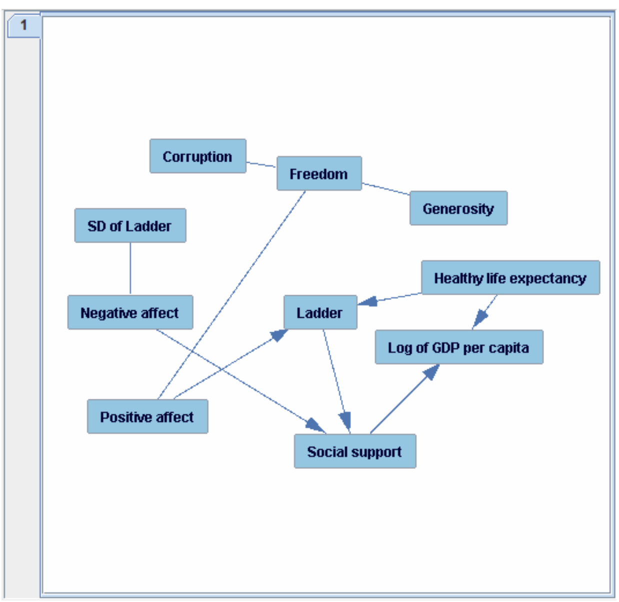

Figure 3: PC

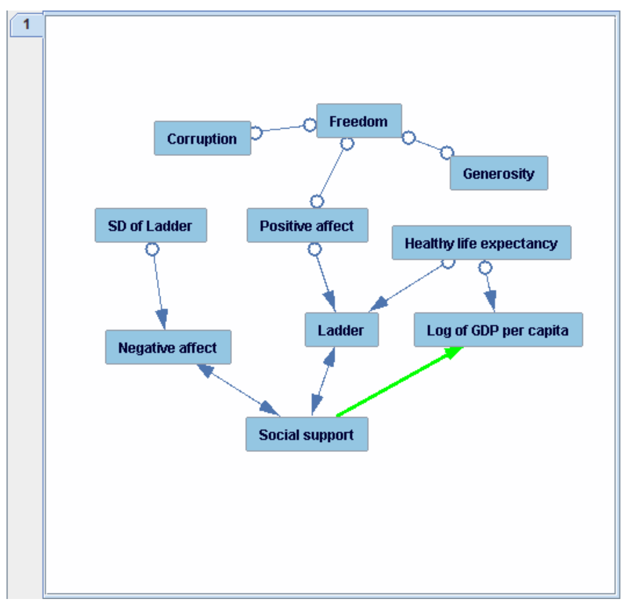

Figure 4: FCI

[5] Dan Claudiu Ciresan, Ueli Meier, Jonathan Masci, Luca Maria Gambardella, Jurgen Schmidhuber. Flexible, High Performance Convolutional Neural Networks for Image classification. International Joint Conference on Artificial Intelligence, 2011, 1237–1242.

[6] Dan Claudiu Ciresan, Ueli Meier, Luca Maria Gambardella, Jurgen Schmidhuber. Convolutional Neural Network Committees for Handwritten Character Classification. International Conference on Document Analysis and Recognition, 2011, 1135–1139.

[7] Dan C. Ciresan, Ueli Meier, Jurgen Schmidhuber. Multi-column deep neural networks for image classification. Multi-column deep neural networks for image classification, 2012, 3642–3649.