Slimming Down the Gene Pool: A Lean Approach to Leukemia Classification

Ewan Wallace

This study examines dimensionality reduction techniques to address the "curse of dimensionality" in genetic microarray datasets, which often have far more features than samples. We compare feature selection and extraction methods for reducing dimensionality while maintaining classification accuracy. A Support Vector Machine classifier achieved 100% accuracy on the test set using all features. We found that Recursive Feature Addition required only 3 features to maintain perfect accuracy, outperforming standard feature extraction methods like Principal Component Analysis. Our results suggest that for applications requiring model interpretability, such as cancer research, feature selection methods can match or exceed the performance of feature extraction while preserving feature meanings.

1 Introduction

Cancer diagnosis has been a huge area of research for many years, however despite continued breakthroughs it is still the second leading cause of death globally[1]. Insights from machine learning (ML) on microarray datasets are the source of many of these breakthroughs[2], learning patterns in general gene expressions that represent DNA sequences of biological samples. The trouble is that DNA is incredibly complex. Datasets often contain tens of thousands of gene expressions, however collection and labelling of sample data is expensive, often requiring hours of expert attention. This leaves us with the difficult situation of a sparse dataset, where the number of features is often significantly larger than the number of data points. This sparsity is known as the curse of dimensionality[3] and can lead to overfitting, high variance and a lack of interpretability of ML models[4,5].

Methods for reducing dimensionality fall into two main categories: feature selection and feature extraction. Feature selection picks only the most important features and discards the rest, reducing the original feature space into a subspace without transformation. Feature extraction creates entirely new features by transforming the original feature space into a distinct space with a different set of axes. Examples of these methods are recursive feature addition/elimination (RFA/E) and principle component analysis (PCA) respectively.

In simple terms, feature extraction creates features often with no physical meaning - difficult for interpretation[6]. Feature selection is superior in terms of readability and interpretability[7] but may in turn require more features. Reducing dimensionality can not only improves prediction performance, understandability, scalability and generalisation of a classifier, but it also reduces computational complexity and storage, providing faster and more cost effective models[8]. Specifically for microarrays, effective gene selection has been shown to improve early tumor detection and cancer discovery as it leads to more reliable cancer diagnosis[9].

In this work we will look at a leukemia dataset[10] from the Gene Expression Omnibus (GEO)[11] that exemplifies this problem, containing 22,283 features and only 64 data points. We will explore the effectiveness of different feature selection and extraction techniques and compare them based on accuracy, number of features and interpretability.

2 Datasets

2.1 CuMiDa

The dataset (Leukemia GSE9476[10]) examined in this work is one of the datasets from the *Curated Microarray Dataset* (CuMiDa)[12]. The CuMiDa dataset selects 78 microarray datasets from over 30,000 in the GEO[11] using a strict set of criteria (*e.g.* studies without chemotherapics and only studies performed on *Homo sapiens*). Once chosen, the datasets underwent extensive preprocessing including quality and background corrections and normalisation[12].

CuMiDa successfully collects high quality datasets spanning 13 different types of cancer, created solely for benchmarking and testing of ML approaches applied to cancer research. Universal benchmarks are incredibly important both within academia and industry. Having all models within a certain subject area evaluated on the same datasets allows for direct comparison between methods. This creates more transparency and objectivity within the scientific community and is fundamental for the advancement of bioinformatics[13]. Benchmarking is particularly important within cancer research as the 5 most commonly used datasets are all relatively old, with the most recent being published in 2002[14–18].

2.2 Leukemia GSE9476

Acute myeloid leukemia (AML) is one of the most common and deadly forms of hematopoietic malignancies[10]. The Leukemia GSE9476[10] dataset contains 64 samples - 26 AML patients and 38 healthy donors with normal hematopoietic cells. The healthy donors are further classified into bone marrow (10), bone marrow with CD34+ cells (8), peripheral bloods (10) and peripheral bloods with CD34+ (8).

There are 22,283 features for each sample with each feature in the dataset being a real number indicating how much a given gene is expressed.

3 Methodology

3.1 Data Cleaning

Due to the selection criteria implementing in collating CuMiDa[12], GSE9476 is a high quality dataset and requires no data cleaning.

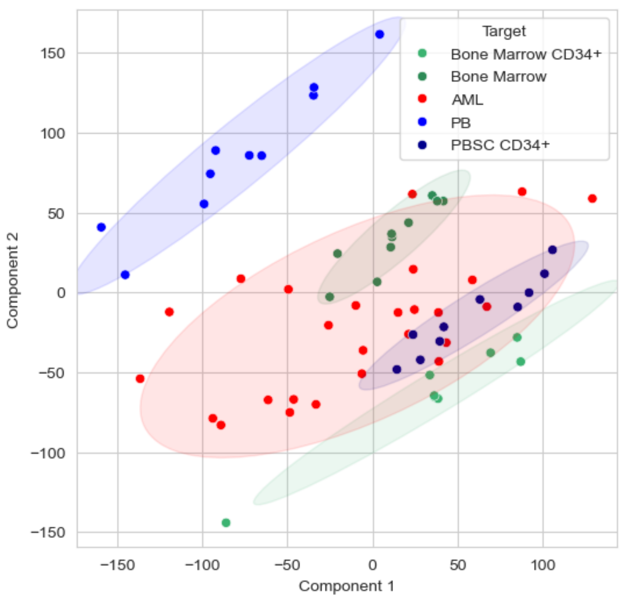

Fig. 1 2D principle component analysis of the dataset. Shaded ellipses indicates 95% confidence intervals for each class.

3.2 Data Exploration

Principle component analysis (PCA) of the dataset (Fig. 1) demonstrates that there is significant overlap between AML and other healthy donor classes in 2 dimensions. AML also has a noticeably larger variance, as indicated by the size of its confidence interval ellipse. Both of these indicate that it may be more challenging to classify than other classes. Interestingly there is significant separation between CD34+ classes and their respective normal classes, demonstrating that this is an important feature for classification. Equivalent plots were made for factor analysis (FA) and singular vector decomposition (SVD) however there was negligible difference to the PCA plot.

There is a substantial class population imbalance between AML and all healthy classes (Fig. 2). A binary classification was considered, however due to the high performance across all models on the harder multi-classification problem (Table 1), multi-classification was chosen. This also allows for easier comparison with literature evaluated on the same dataset.

3.3 Model Training and Selection

The dataset is split into training and testing sets using a stratified split and 80% training data (51:13) (Fig. 2). All models trained are evaluated on the training set using leave-one-out cross validation (LOOCV) and additionally evaluated on the test set. LOOCV achieves a low bias at the cost of high variance and computational expense. CuMiDa uses k-fold cross validation (k=3) however with an already small dataset, training on only 2/3 of the data at once is anticipated to harm performance. As the dataset has only 64 samples, LOOCV is not very expensive and is chosen here.

The models evaluated were chosen to match those used in the CuMiDa. Baseline models use sklearn's default parameters unless stated otherwise. Each model then underwent a hyperparameter grid search with the results given in 1.

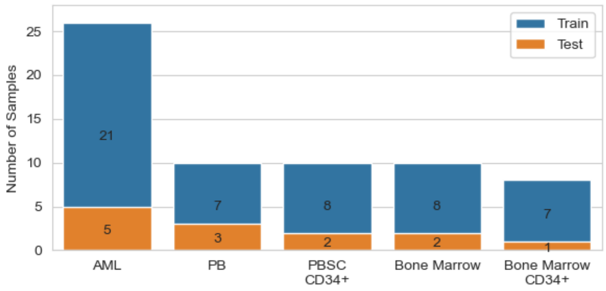

Fig. 2 Population of each class as well as their train/test split.

3.4 Support Vector Machines

SVM is a widely used supervised learning algorithm for microarray classification[19–22] due to it's robustness in high-dimensionality datasets. It seeks the best hyperplane to separate data points into classes, maximizing the margin. It transforms input data into a higher-dimensional space using the kernel trick for effective non-linear boundary separation.

3.5 Feature Extraction Methods

Factor Analysis: FA identifies underlying factors by modeling the observed variables as linear combinations of latent factors plus error terms. It aims to capture the common variance among observed variables and express it in terms of a smaller number of unobserved factors.

Principle Component Analysis: PCA works by finding the orthogonal axes (principal components) along which the data varies the most. It transforms the original variables into a new set of uncorrelated variables, capturing the maximum variance in the data. The first principal component corresponds to the direction of maximum variance, and subsequent components capture the remaining orthogonal variance.

Singular Value Decomposition: SVD decomposes a matrix into three constituent matrices, representing the input matrix as a product of singular vectors and singular values. It identifies the orthogonal directions in the input space that capture the most variation and expresses the data in terms of these orthogonal directions.

3.6 Feature Selection Methods

Recursive Feature Addition: RFA iteratively builds a feature set to optimize model performance. Starting with an empty set, RFA adds features one at a time. In each iteration, it evaluates all remaining features, selecting the one that, when combined with the current set, maximizes a chosen performance metric, in this case accuracy. This process continues until adding more features no longer significantly improves the model's performance.

Multiple Filter Multiple Wrapper[19]: MFMW is a filter-wrapper hybrid method that builds upon Single Filter Single Wrapper (SFSW) to give a more accurate and robust method. Multiple filters are used to select genes subsets that are then combined into a unified subset. A wrapper with multiple classifiers is then used, and unanimous voting resolves classification conflicts. The process iterates, selecting genes to minimize misclassifications and indecisive outcomes.

Minimum Redundancy–Maximum Relevance & Genetic Algorithm[20]: MRMR-GA is a hybrid method combining the MRMR filter and GA wrapper. MRMR is first used to remove noisy and redundant genes before GA selects highly discriminatory genes. MRMR iteratively balances maximum relevance (genes with the highest mutual information with the class label) and minimum redundancy (mutual information among selected genes) to select genes. In GA genes are encoded as binary values in chromosomes. An initial population is evaluated for fitness, typically classifier accuracy. Through genetic operations like crossover and mutation, the population evolves, improving gene subsets over generations.

Effective Range based Gene Selection[21]: ERGS is a non-iterative filter feature selection method that does not require a search strategy. It assigns higher weights to features that clearly distinguish between classes by evaluating the effective range of each feature. The effective range is defined statistically, considering the mean and standard deviation of each feature for different classes, and is scaled by the class probability to minimize the effect of high-probability classes. ERGS ensures that features with minimal overlap between classes are prioritized, thus enhancing classification accuracy.

3.7 Dimensionality Reduction Analysis

3 feature extraction and 4 feature selection methods are tested. The minimum number of features required to achieve perfect accuracy on the test set was determined for all methods (Table 2). A SVM classifier was used in all cases, however in hybrid models an ensemble of wrappers was used (MFMW: SVN and KNN, MRMR-GA: SVM and GA).

4 Results

4.1 Model Selection

The models evaluated (Table 1) were chosen to match those used in the CuMiDa database: support vector machine (SVM), decision tree (DT), random forest (RF), naive bayes (NB), multi-layer perceptron (MLP) with a single hidden layer of 10 neurons, k-nearest neighbours (KNN).

Model	CuMiDa[12]	Baseline	Tuned	Test
SVM	0.98	0.94	0.98	1.00
DT	0.89	0.73	0.80	0.85
RF	0.98	0.96	1.00	1.00
NB	0.89	0.88	0.88	0.85
MLP	0.94	0.86	0.90	0.77
KNN	0.89	0.88	0.88	0.85

Table 1 Accuracies of a range of classification algorithms. Test is the model with best performing hyperparameters evaluated on the test set.

After the hyperparameter sweep SVM and RF perform the best, giving both good LOOCV accuracy on the training set and perfect accuracy on the test set. There is no consensus in the literature on the best classification models for genetic expression[23], however SVMs and RFs are found to be top performers in many use cases[24–26]. This is unsurprising as both models are known for their accuracy and robustness in high-dimensionality datasets.

The SVM is chosen for further comparison for two reasons: its prevalence in the literature results and its test accuracy > train LOOCV accuracy, demonstrating that it may generalise better than the RF. A linear kernel and regularisation parameter (C) = 0.1 are used for all subsequent SVM models.

4.2 Dimensionality Reduction

The minimum number of features required to achieve 100% accuracy were calculated for RFA, PCA, FA and SVD. For comparison the accuracy of n randomly chosen features was also calculated. For each value of n, 100 sets of n randomly chosen features were used to train and evaluate a SVM model. The average accuracy is then plotted for each value of n in Fig. 3. SVD was also evaluated, however it's results were identical to those of PCA and so has been omitted from the Fig. 3.

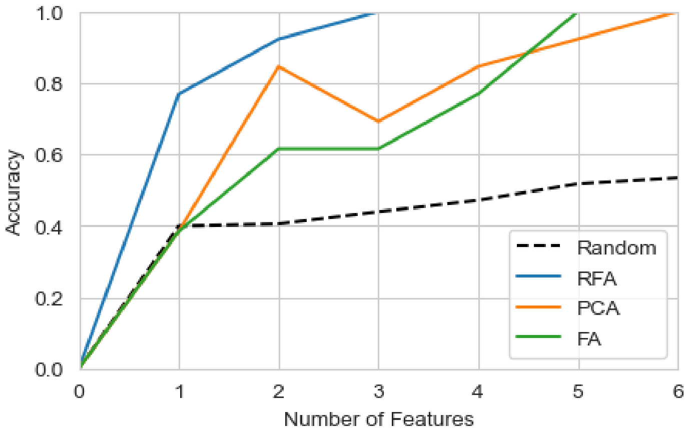

Fig. 3 Number of features required to achieve 100% accuracy for different dimensionality reduction methods. The values for SVD and PCA were identical so SVD is omitted.

RFA was by far the most computationally expensive dimensionality reduction method calculated as it scales linearly with the dataset feature space, requiring the training and evaluation of over 22,000 models for each value of n. However, it gives the best performance requiring only 3 features to achieve 100% accuracy on the multi-classification.

Interestingly, when n=1 all feature extraction methods perform no better than a random choice of gene, despite RFA giving an accuracy of almost double.

Ang *et al.*[27] present a thorough review of supervised, unsupervised and semi-supervised feature selection methods, 23 of which are evaluated on out dataset, GSE9476[10]. Of the 23 methods evaluated, 15 include the number of features required for a given performance (range:4-100, median:25) and 10 achieve perfect performance. The 4 best performing and most similar supervised methods are summarised in Table 2.

RFA requires the fewest features to achieve perfect performance on the test set, followed closely by MFMW - a hybrid of ensembled filters and ensembled classifiers. This seems to indicate that the top feature selection method out-perform all feature extraction

	Grouping Method	Features	Accuracy
Feature Extraction	FA	5	1.00
	PCA	6	1.00
	SVD	6	1.00
Feature Selection	RFA (filter)	3	1.00
	MFMW[19] (hybrid)	4	1.00
	MRMR-GA[20] (hybrid)	15	1.00
	ERGS[21] (filter)	80	1.00

Table 2 Performance of dimensionality reduction methods and the minimum number of featrues require to achieve 100% accuracy on GSE9476[10]. All feature selection methods used LOOCV. All classification were done using a SVM (or an ensemble including a SVM).

methods. This is true in this case, however the feature extraction methods tested here still require significantly fewer features than the median number (25) required in Ang *et al.*'s review[27].

Rank	PCA	SVD	FA	RFA
1	221864_at	221864_at	220997_s_at	205174_s_at
2	208734_x_at	208734_x_at	211494_s_at	201215_at
3	200965_s_at	201745_at	221942_s_at	1053_at
4	201745_at	200965_s_at	221910_at	
5	212827_at	212827_at	207639_at	

Table 3 Features ranked by importance for each model.

Feature importance can be a useful metric to understand how ML models are learning from such complex and high dimensionality data. The 5 most important features for each method are given in Table 3. Unsurprisingly given their identical performance, the most important features for PCA and SVD are almost identical with only a difference being in the 3^{rd} and 4^{th} positions. There is no overlap between these 5 features, FA's top features or RFA's top features, indicating that despite all achieving perfect accuracy, each method is learning a completely separate way of classifying samples.

5 Discussion

The curse of dimensionality presents a significant challenge in the context of high-dimensional datasets, such as those often encountered in gene expression studies. As the number of features increases, the volume of the feature space grows exponentially, leading to sparsity of data points. This sparsity complicates the model's ability to discern meaningful patterns and often results in overfitting, where the model captures noise rather than the underlying signal. Consequently, the model's performance deteriorates on unseen data, manifesting as poor generalization.

The tuned Support Vector Machine (SVM) model trained here exhibited better performance on the unseen test set compared to LOOCV, suggesting that overfitting may not be a significant issue. However, given the small test set size, this conclusion remains tentative. LOOCV was employed to mitigate high variance, a common problem in high-dimensional feature spaces with limited data points. Despite this, the model's sensitivity to minor fluctuations in the training data underscores the need for robust feature selection and dimensionality reduction techniques.

Creating and applying new feature selection methods can uncover novel insights into the importance and interplay of certain genes. This can lead to breakthroughs in understanding disease mechanisms and identifying potential therapeutic targets.

Although RFA performs best in this specific context, for larger datasets with more data points or features, its linear scaling may make it prohibitively expensive. The additional literature feature selection methods given in Table 2 successfully define several generic algorithms that can be applied to any high dimensional dataset.

While there is large variability in the number of features required for feature selection models, there is a large array of specialised selection methods created specifically for microarray data. This study indicates that the best selection model is likely to match or exceed the performance of standard feature extraction methods, while crucially maintaining interpretability.

Although all feature extraction methods used here assign an importance to each feature to increase explainability, the lack of consistency in the assignment of important features between feature extraction methods and a complete disconnect with the features deemed most important by RFA clearly show that feature extraction methods lose much of their interpretability when compared to feature selection methods.

In the healthcare sector, model interpretability is crucial. A high-performing model that cannot be explained will likely never be applied in practice. Clinicians need to understand the rationale behind model predictions to trust and act upon them. Therefore, balancing model performance with interpretability should be a priority. As such, we conclude that in cases where interpretability is important, and even in some where it isn't, feature selection is the clear choice over feature extraction.

It is common in the literature[19–21,27] reviewing dimensionality reduction techniques to use number of features as a metric for performance. The same metric has been used here to assess methods, however it is worth questioning the utility of this metric. Reducing down the feature set from 22,283 is crucial for human interpretability of a model, however does reducing the number of features from 20 to 10, or 10 to 5 necessarily improve interpretability? I propose instead the use of log(number of features) with a universally agreed cut-off (perhaps 1 = log(10)), below which all models are considered equivalent so as not to unnecessarily favour methods with the lowest number of features.

The dataset used in this study is relatively small, often resulting in perfect accuracy across various methods, complicating objective comparisons. This is why in many comparisons on dimensionality reduction methods a wide range of microarray datasets are used to test models[19–21,27]. This work too would benefit from additional comparisons across many datasets, such as those provided by CuMiDa.

Due to the prevalence of unlabelled data in this space, it is worth considering the potential of unsupervised and semi-supervised feature selection methods. Ang *et al.* concludes that semi-supervised is the best approach making use of low dimensional embedding from the unlabeled data and supervised algorithms to learn reasonably accurate classifiers from the labeled data. Additionally this allows for training on much larger datasets leading to more statistically significant results.

Finally, this study exclusively employed standard feature extraction methods, leaving room for comparison with gene-specific feature extraction methods. These tailored methods might yield more biologically relevant features, thereby enhancing model performance and interpretability.

6 Conclusion

Microarrays of genetic data often have a very high dimensionality leading to uninterruptible and sometime poorly performing ML models. In this case a tuned SVM trained on all 22,283 features achieves 100% accuracy on the test set. Feature selection methods are required to increase the explainability of the model without forfeiting any accuracy. For studies requiring explainability of models, such as those in cancer research and health more generally, feature selection methods perform sufficiently well as to leave standard feature extraction methods mostly obsolete.

Feature selection methods are often ranked on both their accuracy and number of features required. In this paper we propose that number of features metric should be replaced with log(number of features) with all methods achieving below a certain threshold being considered equal.

Code availability

The methodology laid out in this paper follows the steps taken in the attached Jupyter notebook and is also available on github (https://github.com/erwallace/leukemia-classification).

Data availability

The Leukemia GSE9476 dataset is available either from kaggle (https://www.kaggle.com/datasets/brunogrisci/leukemia-gene-expression-cumida) or directly from CuMiDa (https://sbcb.inf.ufrgs.br/cumida) by searching GSE9476. There you can also find evaluation details from the 6 baseline models that CuMiDa applies to the dataset.

References

1 F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre and A. Jemal, *CA: a cancer journal for clinicians*, 2018, **68**, 394–424.

2 Z. Tao, A. Shi, R. Li, Y. Wang, X. Wang and J. Zhao, *J buon*, 2017, **22**, 838–843.

3 R. Bellman and R. Kalaba, *Proceedings of the National Academy of Sciences*, 1959, **45**, 1288–1290.

4 W. Worzel, A. Almal and C. MacLean, *Genetic Programming Theory and Practice IV*, 2007, 29–40.

5 M. Mramor, G. Leban, J. Demšar and B. Zupan, Artificial Intelligence in Medicine: 10th Conference on Artificial Intelligence in Medicine, AIME 2005, Aberdeen, UK, July 23-27, 2005. Proceedings 10, 2005, pp. 514–523.

6 P. Krızek, *PhD thesis*, Ph. D. thesis, Czech Technical University in Prague, 2008.

7 S. Alelyani, J. Tang and H. Liu, *Data Clustering*, 2018, 29–60.

8 M. Gutkin, R. Shamir and G. Dror, *PloS one*, 2009, **4**, e6416.

9 A.-L. Boulesteix, C. Strobl, T. Augustin and M. Daumer, *Cancer informatics*, 2008, **6**, CIN–S408.

10 D. L. Stirewalt, S. Meshinchi, K. J. Kopecky, W. Fan, E. L. Pogosova-Agadjanyan, J. H. Engel, M. R. Cronk, K. S. Dorcy, A. R. McQuary, D. Hockenbery *et al.*, *Genes, Chromosomes and Cancer*, 2008, **47**, 8–20.

11 T. Barrett, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F. Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, M. Holko *et al.*, *Nucleic acids research*, 2012, **41**, D991–D995.

12 B. C. Feltes, E. B. Chandelier, B. I. Grisci and M. Dorn, *Journal of Computational Biology*, 2019, **26**, 376–386.

13 B. Peters, S. E. Brenner, E. Wang, D. Slonim and M. G. Kann, *Putting benchmarks in their rightful place: the heart of computational biology*, 2018.

14 T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri *et al.*, *science*, 1999, **286**, 531–537.

15 U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack and A. J. Levine, *Proceedings of the National Academy of Sciences*, 1999, **96**, 6745–6750.

16 D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie *et al.*, *Cancer cell*, 2002, **1**, 203–209.

17 J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson *et al.*, *Nature medicine*, 2001, **7**, 673–679.

18 A. A. Alizadeh, M. B. Eisen, R. E. Davis, C. Ma, I. S. Lossos, A. Rosenwald, J. C. Boldrick, H. Sabet, T. Tran, X. Yu *et al.*, *Nature*, 2000, **403**, 503–511.

19 Y. Leung and Y. Hung, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2008, **7**, 108–117.

20 A. El Akadi, A. Amine, A. El Ouardighi and D. Aboutajdine, *Knowledge and Information Systems*, 2011, **26**, 487–500.

21 B. Chandra and M. Gupta, *Journal of biomedical informatics*, 2011, **44**, 529–535.

22 B. Liao, Y. Jiang, W. Liang, W. Zhu, L. Cai and Z. Cao, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2014, **11**, 1146–1156.

23 D. B. Allison, X. Cui, G. P. Page and M. Sabripour, *Nature reviews genetics*, 2006, **7**, 55–65.

24 J. W. Lee, J. B. Lee, M. Park and S. H. Song, *Computational Statistics & Data Analysis*, 2005, **48**, 869–885.

25 M. Pirooznia, J. Y. Yang, M. Q. Yang and Y. Deng, *BMC genomics*, 2008, **9**, 1–13.

26 A. Statnikov, L. Wang and C. F. Aliferis, *BMC bioinformatics*, 2008, **9**, 1–10.

27 J. C. Ang, A. Mirzal, H. Haron and H. N. A. Hamed, *IEEE/ACM transactions on computational biology and bioinformatics*, 2015, **13**, 971–989.