Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

BACHELOR PROJECT ASSIGNMENT

Student: Vladimír K u n c

Study programme: Open Informatics

Specialisation: Computer and Information Science

Title of Bachelor Project: Increasing Weak Classifier Diversity in Ensemble Models by Feature Graphs

Guidelines:

1. Get familiar with the ensemble learning theory. Focus on the aspects of weak classifier decorrelation.
2. Assume that prior knowledge is available. It is a feature interaction network where vertices correspond to features and edges to their interactions. Study the recent Network-constrained forest method that employs the feature network to decorrelate and make more accurate the well-known random forest model.
3. Try to enhance the method ad 2. The enhancement can consist in a change of the type of weak classifers, ensemble structure or feature network treatment.
4. Test the proposed method on a real or an artificial bioinformatics datasets provided by your supervisor. The main criteria shall be classification accuracy and interpretability of the resulting model.

Bibliography/Sources:

[1] Dietterich, T.: Ensemble Methods in Machine Learning. Multiple Classifier Systems, LNCS Volume 1857, pp 1-15, 2000.
[2] Breiman, L.: Random Forests. Machine Learning, Volume 45, Issue 1, pp 5-32, October 2001.
[3] Li, N., Yu, Y., Zhou, Z.: Diversity regularized ensemble pruning, Machine Learning and Knowledge Discovery in Databases, LNCS Vol. 7523, pp 330-345, 2012.
[4] Kuncheva, L., Whitaker, Ch.: Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine Learning, Volume 51, Issue 2, pp 181-207, May 2003.
[5] Andel, M., Klema, J., Krejcik, Z.: Network-Constrained Forest for Regularized Omics Data Classification. In Proceedings of The IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 410-417, 2014.

Bachelor Project Supervisor: doc. Ing. Jiří Kléma, Ph.D.

Valid until: the end of the summer semester of academic year 2015/2016

L.S.

doc. Dr. Ing. Jan Kybic
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, January 26, 2015

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Vladimír K u n c

Studijní program: Otevřená informatika (bakalářský)

Obor: Informatika a počítačové vědy

Název tématu: Zvyšování rozmanitosti slabých klasifikátorů ve složených klasifikátorech pomocí příznakových grafů

Pokyny pro vypracování:

1. Seznamte se s teorií složených klasifikátorů a rolí jejich dekorelace v učení.
2. Uvažujte apriorní znalost ve formě sítě příznaků, kde vrcholy odpovídají příznakům a hrany jejich interakcím. Prostudujte existující Network-constrained forests metodu, která na základě příznakové sítě dekoreluje rozhodovací stromy a maximalizuje tím přesnost výsledného rozhodovacího lesa.
3. Pokuste se o rozšíření přístupu ad 2, rozšíření může spočívat ve změně typu slabých klasifikátorů, změně struktury složeného klasifikátoru či zobecnění samotného přístupu k příznakové síti.
4. Navrženou metodu testujte nad reálnými, popřípadě umělými bioinformatickými daty dodanými vedoucím práce. Hodnoťte z pohledu klasifikační přesnosti a interpretability.

Seznam odborné literatury:

[1] Dietterich, T.: Ensemble Methods in Machine Learning. Multiple Classifier Systems, LNCS Volume 1857, pp 1-15, 2000.
[2] Breiman, L.: Random Forests. Machine Learning, Volume 45, Issue 1, pp 5-32, October 2001.
[3] Li, N., Yu, Y., Zhou, Z.: Diversity regularized ensemble pruning, Machine Learning and Knowledge Discovery in Databases, LNCS Vol. 7523, pp 330-345, 2012.
[4] Kuncheva, L., Whitaker, Ch.: Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine Learning, Volume 51, Issue 2, pp 181-207, May 2003.
[5] Andel, M., Klema, J., Krejcik, Z.: Network-Constrained Forest for Regularized Omics Data Classification. In Proceedings of The IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 410-417, 2014.

Vedoucí bakalářské práce: doc. Ing. Jiří Kléma, Ph.D.

Platnost zadání: do konce letního semestru 2015/2016

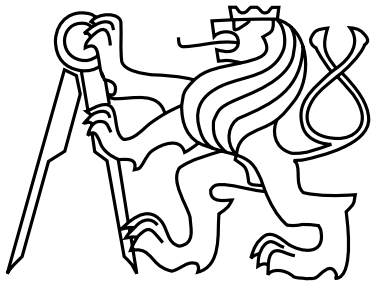
L.S.

doc. Dr. Ing. Jan Kybic
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 26. 1. 2015

Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics



Bachelor Project

Increasing weak classifier diversity in ensemble models by feature graphs

Vladimír Kunc

Supervisor: doc. Ing. Jiří Kléma, Ph.D.

Study Programme: Open Informatics

Field of Study: Computer and Information Science

May 22, 2015

Aknowledgements

I would like to express my sincere gratitude to my advisor doc. Ing. Jiří Kléma, Ph.D for the support of my project and for his patience with lengthy discussions.

Prohlášení autora práce

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 15. května 2015 .....

Podpis autora práce

Abstract

One of the common problems in machine learning from gene expression data is the scarcity of samples — these datasets usually have around tens of thousands of features but only several dozens of samples at best. Moreover, samples obtained using microarray technology are often very noisy. Therefore models built solely from measured data often suffer from overfitting. One of possible methods dealing with overfitting is to use prior knowledge for regularization. This work analyzes *network-constrained forest* (NCF) method proposed by Anděl and Kléma and proposes generalization of this method using other types of weak classifiers. The proposed method is analysed in terms of diversity and accuracy over several datasets. Moreover, this work empirically tests proposed convergence of NCF for increasing length of random walk used for feature sampling.

Keywords: ensemble, prior knowledge, diversity

Abstrakt

Jedním z běžných problémů strojového učení na datech genové exprese je nedostatek vzorků — tyto datasety mají obvykle několik desítek tisíc atributů ale v nejlepším případě jen několik desítek vzorků, navíc, vzorky získané pomocí technologie *microarray* obsahují velké množství šumu. Z těchto důvodu modely postavené výhradně z naměřených dat obvykle trpí přeučením. Jednou možnou metodou řešící problém přeučení je použití apriorní znalosti k regularizaci. Tato práce analyzuje metodu *network-constrained forest* (NCF) navrženou Andělem a Klémou a dále předkládá zobecnění této metody používající jiné typy slabých klasifikátorů. Navržená metode je analyzována z pohledu diverzity a přesnosti na několika datasetech. Navíc, tato práce empiricky testuje teoretickou konvergenci NCF pro zvyšující se délky náhodné procházky použíté pro vzorkování atributů.

Klíčová slova: sdružené klasifikátory, apriorní znalost, diverzita

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, the field of genomics was strongly influenced by progress in technology which made sequencing DNA much simpler and cheaper. Moreover, the amount of data led to creation of a new discipline - bioinformatics. The goal of this thesis is to show how ensemble classifiers with prior knowledge can be used for better predictions of the onset and progression of heterogenous multifactorial diseases such as *myelodysplastic syndrome*.

Ensemble methods train multiple classifiers and use these classifiers to create a compound classifier for a single task. These compound ensemble classifiers usually outperform each of the base classifiers, from which they are created, in most classification tasks [19, 39, 40, 99, 134, 195]. They are among state-of-art machine learning approaches. Ensemble methods represent many different approaches with different advantages and disadvantages. The key assumption of ensemble classifiers is that the underlying classifiers are diverse, i.e., that they make different errors, and thus they can together achieve higher predictive performance than could be obtained from any of the individual constituent classifiers [40, 158, 178]. The theoretical introduction to ensembles together with their taxonomy is in Chapter 3 Ensemble classifiers where there are described various ensemble approaches together with references to many examples of their use or to their latest extensions proposed in the literature. As every ensemble is compounded of few or many weak classifiers, Chapter 4 Weak classifiers describes several weak classifiers that are of particular interest to this work — *Decision trees*, *Logistic regression* and *Naïve Bayes*.

As was stated above, the diversity is crucial for ensembles of weak classifiers, thus the concept of diversity and explanation why the diversity is so important is presented in Chapter 5 Diversity. Moreover, this chapter also contains the description of various ensemble diversity measures that were developed over the years — several of these measures are used in the Chapter 7 Experiments.

The potential of ensemble methods with prior knowledge is demonstrated in Chapter 7 Experiments using gene expression profile data related to *myelodysplastic syndrome* (MDS). The chapter also contains thorough discussion of obtained experimental results and shows the relationship between diversity and ensembles with prior knowledge and also that the use of other types of weak classifiers such as *Logistic regression* or *Naïve Bayes* in network-constrained ensembles is beneficial as well.

Chapter 2

Related work

This work is an intersection of several various fields — e.g., biology, or genetics, however the main focus of the work is on the machine learning field, therefore the reader is assumed to know basic facts about genetics, DNA–protein relationships, etc., however for further information about genetics, the *Principle of Genetics* by Snustad and Simmons is recommended as comprehensive source about the field [161] (Czech translation is also available [160]).

Moreover, since this work is divided into several, dissimilar chapters, related work is always in the appropriate chapter. Being that said, there are still several works that are very significant to this work and they are worth mentioning here. The works of Rokach [154, 155], Kuncheva [95] and Dietterich [40] were very significant for the Chapter 3 Ensemble classifiers. Another work by Kuncheva [97] and Brown's work [27] were important source for the Chapter 5 Diversity. However, probably the most important is the work by Kléma and Anděl [9, 10] because this thesis is a continuation of their work and the proposed *Network-constrained random subspace* method is a direct extension of Anděl's and Kléma's *network-constrained forest*.

Chapter 3

Ensemble classifiers

Ensemble methods create a predictive model by integrating multiple models and several studies show that ensemble methods very often outperform single classifier method in terms of prediction performance [24, 49, 154, 155]. Ensemble methods and classifier are extensively studied over past two decades and represent state-of-art method for classification.

The idea of ensemble methods has been researched since Tukey combined two linear regression models in 1977 [155]. However, the main development of ensemble methods begun in the 1990 when Hansen and Salamon published an ensemble of artificial neural networks [60] and foundations for well known Adaboost ensemble method were laid in Schapire's work *The strength of weak learnability* [157]. The algorithm itself was published six year later by Freund and Schapire [49] and it produced a strong classifiers using combination of weak classifiers. In the same year, Breiman used Bagging predictors for improving accuracy of tree classifiers [25]. This method can be used generally for any type of classifiers but its tree version led to development of the popular random forest classifier in 2001 [26]. Many different and sophisticated ensemble methods were developed since then and a review of those methods can be found in [154] or [192].

3.1 Ensemble terminology

We used terminology presented in [155]. According to the work, an ensemble classifier is made up from several building blocks — a training set, an inducer, an ensemble generator and a combiner.

1. Training set — it is a labeled dataset used for training the ensemble classifier in the supervised learning. It is made up from samples described by an attribute-valued vectors. The dataset contains both labeled and unlabeled instances in the semi-supervised learning.

2. Inducer — it obtains a training set and uses it for forming a base classifier which represents the generalized relationship between the input attributes and the target attribute [155].

3. Ensemble generator — it creates diverse base classifiers for the ensemble

4. Combiner — it combines the decisions of various base classifiers and produces final decision (classification)

3.2 Ensemble taxonomy

Ensemble classifiers have many things in common and there were several attempts to categorize them. These attempts include [27, 40, 95, 141]. However, the most complete taxonomy was presented in [155] and [154]. This taxonomy was be used for purposes of this work and its brief description is be presented in this chapter, however, several new sections and examples were added (e.g., *rotation forest*, *confidence based method*) and some sections from the taxonomy were excluded.

The focus of the taxonomy is on several dimension in which a classifier might be categorized:

1. Combiner usage — it represents the relationship between base classifiers and the combiner.

2. Classifiers dependency — determines how base classifiers affect each other during training phases.

3. Diversity generator — if base classifiers are diverse, they may be more effective. Whole Chapter 5 Diversity will discuss diversity and its influence.

4. Ensemble size — it represents the number of base classifiers in the ensemble and also methods how undesirable classifiers are removed from the ensemble

5. Cross-inducer — determines whether the ensemble method was built for one type of inducer or it might be used for different inducers

Each of these dimensions of Rokach's taxonomy is described further in following sections.

3.3 Combiner usage

There are two main categories in this dimensions — named *weighting* and *meta-learning* in [155] but named as *trainable* and *nontrainable* ensembles in [6, 95] or as *fixed combiners* and *trained combiners* in [44]. The trainability specifies whether the ensemble is trained to make final decision from its base classifiers' decisions in or whether some fixed rule is used for combining those outputs such as *majority voting* or *weighted voting*. Moreover, the trainable ensembles might be trained after all base classifiers are trained or it might be trained during the training of individual base classifier — example of the latter is AdaBoost [49] and its derivations [50]. Another important feature in this dimension is whether the ensemble generator is combiner specific or it is combiner independent and the combination method is provided as an input to the framework [155].

In following sections, there is a brief description of several frequently used combiners, more complete list of combiners is available in [6, 24, 44, 95, 154, 155]. Also, an empirical comparison of some of the combiners on biological data is in [6].

3.4 Fixed combiners

3.4.1 Majority voting

This method is probably one of the most used combiners, for example, it is used in [5, 24, 26, 164]. Ensemble using majority voting selects the class that receives the highest number of votes among the base classifiers where each vote has the same weight:

$$\text{class}(\mathbf{x}) = \underset{c_j \in \text{dom}(y)}{\arg\max} \left(\sum_k g(y_k(\mathbf{x}), c_i) \right)$$

where $g(y, c)$ is an indicator function defined as:

$$g(y, c) = \begin{cases} 1 & y = c \\ 0 & y \neq c \end{cases}$$

and $y_k(\mathbf{x})$ is the classification of the k'th base classifier.

Further analysis of majority voting is in [6, 27, 95, 99]. Its empirical performance is compared with *weighted voting* and other combining methods in [24], double layer voting is used in [186].

3.4.2 Weighted voting

This method can be both in Fixed combiner and Trained combiners because the category depends on the origin of the weights. These weights can be set apriori or can be trained as well — both during and after training separate base classifiers. The most known example is AdaBoost which learns weights when training base classifiers [49]. The mathematical expression is very similar to majority voting:

$$\text{class}(\mathbf{x}) = \underset{c_j \in \text{dom}(y)}{\arg\max} \left(\sum_k \alpha_k g(y_k(\mathbf{x}), c_i) \right)$$

where $g(y, c)$,$y_k(\mathbf{x})$ are same as in majority voting and α_k is the weight of k'th classifier [95]. These weights might be set apriori — for example based on known characteristics of classifiers. In case of binary classification, the voting can be further simplified to

$$(\mathbf{x}) = \text{sign} \left(\sum_k \alpha_k h_k(\mathbf{x}) \right)$$

where $h_k(\mathbf{x})$ returns the predicted class by k'th classifer for $\mathbf{x}$:

$$h_k(\mathbf{x}) = \begin{cases} 1 & \text{classifed as class 1} \\ -1 & \text{classified as class -1} \end{cases}$$

as presented for example in [49]. More detailed description and analysis are available in [95, 125].

3.4.3 Performance weighting

Performance weighting is a special case of weighted voting where weights are assigned on the basis of base classifiers' performance on the validation set [155]. One possible performance weighting mentioned in [95, 133] is:

$$\alpha_i = \frac{1 - E_i}{\sum_k (1 - E_k)}$$

where $1 - E_k$ is classifer k's validation set accuracy (or training-set accuracy if validation set is not used). Other possible weightings [95]:

$$\alpha_i = E_i^{E_i}(1 - E_i)^{(1-E_i)}$$

$$\alpha_i = \frac{1}{E_i}$$

These weights could be normalized so they sum up to one as in the first weighting example but it is not necessary as division by a constant does not affect the ordering.

3.4.4 Distribution summation

Very simple combiner that requires probabilistic output from classifiers instead of just class [154, 155]:

$$\text{class}(\mathbf{x}) = \underset{c_j \in \text{dom}(y)}{\arg\max} \sum_k \widehat{P_{M_k}}(y = c_j|\mathbf{x})$$

3.4.5 Naïve Bayes

The Naïve Bayes combiner (also called *independence model*, *simple Bayes* or *idiot's Bayes*) is very similar to *Distribution notation* mentioned above and it assumes that the classifiers are mutually conditionally independent given a class label [95, 155]:

$$\text{class}(\mathbf{x}) = \underset{\substack{c_j \in \text{dom}(y) \\ \widehat{P}(y=c_j)>0}}{\arg\max} \sum_k \widehat{P}(y = c_j) \cdot \prod_{k=1} \frac{\widehat{P_{M_k}}(y = c_j|\mathbf{x})}{\widehat{P}(y = c_j)}$$

Further description is in [95].

3.4.6 Confidence based method

Sometimes the underlying base classifiers can supply not only class but also the confidence of the class, which might be better for using in the ensemble than just the class output. These methods are very similar to methods above and can be generalized as [95]:

$$\mu_j(\mathbf{x}) = \mathcal{F}(d_{1,j}(\mathbf{x}), \ldots, d_{L,j}(\mathbf{x}))$$

Several often used choices of $\mathcal{F}$:

Simple mean(average)

$$\mu_j(\mathbf{x}) = \frac{1}{L}\sum_{i=1}^{L} d_{i,j}(\mathbf{x})$$

Minimum

$$\mu_j(\mathbf{x}) = \min_i d_{i,j}(\mathbf{x})$$

Maximum

$$\mu_j(\mathbf{x}) = \max_i d_{i,j}(\mathbf{x})$$

Median

$$\mu_j(\mathbf{x}) = \text{median}_i d_{i,j}(\mathbf{x})$$

Product

$$\mu_j(\mathbf{x}) = \prod_{i=1}^{L} d_{i,j}(\mathbf{x})$$

Generalized mean The generalized mean may represent *arithmetic mean*,*harmonic mean*, *geometric mean*, and many others means with correct use of α [95]:

$$\mu_j(\mathbf{x}, \alpha) = \left(\frac{1}{L}\sum_{i=1}^{L} d_{i,j}(\mathbf{x})^{\alpha}\right)^{\frac{1}{\alpha}}$$

3.5 Trained combiners

3.5.1 Weighted Average

This method is similar to confidence fixed methods but it adds weights. There are two main types of weighted average combiners [47, 95]:

- *L weights*. This model uses one weight per classifier (similarly to *Weighted voting*)

$$\mu_j(\mathbf{x}) = \sum_{i=1}^{L} \alpha_i d_{i,j}(\mathbf{x})$$

- $L \times c$ *weights.* This model uses one weight per classifier and class:

$$\mu_j(\mathbf{x}) = \sum_{i=1}^{L} \alpha_{ij} d_{i,j}(\mathbf{x})$$

  For this kind of model, linear regression is usually used for deriving the weights α_{ij} [95].

3.5.2 Stacking

In stacking, another classifier is used for creating decision from multiple decisions made by base classifiers. It can be viewed as creation of a new dataset, where with same number of instances as in original dataset, but instead of original attributes, its attributes are decisions of base classifiers — one attribute per base classifier [154, 155]. Stacking can be used both for classifiers induced by one inducer or for classifier induced by different inducers. Special stacking method are even suitable for online learning [74].

3.6 Classifier dependency

Another dimension from Rokach's taxonomy [154, 155] is whether the base classifiers are dependent or independent. If they are dependent, output from one classifier affects the creation of the next classifier.

3.7 Dependent methods

These method are also called *successive* because training of one classifier influence the training of the following classifier. It is possible to distinguish two classes of dependent methods [144]. First is *Model-guided instance selection* where the classifiers from previous iterations are used for manipulating the training set for the following iteration while the other is called *Incremental batch learning* method — it uses classification from previous iteration as prior knowledge to the algorithm in following iteration [144, 154, 155]. Because the output of the previous classifiers is not captured by weighting of the training samples and the classifiers but used directly as a feature of an instance, this method does not use any combiner but directly uses the output of the last classifier.

3.7.1 Boosting

One of the most popular method of model-guided instance selection is called *boosting* also known as *Adaptive Resampling and Combining* or briefly *arcing.* Boosting approach evolved from online learning algorithm called *Hedge(β)* [95] and the most known boosting classifier is Adabost, which reweights instances in the training set in each iteration based on the classification from previous iteration — the weights of misclassified instances are increased and the weights of correctly classified instances are decreased [49, 153]. Consequently, it

tries to create classifiers that complement each other and then combines their decisions using weighted voting.

The key assumption of model-guided instance selection boosting is that all used weak inducers can work with weighted instances. If it is not the case, a possible workaround is generating unweighted dataset using a resampling technique where instances are chosen with probability according to their weights [154, 155].

It was showed that boosting can greatly improve classifier's accuracy [19, 40, 49, 91, 134, 145], however it can sometimes lead to a loss of generalization, mostly when the algorithm over-fits [145] which is expected because noisy examples tend to be misclassified which will result in an increase of the weight of the instance [91].

3.8 Independent methods

In the independent methods, the classifiers are trained independently of each other, the training dataset is usually partitioned into several subsets which are then used for training classifiers. These subset may be either mutually exclusive (disjoint) or overlapping [155].

3.8.1 Bagging

One of the most known independent methods is called *bagging*, which is just abbreviation for *bootstrap aggregating*. It was first used by Breiman in 1996 in [25]. It is a method for generating multiple versions of a predictor and using them to get an aggregated predictor [25, 95, 135, 153–155, 192]. Each classifier is trained using new training set that is created by taking instance from original training with replacement and thus some instances from original dataset may be more than once in the new dataset or not included at all [155]. Bagging ensembles usually use majority voting for combining decisions of base classifiers which are independently trained on their new datasets.

The bagging approach work best with unstable classifiers — that is with classifiers for which a small change in training instances can result in large changes in the classifier [19, 25, 95, 145, 154, 155]. Bagging is usually best suited for middle sized datasets because a single classifier can be very accurate given large dataset and if the dataset is tiny, then the gains achieved by using bagging cannot recompense for the decrease in accuracy of individual base classifiers [91].

Several empirical studies show that bagging almost always increased the accuracy of base classifier and in no cases it led to worse performance than the performance of base classifier on the original dataset [19, 40, 91, 134, 145, 195]. Also several new algorithms based on bagging were produced — for example *SubBag* in [135], *FuzzyBagging* in [127] or bagging based algorithms in [178].

3.8.2 Wagging

Less known but very useful method is called wagging. In this method, each base classifier is trained on the entire training set, however each instance is stochastically assigned a weight [153–155]. Traditionally, wagging is described as a variant of bagging [19, 154, 155],

but more accurate is to view bagging as a variant of wagging because bagging is a wagging with allocation of weight from the Poisson distribution where each instance is represented in the sample a discrete number of times [153–155, 185]. Continuous equivalent of bagging can be modeled by using the exponential distribution instead of discrete Poisson distribution. Individual random instance weights from the Poisson distribution can be easily generated using following well known formula:

$$T = -\frac{\log(U_i)}{\lambda} \tag{3.1}$$

where U_i is a uniformly distributed random number on $(0, 1)$. Since pseudo-random generators sampling uniform distribution over $(0, 1)$ are common, it is very easy to implement continuous equivalent of bagging.

3.8.3 Random subspace

The random subspace method (also called attribute bagging) creates subspaces from the input feature spaces. Each subspace is created by randomly picking features from the entire input feature space and each base classifier is trained using one of these subspaces [65, 154, 155, 202]. This method is very useful especially when the dimension of the feature space is very high and most other classification methods suffers from the *curse of dimensionality* [65]. This method can be applied to many different inducer, e.g., decision trees [26, 65], nearest neighbor classifiers [65, 154, 155] or linear discriminators [154, 155].

3.8.4 Random forest

The random forest ensemble method was first proposed by Breiman in 2001 in [26] and became very popular since then. The random forest ensemble is very similar to random subspace method but it is tree-specific. Instead of subspacing features space for each base classifier, the entire feature space is sampled at each node of each decision tree in the ensemble. Hence the tree does not choose the best split among all features but only among features from subspace of the entire feature space [26, 154, 155]. These base classifiers' decision are then combined using majority vote combiner. Another advantage of random forest is that they can be also used for regression [15, 26, 40] where the outputs of base classifiers are combined by averaging combiner instead of majority vote.

The random forest is also popular because it provides both accurate classification and insight regarding the discriminative ability of individual features as it provides a feature importance measure and also out-of-bag estimates of generalization error [12, 26, 61, 62, 154]. Moreover, random forests can also be used for feature selection as a pre-processing for other classifiers [14, 52, 61, 62]. Unlike other feature selection methods, random forests can implicitly deal with missing values [62]. Furthermore, it it possible to interpret nonlinear relationships between process variables by using of random forests whereas the current most popular tool for modeling complex nonlinear system — neural networks — do not provide explicit insight into the relationships between process and target variables. [15]. However, forest variable importance measures are less accurate when some input features

are correlated [14], more detailed study of tree ensemble variable importance measures is available in [14], where there are also compared importance measures from *boosting trees* and *conditional inference forest*.

3.8.5 Rotation forest

Rotation forest is another independent method specific for trees and it is an important enhancement of random forest first introduced in 2006 in [152]. While this method is very similar to random forest, rotation forests outperform random forests on many datasets [7, 152]. Rotation forest aims at building accurate and diverse base classifiers. Similarly to bagging, bootstrap samples are used for training the individual base classifiers but the rotation forest also transforms the data into a new feature space [96, 152]. For each base classifier, the input data are split to several disjoint subsets on which a transformation method such as Principal component analysis (PCA) is consequently applied. The classifier was first proposed with PCA as the transformation method [152] but its behavior was later analysed with other transformation method such as non-parametric discriminant analysis (NDA), Random projections (RP), Sparse random projections [96] or with Independent component analysis (ICA) [46]. However, Kuncheva and Rodriguez stated in [96] that the PCA gave the highest results and it also preserves the discriminatory features, second best transformation was found to be the NDA.

The rotation forest offers good trade-off between the accuracy and the diversity, which are considered to be contradictory [27, 46], because it provides good accuracy through gathering results and it uses the rotation process to create diversity [46].

However, the rotation forest has one big disadvantage in comparisons with its biggest rival random forest because while it often provides better accuracy than random forest, it cannot be used for assessment of feature importance due to the transformation of feature space [152]. The absence of implicit feature importance measure makes rotation forest less suitable for application where explicit insight into the relationships between process and target variables is needed.

Another disadvantage comes from the use of PCA as the transformation method because rotation forests, unlike the random forests, cannot implicitly deal with missing data because the PCA itself fails to process missing elements and moreover, it is very sensitive to outliers [37]. Even though several robust PCA have been proposed and tested over the years [37], their use in rotation forests was not well analysed. Moreover, the use of PCA in rotation forests significantly increases computational difficulty in comparison with the random forest.

3.8.6 Cross-validated committees

This method is very similar to bagging but uses different strategy: it creates k classifiers by dividing the training set into k-equal-sized sets and trains them on all but the ith set [154, 155]. However, this method is less favoured than the similar bagging method. Some interesting recent applications are *Graph-Based Cross-Validated Committees* from 2012

in [115] and *Crossboost* from 2006 in [173] that combines Adaboost and cross-validated committees with neural networks.

3.9 Ensemble diversity

The ensemble provides higher accuracy, only if the ensemble members disagree about some inputs [27, 97, 154, 155, 176]. The concept of diversity is further described in Chapter 5 Diversity. This section is focused on the taxonomy of methods for diversity creation based mostly on [27, 154, 155].

3.9.1 Manipulating the inducer

Diversity can be gained by manipulating the inducer that creates the base classifiers. The inducer can be manipulated in several possible ways.

3.9.1.1 Manipulating parameters

Most of the base inducers used for ensemble can by controlled by a set parameters. In the context of artificial neural networks, number of layers and number of nodes in a layer can be such parameters or more generally, the whole topology for the network can be the manipulated parameter. Variation of number of nodes was used in [77, 137], variation of parameters and learning algorithm with selection of diverse members using clustering algorithm was used in [110] and varying number of hidden neurons, types of activation functions and learning rate using *genetic algorithms* was used in [126]. Another common example, where the diversity is obtained using different parameters, is an ensemble of decision trees [155] where, for example, the trees might be prunned early.

3.9.1.2 Starting point in hypothesis space

It is possible to achieve higher diversity by setting different point in hypothesis space where the individual base learners starts the search. The change of origin influences where in space the learner converges to [27]. In context of neural networks, it is possible to manipulate the back-propagation inducer by assigning different initial weights to the network [155].

3.9.1.3 Traversal of hypothesis space

The way of searching the hypothesis space greatly influences the final ensemble [27, 155]. According to Rokach, there are two methods of manipulating the space traversal — *Random-based strategy* and *Collective performance strategy* [155]

Random-based strategy
Randomness can lead to higher diversity, one of the most common examples is a random forest, in which each classifiers is selecting not the best feature in each node but selecting only the best feature from a subset of nodes [26]. Different forest randomization strategy

was used in [39],where, for each tree, "the 20 best candidate splits are computed, and then one of these is chosen uniformly at random". This randomization was comparable to *bagging* but might be more accurate in setting with low noise [39].

Collective Performance based strategy

This type of strategy creates the ensemble as a whole while trying to increase its accuracy by various means. The base classifiers might cooperate with each other in order to specialize, i.e., be diverse from others [155]. There are two main approaches — *penalty method* and *evolutionary methods* [27].

Penalty methods

When using penalty, a *penalty term* is added to the *error function* of an ensemble to encourage diversity among base classifiers [27, 154, 155]. Several penalty methods were proposed and analysed in the literature — e.g., *Negative Correlation Learning* [27, 155] or *Root Quartic Negative Correlation Learning* [27]

Evolutionary methods

Using this family of methods, the ensemble is evolved from initial population of classifiers [27]. One of the examples is [126], where an evolutionary algorithm was used for adding diverse members to the ensemble. Three-level evolutionary algorithm was used in [31]. The multi-objective evolutionary algorithm was used in [30] for finding a good trade-off between diversity and accuracy, another example of use of multi-objective evolutionary algorithms is available in [122]. Another possible approach was used in [71, 199], where they created ensemble by selecting diverse classification rules obtained by genetic programming [71].

3.9.2 Manipulating the training samples

One possible method for gaining diversity is to train the base classifiers on a variation of a subset of the original dataset [155]. This method is especially used of unstable classifiers such as neural networks and decision trees [155] Most known examples are bagging, boosting or wagging.

3.9.2.1 Resampling

The dataset is resampled into datasets on which the base classifiers are trained, the most common resampling scheme is *random sampling* as used in bagging or *proportional* that distributes samples based on the class distribution in the original datasets in such a way that the class distribution in each subset is approximating the distribution in the original dataset [155].

3.9.2.2 Partitioning

This method is useful when the number of training instances is very high and the training could become a bottleneck [154] Partitioning divides the training set into disjoint subsets and can both improve speed and diversity when training on massive datasets [154]. Moreover, it has has been empirically shown that the performance of this approach is equivalent to the bagging and other frequently used ensemble methods [154].

3.9.2.3 Reweighting

This method is similar to resampling, in certain cases both method can be equivalent [153–155, 185]. Instead of resampling, this method manipulates the weight of instances in the training set, most known examples are AdaBoost (boosting) and wagging [155].

3.9.2.4 Creation

Another possible method is to dynamically create new datasets on which the members of the ensemble are trained. One of the most known example is *DECORATE* algorithm presented in 2003, it is a dependent method in which members are added iteratively and they are trained on artificial datasets combining original instances with fabricated instances [154, 155].

Similar approach was used in [3], where neural network ensemble was trained on artificial dataset with fabricated instances to maximize diversity among neural networks.

3.9.3 Manipulating the target attribute representation

According to Rokach [155], methods using this approach usually work with several classifiers with different and simpler representations of target attributes instead of using a single complicated classifier. There are two main types: *Concept Aggregation*, in which the manipulation is based on an aggregation of original target's values, and *Function Decomposition*, which is based on more complicated functions than *Concept Aggregation* [154, 155]

3.9.4 Partitioning the search space

When using this approach, every ensemble member explores a different part of the search space, i.e., the original space is divided into several sub-spaces and each of these subspaces is considered independently from the rest of sub-spaces. Subspaces can overlap or could disjoint, the total model is then a union of simpler models [154]. Rokach in [154] distinguishes two main types of partitioning — *Divide and conquer* and *Feature subset-based methods*.

3.9.4.1 Divide and conquer

This approach divides the instance space into several subspaces and for each of the subspaces, a model is created. Such model is called an expert and decision of experts is then

combined to create final decision [154, 155]. The subspaces might divided apriori or using a partitioning algorithm such as *K-Means* or *Decision tree*. Further details about this approach are available in [155] in section *Horizontal partitioning* and in [154] in section *Divide and conquer*.

3.9.4.2 Feature subset-based methods

Using this method, also called *Vertical partitioning* in [155], each ensemble member is created by manipulating the original feature set, i.e., each ensemble member is trained using a different projection of the training set [154]. It can be very useful for datasets with high dimensionality [155] as it can even alleviate the curse of dimensionality [80], furthermore it reduces the correlation among the classifiers and thus it can improve performance of the whole ensemble [154], moreover the reduced size of the dataset leads to faster induction of classifiers [154]. Most of strategies for creating feature subset-based ensembles can be divided into three main categories — *random-based*,*reduct-based* and *collective performance based* strategy [155].

Random-based strategy
A simple and often very efficient strategy in which the subsets are created using a random selection or a random projection [155]. Random subsets of the feature space are used for a creation of a forest of decision trees in [65]. The most known random-based strategy is the *Random Subspace method* (RSM) which creates a random subspace of the whole feature subspace. Currently, it is one of the most popular method for creating ensembles [51], especially in the image recognition [56, 63, 80, 170, 171], moreover, several extensions were proposed in the literature, e.g., *weighted RSM with automatic dimensionality reduction* in [98], *Directed RSM* in [63] or *RSM with Canonical Correlation Analysis* in [208]. Moreover, several novel applications of the idea of RSM were proposed, for example *RSM for Co-training* in [182], RSM for creation of artificial training data in [2] or *RSM for Gene ranking* in [28].

Reduct-based strategy
The smallest feature subset with the same predictive power as the original feature set is called a reduct [155]. The ensembles created using reducts are limited in size because reducts are limited to the number of features [155]. Reducts are often used for a feature selection for training a single classifier (e.g., [13, 118], however, they can also be used for ensembles, e.g., *reduct based K-Means* in [76] or *reduct-based ensemble of UCS* in [38]. Moreover, several different classifiers can be used for construction of the reduct as in [124]. In comparison with bagging and the random subspace method, reduct based ensembles have more opportunities to get good generalization [38].

Collective Performance based strategy
The idea is the same as in the *collective performance based strategy* in Section 3.9.1 — firstly, features are sampled into subsets, usually randomly, then an iterative scheme is

introduced to refine this selection in order to improve the accuracy of the ensemble. The iterative scheme can be based on a hill-climbing search [155] or it can be based on a genetic search [132, 155, 175]

3.9.5 Multi-Inducers

Each type of inducers contains bias that results in preferring certain generalizations over others [154, 155], therefore, it is possible to obtain higher diversity using different types of inducers. The goal of this method is to produce synergistic effects that would lead to higher performance "that neither atomic approach by itself would be able to achieve" [155].

Chapter 4

Weak Classifiers

Various classifiers were proposed in the literature over the decades, hence it is greatly out of the scope of this work to categorize and thoroughly describe even just one particular class of classifiers (e.g., *neural networks*). This chapter will therefore describe only classifiers that are of particular interest to this work.

4.1 Decision tree

Decision tree (DT) classifiers are one of the most simple and yet most successful techniques in machine learning. They are even more useful because they are easily interpretable for humans — even if they are not from the machine learning community. The decision tree iteratively splits the input space in each node, the target classification is represented by a leaf. However, there are many methods how to create the decision tree — they vary from simple, greedy ones to more complex methods. There were several earliest algorithms such as AID, THAID, MAID, ELISEE and others — their review is available in [151]. One of latter algorithms was ID3 proposed by Quinlan in 1986 [146], it uses greedy approach and splits the subsets using the attribute with maximum information gain. He later extended the method into new C4.5 algorithm [88], which, unlike ID3, handles both continuous and discrete attributes, it also handles missing data, attributes with differing costs and prunes the tree after its creation. Another often used algorithm is CART, which uses a generalization of the binomial variance called the Gini index for finding suitable splits [116]. Unline THAID, both CART and C4.5 first grow a large tree and then prune it to a smaller size [116]. However, many different algorithms were proposed over the years, e.g., *Optimized Very Fast Decision Tree* (OVFDT) [191], *fuzzy decision trees* [34, 106, 121, 166], *fuzzy SLIQ decision trees* [33], and moreover, decision trees are used in other complex methods, e.g., decision tree growing with genetic programming [78, 85] or other evolutionary algorithms [17, 58]. Moreover, decision trees may be combined with other algorithms, for example combination of SVM and decision tree [93, 183].

Furthermore, decision trees are one of the most favourite types of weak classifiers, their use in ensembles is well researched — more details in [1, 2, 14, 39, 83, 108]. Comparison of some decision tree algorithms are available in [128, 151, 166] and reviews of various

decision tree algorithms are provided in [90, 116], survey of evolutionary algorithms for decision trees is available in [18].

4.2 Logistic regression

Logistic regression (LR) is a well known method of statistical learning but its description is not provided here as it is out of the scope of this work, however the description is available almost in any statistical learning or econometrics textbook, for example in *Introductory Econometrics: A Modern Approach* by Wooldridge [188].

This method is not used only in econometrics but also in bioinformatics as it can achieve comparable performances to decision trees or neural networks [84],e.g., predicting In-Hospital-Death [59], hypertension prediction [181], or classifying heart disease patients [84]. The logistic regression was also used for a gene–gene interaction analysis for genome-wide studies using a CUDA accelerated LR [102]

Many extensions of logistic regression or algorithms based on LR were proposed in literature — *Class Imbalance Oriented Logistic Regression* (CILR) [43], or *Choquet integral logistic regression* [112], moreover, the logistic regression may also be combined with other classifiers, for example combination of logistic regression and SVM for influenza host classification [113], or hybrid classifier using logistic regression with evolutionary RBF neural network [57].

4.3 Naïve Bayes Classifier

This classifiers is called *Naïve* because its crucial assumption is unrealistic and greatly simplifying — the Naïve Bayes assumes that all features all conditionally independent given class. However, despite this assumption, the classifier is often unexpectedly effective in practice [150]. Even though the independence assumption does not hold in nature in most cases, the Naïve Bayes classifiers often shows great performance on various biomedical taks — e.g., detection of abnormal gait patterns in Parkinson Disease [117], mature miRNA identification [54], classification of proteomics data [111], or classification of gene expression data [32, 48]. Moreover, the Naïve Bayes classifier is often used in ensembles [109, 129] In last decades, many extensions to the basic classifiers were proposed, for example *Extended Naïve Bayes* [86], *Fuzzy Naïve Bayes* [130, 169], *Repeat Based Naïve Bayes Classifier* (RBNBC) [147] for biological sequences, *iterative Naïve Bayes learning with missing data* [103], *Link-Based Naïve Bayes Classifier* (LNBC) [22], *Randomly Selected Naive Bayes* (RSNB) for feature extraction [79], or novel method for Naïve Bayes with continuous variables (NBC4D) [196].

Chapter 5

Diversity

5.1 Ensembles and diversity

The ensemble of classifiers can be only more accurate when the base classifiers differ and disagree about some inputs [154] , if it was not the case, the ensemble would be the same as a single base classifier. The ultimate goal is to have an ensemble with many base learners whose errors are independent of each other because then the ensemble would approach, in a classification context, the *Bayes error*, which is the minimal error that is achieved given the *optimal decision boundary*. If we had classes a and b with *posterior probabilities* $P(a|\mathbf{x})$ and $P(b|\mathbf{x})$ respectively and a classifier i whose outputs are estimates of posterior probabilities $\widehat{P}_i(c|\mathbf{x})$, $c \in \{a, b\}$ so that

$$\widehat{P}_i(c|\mathbf{x}) = P(c|\mathbf{x}) + \mu_i(c|\mathbf{x}) \tag{5.1}$$

where $\mu_i(c|\mathbf{x})$ is the estimation error of base classifier i, then, assuming that the estimation errors on different classes are independent and identically distributed random variables with zero mean and variance $\sigma^2_{\mu_i}$, the *expected added error* of base classifier i satisfies

$$E_{\text{i,add}} = \frac{\sigma^2_{\mu_i}}{P'(a|\mathbf{x}) - P'(b|\mathbf{x})} \tag{5.2}$$

where $P'(a|\mathbf{x})$ and $P'(b|\mathbf{x})$ are the derivatives of true posterior probabilities of classes a and b [27]. Moreover, if the decision is made by an ensemble of classifier instead of only one classifier with combiner averaging estimates of posterior probabilities, it can be shown [27, 97] that

$$E^{\text{ave}}_{\text{add}} = E_{\text{add}} \left(\frac{1 + \delta(L-1)}{L} \right) \tag{5.3}$$

where E_{add} is the added error of base classifiers (assuming all have the same error) and δ is the correlation coefficient computed as the sum of averaged pairwise correlations between $P_i(c_k|\mathbf{x})$ and $P_j(c_k|\mathbf{x})$, $i, j = 1, \ldots, L$ calculated for every class c_k weighted by the prior probabilities $\widehat{P}(c_k)$ [97]:

$$\delta = \sum_{c_k} \widehat{P}(c_k) \cdot \left(\frac{2\delta_{i,j}}{L(L-1)} \right) \tag{5.4}$$

where $\delta_{i,j}$ is the correlation between $P_i(c_k|\mathbf{x})$ and $P_j(c_k|\mathbf{x})$.

5.2 Diversity measures

Over the years, there were many diversity measures proposed in the literature (e.g., [53, 136, 159, 201]) however good review and comparison was made in [27, 97]. The diversity can be measured for both classification and regression ensembles, however, since our research is focused on the classification context, the diversity theory for continuous-valued outputs is out of the scope of this work.

There are two main approaches of measuring the diversity, *pairwise* and *non-pairwise* [27, 97]. The *non-pairwise* measures mostly compare the output of base classifiers with the averaged output of the whole ensemble or are based on the idea of entropy whereas the *pairwise* measures calculates the average of a particular measure of all possible pairings of ensemble members [27].

5.2.1 Used notation

Let $\mathbf{S} = \{\mathbf{s}_1, \ldots, \mathbf{s}_N\}$ be a labeled dataset, $\mathbf{s}_j \in \mathbb{R}^N$, $D = \{D_1, \ldots, D_L\}$ be an ensemble of classifiers and $y_{j,i} := 1$ if D_i classifies $\mathbf{s}_j$ correctly and $y_{j,i} := 0$ otherwise.

Also, for two classifier D_i and D_k, let N^{ab} be the number of samples $\mathbf{s}_j \in \mathbf{S}$ for which $y_{j,i} = a$ and $y_{j,k} = b$ [97] as shows Table 5.1. Also $N = N^{00} + N^{01} + N^{10} + N^{11}$, however, these relationships are sometimes stated in probabilities, not absolute numbers, as in [27]. For the estimates of such probabilities, we can divide N^{ab} by the number of samples N, i.e., $p_{ab} = \frac{N^{ab}}{N}$ and then $p_{00} + p_{01} + p_{10} + p_{11} = 1$.

	D_k correct	D_k wrong
D_i correct	N^{11}	N^{10}
D_i wrong	N^{01}	N^{00}

Table 5.1: A table of relationships between a pair of classifiers taken from [97].

5.2.2 Non-pairwise measures

5.2.2.1 The entropy measure E

There are several entropy based measures in the literature [27, 97], one possible measure is from [97]. It is based on the concept that the highest diversity among classifiers for a particular sample s_i,$i = 1, \ldots, L$, from the dataset is when $\lfloor \frac{L}{2} \rfloor$ votes of the ensemble produce same value (0 or 1) and the other $L - \lfloor \frac{L}{2} \rfloor$ produce the alternative value [97].

$$E = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{\left(L - \lceil \frac{L}{2} \rceil\right)} \min \left\{ l(\mathbf{s}_j), L - l(\mathbf{s}_j) \right\}. \tag{5.5}$$

where $l(\mathbf{s}_j)$ is the number of classifiers from the ensemble that correctly classifies the $\mathbf{s}_j$:

$$l(\mathbf{s}_j) = \sum_{i=1}^{L} y_{j,i} \tag{5.6}$$

where $y_{j,i} = 1$ if classifier D_i classifies correctly $\mathbf{s}_j$ and 0 otherwise.

5.2.2.2 Kohavi-Wolpert measure

This measure is based on a decomposition formula for the error rate of classifier derived by Kohavi and Wolpert, which gives the variability of the predicted class label y for $\mathbf{x}$ across training sets for a specific classifier model [89, 97] .

$$\mathrm{var}_x = \frac{1}{2}\left(1 - \sum_{i=1}^{c} P(y = \omega_i|\mathbf{x})^2\right) \tag{5.7}$$

Based on this formula, the Kohavi-Wolpert measure was derived in [97]:

$$KW = \frac{1}{NL^2}\sum_{j=1}^{N} l(\mathbf{s}_j)(L - l(\mathbf{s}_j)) \tag{5.8}$$

Moreover, the Kohavi-Wolpert measure is closely linked with the pairwise averaged disagreement measure $\mathrm{Dis}_{\mathrm{ave}}$ defined in following section:

$$KW = \frac{L-1}{2L}\mathrm{Dis}_{\mathrm{ave}} \tag{5.9}$$

This equivalence was proved in [97].

5.2.2.3 Measurement of interrater agreement κ

It was first introduced in 1960 by Cohen [16],however since then many modification and extensions were made, especially for the use in biostatistics. For the purpose of this work, by κ we will mean the pairwise and non-pairwise measures described in [97], more information about κ statistic is available in [16, 92, 193, 194]

κ is another measure closely related to the Kohavi-Wolpert measure and the disagreement measure, it was developed as a measure of interrater reliability κ, i.e., to measure the level of agreement while correcting for chance [97]:

$$\kappa = 1 - \frac{\frac{1}{L}\sum_{j=1}^{N} l(\mathbf{s}_j)(L - l(\mathbf{s}_j))}{N(L-1)\bar{p}(1-\bar{p})} \tag{5.10}$$

where $\bar{p}$ is the average individual classification accuracy:

$$\bar{p} = \frac{1}{NL}\sum_{j=1}^{N}\sum_{i=1}^{L} y_{j,i} \tag{5.11}$$

The κ is related to KW and Dis_{ave} [97]:

$$\kappa = 1 - \frac{L}{(L-1)\bar{p}(1-\bar{p})} KW = 1 - \frac{1}{2\bar{p}(1-\bar{p})} \text{Dis}_{\text{ave}} \tag{5.12}$$

5.2.2.4 Coincident failure diversity

Coincident failure diversity is a modification of another diversity measure called *Generalized diversity*. It is designed in a such way so it reaches minimum value of 0 when all classifiers are simultaneously either correct or wrong and it reaches its maximum of 1 when all misclassifications are unique, i.e., at most one classifier fails on any randomly chosen sample [97, 136].

$$CFD = \begin{cases} 0 & p_0 = 1 \\ \frac{1}{1-p_0} \sum_{i=1}^{L} \frac{L-i}{L-1} p_i & p_0 < 1 \end{cases} \tag{5.13}$$

where p_i is the probability that i of L classifiers are incorrect on randomly drawn object $\mathbf{x} \in \mathbb{R}^n$.

5.2.2.5 Other diversity measures

Other pairwise diversity measures were also proposed in the literature - *generalized diversity* [97, 136], *measure of difficulty* θ described in [97] or *classification Ambiguity* [27] and others.

5.2.3 Pairwise measures

This type of measures is calculated for every possible pair of classifiers from the ensemble and then averaged, e.g.,

$$Q_{\text{ave}} = \frac{2}{L(L-1)} \sum_{i=1}^{L-1} \sum_{k=i+1}^{L} Q_{i,k} \tag{5.14}$$

where $Q_{i,k}$ is the Q statistic defined in following section 5.2.3.1 but the same approach is used for all pairwise measures.

5.2.3.1 Q statistic

The statistics was used by Yule in 1900 in [201] and the Q statistic for a pair of classifiers D_i and D_k is [97, 201]:

$$Q_{i,k} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}} \tag{5.15}$$

The Q takes values between -1 and 1, and for statistically independent classifiers D_i and D_k the expected value is 0, i.e., $E[Q_{i,k}] = 0$. If both classifiers tend to classify the same samples correctly, they will have positive values of $Q_{i,k}$ whereas if classifiers tend to fail on different samples, the $Q_{i,k}$ will be negative.

5.2.3.2 Correlation coefficient ρ

This measure is related to the Q statistics, for any two classifiers D_i and D_k, $Q_{i,k}$ and $\rho_{i,k}$ have the same sign and moreover, $|\rho_{i,k}| \leq |Q_{i,k}|$ [97].

$$\rho_{i,k} = \frac{N^{11}N^{00} - N^{01}N^{10}}{\sqrt{(N^{11} + N^{10})(N^{01} + N^{00})(N^{11} + N^{01})(N^{10} + N^{00})}} \tag{5.16}$$

5.2.3.3 κ statistic

The pairwise κ statistics from [97]:

$$\kappa_{i,k} = \frac{2(N^{11}N^{00} - N^{01}N^{10})}{(N^{11} + N^{10})(N^{01} + N^{00}) + (N^{11} + N^{01})(N^{10} + N^{00})} \tag{5.17}$$

however, this statistic is not equivalent to the interrater agreement κ defined in Section 5.2.2.3 because it can be shown that κ is not equal to the average pairwise κ_{ave} [97].

Dietterich defines multi-class κ statistic in [39] and uses it for drawing κ-error diagrams for graphical visualization of diversity. The multi-class κ statistic is defined as follows. Let C be an $L \times L$ matrix such that $C_{r,s}$, $r, s = 1, \ldots, L$ is the number of samples assigned to class r by the classifier i and into class s by the classifier k, then

$$\kappa_{i,k} = \frac{\theta_1 - \theta_2}{1 - \theta_2} \tag{5.18}$$

where

$$\theta_1 = \frac{\sum_{r=1}^{L} C_{rr}}{N} \tag{5.19}$$

and

$$\theta_2 = \sum_{r=1}^{L} \left(\sum_{s=1}^{L} \frac{C_{rs}}{N} \sum_{s=1}^{L} \frac{C_{sr}}{N} \right) \tag{5.20}$$

This κ statistic equals 0 when the agreement of the two classifiers D_i and D_k equals the one expected by chance and it equals 1 when D_i and D_j agree on every sample.

5.2.3.4 The double-fault measure

The double-fault measure is defined as the percentage of cases that have been misclassified by both classifiers [53, 97] and it was used for selecting diverse neural network classifiers in [53].

$$DF_{i,k} = \frac{N^{00}}{N^{00} + N^{01} + N^{10} + N^{11}} = \frac{N^{00}}{N} \tag{5.21}$$

5.2.3.5 The disagreement measure

This measure is defined as the percentage "of test instances for which the base and complementary classifiers make different predictions but for which one of them is correct" [159]. The same measure, but in the form of *agreement measure* was used in [65] for measuring the tree agreement. Using notation from [97], the disagreement measure is defined in a following way [65, 97, 159]:

$$\mathrm{Dis}_{i,k} = \frac{N^{01} + N^{10}}{N^{00} + N^{01} + N^{10} + N^{11}} = \frac{N^{01} + N^{10}}{N} \tag{5.22}$$

Chapter 6

Ensembles using prior knowledge for omics task

During the last two decades, microarray technology has become an important part of modern genomics and the technology has stimulated new research line in bioinformatics and in machine learning. DNA microarrays allows parallel analysis of thousands of genes and their research, moreover they can be useful for disease diagnosis, especially in connection with bioinformatics and machine learning, for the diagnosis of heterogenous multifactorial diseases. One possible task is to distinguish patients with certain disease from healthy patients using their gene expression profile. Microarray technology can potentially allow to classify many diseases by simply obtaining one sample, however, to be able to do this, we need reliable classifiers with a high degree of accuracy and a low computational complexity [119].

However, building such classifier is a very difficult task for many microarray datasets because the sample size n is typically much smaller than the number of measured features p [9], $n \ll p$, also called the *curse of dimensionality* [81], which often leads to overfitting [10]. This problem can be addressed using several possible approaches. The most straightforward one is to use a robust classifier with good generalization such as *support vector machines* (SVMs) or other SVM based methods, which are capable of dealing with large dimensionality [10, 11, 21, 73, 123, 163, 170]. Several types of ensembles also improve generalization — such as *bagging* [155]. Ensembles might be well used for prediction functional proteins [172, 189], more information about classifiers for functional genomic is available in survey in [172]. Even though the SVM is not typical classifier for ensembles, it can be efficiently used in ensembles as well, e.g., bagged ensembles of SVM for gene expression data analysis [177]. Another common approach is *regularization*, which restrains the space of all hypotheses to improve generalization [9, 64]. One of the common regularizing methods is *feature selection* that tries to select the most discriminating features and then use these features for building a classifier in feature space with lower dimension [41, 42, 100, 107, 114, 119, 123, 139, 168, 197, 198]. Another regularizing approach is to use *prior knowledge* [4, 9, 10, 104, 148, 149, 167, 184], — e.g., biological network constrained regularization of linear models [104], inductive logic programming using gene ontology [156, 174],

statistical learning [75, 131], learning based on *Markov random field* [105, 203, 204]. Also, several works attempt to predict protein-protein interaction and analyse PPI networks using random walks [29, 36], other use decision trees for prediction genetic interaction by integrating genomic and proteomic information [187].

Furthermore, these approaches to the *curse of dimensionality* are often combined together. One possible combination is using prior knowledge together with feature selection — for example using prior knowledge for feature selection and then building a single SVM classifier [55], feature selection with Markov random field and MCMC algorithm [165] — thorough review of incorporating prior knowledge into feature selection in omics domain is available in [142]. Also it is possible to use SVM-based methods for feature selection, for example *P-SVM* feature selection [123], *Least-Square SVM* (LSSVM) with swarm optimization [168], *Recursive Feature Elimination SVM*(RFE-SVM) and *Greedy Correlation-Icorporated SVM* (GCI-SVM) feature selection [162] or SVM with *Iterative Reduced Forward Selection* (SVM-IRFS) [205]. Several researchers modified SVMs to incorporate prior knowledge. First among them (to the extent of our knowledge), Zhu introduced *Network-based SVM* in 2009 and compared it with other SVM methods(e.g., STD-SVM, L1-SVM) in [207], then various network based SVMs were introduced — e.g., *Network kernel SVM* [190] or *Network-Induced Classification Kernel SVM* (NICK-SVM) [101], which utilises the protein network topology and relations between the different features.

However, another approach is crucial for the purpose of this work — utilising prior knowledge in ensembles of weak classifiers. Zhou et al. used prior knowledge in the form of modules defined as miRNAs which regulate the same context, then detected modules with distinguishing abilities and each of them was used for building a weak classifier separately and then created an ensemble using these weak classifiers and voting combiner [206]. Another ensemble — *Module-Guided Random Forest* — was introduced in [35] and it iteratively builds random forests using weighted sampling of features taken from modules of correlated genes [9, 35].

Finally, there is the *network constrained forest*(NCF), on which is this work based. This algorithm was first proposed in [10] and was produced as a part of wider research concerning the use of machine learning in omics field. This research includes many topics — e.g., integration of mRNA and miRNA profiles [8, 87],cross-genom analysis[66], gene expression classification [68–70, 94] — and its many results are integrated in online tools *XGENE* and *miXGENE* which is further described in [67] and [66] respectively.

The NCF forest is further analysed in [9] and it allows to integrate prior knowledge from three different sources — a network of gene interactions,a network of miRNA and gene interaction and also known causal genes for certain diseases. Because of the importance of NCF to this work, the algorithm will be described further in following section.

6.1 Network-Constrained Forest

This algorithm combines two approaches for solving the $n \ll p$ problem common in the omics field, it utilises prior knowledge for creation of an ensemble of decision trees. While it achieves similar performance as SVM classifiers, it has a great advantage unlike the

black-box SVM — forest of decision tree also provides insight into the problem because it is possible to use them for feature importance ranking based on prior knowledge, moreover, the decision trees in the forest might be utilised for interaction extraction with prior knowledge, more details are available in Section 3.8.4 Random forest and in the context of NCF in [9]. Unlike Random Forest, the NCF biases "the feature sampling process towards the genes and loci in general, which have been previously reported as candidates for causing the phenomenon being studied (...) and consequently the omics features which directly or indirectly interact with those candidate genes" [9]. This sampling process is driven by random walk on the biological interaction network integrating both mRNA and miRNA prior knowledge and the process starts from the candidates causal genes called *seeds*. When candidates causal genes are unknown, seeds are randomly sampled from the entire set and the probability of a gene being sampled as a seed is proportional to its out degree in the network. Further implementation details and pseudo-code are available in [9, 10].

The crucial assumption behind the NCF is that gene that are close in the biological feature network are also correlated in their expression, therefore it is suitable to create weak classifiers grouping these features because it leads to decorrelating the individual weak classifiers and therefore to the creation of better diversity in the ensemble. The diversity in ensemble is very important for its performance, further details are available in Chapter 5 Diversity and in [97]. The biological background behind this method is discussed in [9] and the conclusion presented is that the weak decision "trees may vaguely correspond to the individual disease factors and their network-local manifestations" [9]. The individual trees are constructed using the features in the network neighbourhood of a particular seed gene that was chosen for the tree. The neighbourhood is represented by distribution function using which the feature set is sampled. This distribution is defined as a random walk of length k from the seed gene — it is more dense when closer to the seed gene and also it is not possible to reach genes that are further in the network than k. Therefore, the NCF is parametrized by the walk length k whose optimal value may be different for different tasks as it strongly influences the feature sampling [9]. A heuristic based on *incidence of underfitted trees* for setting the parameter k was proposed in [10]. The influence of k on the accuracy and diversity of weak learners and the overall accuracy of the ensemble is further analyzed in Chapter 7 Experiments. The NCF proposed in [10] was implemented in Python 2 as modification of Random Forest from machine-learning library Scikit-learn [138].

6.2 Proposed method NCRS

In this section, we propose a generalization of the NCF algorithm called *network constrained random subspaces*(NCRS) or *network constrained attribute bagging* (NCAB) which applies the idea of biased sampling of the feature set to the general ensemble *random subspace* method (viz Section 3.8.3 Random subspace). The idea of NCF is not strictly related to ensembles of decision trees and it is easily extensible to ensembles of other weak learners. Even though decision trees as weak classifiers of forest have many advantages as, for

example, direct interpretability and possible use of such forest for feature selection, other classifiers such as *logistic regression* or *naïve Bayes* might be used as well. Implementation details are described in Chapter 7 Experiments as well as the experimental results. Moreover, the integration of NCF to general ensemble method allows simple modification of the algorithm — e.g., modification of the feature sampling process without the need to modify other functions as well. The relationship between RF, RS, NCF, and NCRS is depicted in Figure 6.1 — the RF and the NCF both sample the feature space in each node of each tree, however, the RS and the NCRS both sample the feature space only for each weak learner. The sampling in the NCF and the NCRS is network-constrained, i.e., its sampling procedure generates samples using random walks over the interaction network, while the sampling procedure in the RF and the RS is random.

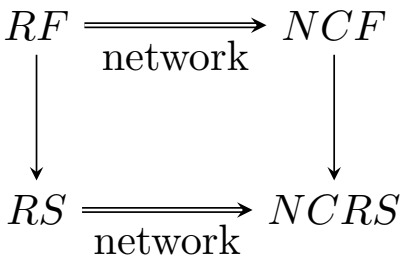


Figure 6.1: The relationship between NCF and NCRS is the same as the relationship between RF and RS — the RF samples features in each node (tree specific) while the RS samples features only in each weak learner.

Chapter 7

Experiments

7.1 Myelodysplastic syndrome

Data related to *myelodysplastic syndrome* (MDS) were used for most of the experiments. It is the same data that was used in the original experiment with NCF [9, 10]. A short description is provided in following section, the full description is available in [9].

7.1.1 Used data

As in [9, 10], MDS datasets were used for experiments with the generalized NCF called NCSR. The data were provided by a collaborative laboratory at the Institute of Hematology and Blood Transfusion in Prague and "informed consent was obtained from all the subjects whose samples were used for expression profiling, and the study was approved by Scientific Board and Ethics Committee of the Institue of Hematology and Blood Transfusion in accordance with the ethical standards of the Declaration of Helsinky" [9]. The data were obtained for analysis of lenalidomide treatment of patients with myelodysplastic syndrome.

The data consist of two datasets — mRNA with 16,666 attributes measuring the gene expression level and miRNA with 1,146 attributes measuring the expression level of particular miRNAs [9]. The samples were obtained from bone marrow (BM) CD34+ progenitor cells and from peripheral blood (PB) CD14+ monocytes and were obtained either before the treatment (BT) or during the treatment (DT). Moreover, the data can be further categorized by the partial deletion of the chromosome 5 (5q or non-5q). Using these categories, the data consisting of 75 samples were divided into 10 related datasets. More details about the biological background, preprocessing and profiling tools are available in [9]. The number of samples in datasets is shown in Table 7.1, and it is clear that with 17812 attributes, we are indeed dealing with the $n \ll p$ problem.

7.1.2 Prior knowledge

Again, for the coherence of experiments with [9, 10], same prior knowledge in form of gene networks was used. Gene networks and candidates causal genes were obtained from author of [9, 10], however, the are publicly available — in vitro validated miRNA-mRNA

Dataset code	number of samples
BMBT DT5q	16
BMH ABT5q	21
BMH ABTnon-5q	16
BMH ADT5q	15
BMnon-5q 5qBT	17
PBBT DT5q	22
PBH ABT5q	19
PBH ABTnon-5q	14
PBH ADT5q	23
PBnon-5q 5qBT	13

Table 7.1: Number of samples in individual MDS datasets

interactions are from TarBase 6.0 [179], in silico predicted interactions are from miRWalk database [45], experimentally validated protein-protein interactions are from Human Protein Reference Database [143], predicted protein-protein interactions are from [23] and MDS causal genes are from [200], according to [9].

7.1.3 Implementation

The NCSR ensemble classifier was implemented in Python 3 as a modification of both the original NCF [9, 10] and general *Bagging classifier* from machine learning library Scikit-learn [138] version 0.16.1 (older versions do not contain the Bagging classifier). The implementation consist mostly of small changes of NCF because of porting from Python 2 to Python 3 and then using the NCF code in the modified Bagging classifier. Other used libraries included Pandas [120], Scipy [82], Matplotlib [72], Cython [20], IPython [140] and NumPy [180].

7.1.4 Purpose of experiments

Experiments with the MDS datasets had several objectives. First of all, the goal was to replicate the results from [9] with the generalized ensemble NCRS. The second objective was to analyze the impact of different values of the parameter k defining the length of a random walk on the accuracy of both the whole ensemble and also of the individual weak classifiers. Moreover, [9, 10] implies that the diversity of weak classifiers should be strongly influenced by the parameter k and in most cases, a longer walk should lead to smaller diversity among the weak classifiers in the ensemble as they become less specialized. The parameter k was analysed for similar values as used in [9, 10] but also for more extreme values — e.g., for a random walk of length 100.

Another objective is to experimentally validate the convergence of NCRS (NCF) as $k \to \infty$. The NCF does not converge to *Random forest*, rather it converges to the stationary distribution of random walk [9] — i.e., to $\pi^{\infty}(v) = \frac{\deg(v)}{|\mathcal{I}|}$, where $\mathcal{I}$ is the the set of edges in the biological network, viz [9] for details. However, the NCF converges to

the stationary distribution only if there are no miRNA interactions present because such interactions are handled in a special way — when encountering the miRNA node in the walk, the walk always ends there, details are again available in [9]. The convergence was not experimentally validated in [9].

And last objective of experiments with MDS data was to determine whether the use of different weak classifiers could useful as well. For this reason, not only Decision trees were used as in [9, 10] but also ensembles with Logistic Regression and Naïve Bayes as weak classifiers. Each of these objectives are not present in [9, 10] and they represent unique experiments.

7.1.5 Experimental Protocol

As mentioned in Section 7.1.3 Implementation, Python 3 with Scikit-learn [138] machine learning library was used for experiments. 10 times repeated Stratified m-fold cross-validation was used for MDS experiments, where $m := \min\{10, c\}$, where c is the number of samples in the smallest class. Exact number of folds for each task is in Table 7.2.

Dataset code	number of folds
BMBT DT5q	5
BMH ABT5q	10
BMH ABTnon-5q	6
BMH ADT5q	5
BMnon-5q 5qBT	6
PBBT DT5q	9
PBH ABT5q	9
PBH ABTnon-5q	4
PBH ADT5q	10
PBnon-5q 5qBT	4

Table 7.2: Number of fold used in cross-validation

All ensembles were built from 1000 weak classifiers using the Random Subspace method, each weak classifier had access to 100 features. The number of weak classifiers was strongly limited by computational costs of both learning period and calculating pairwise diversity measures.

Matthew's correlation coefficient (MCC) was chosen as a measure of quality of classification because it provides a balanced quality measure with respect to classes with different sizes. It returns values from the interval $[-1, 1]$, where $+1$ represents a perfect classification, 0 a random classification and -1 indicates total disagreement between predicted classes and annotated classes. The MCC was calculated for predictions for the whole dataset, not for individual folds, and then averaged over repetitions — in contrast to [9], where median was used instead of averaging. The random walk length k was set to $k \in \{1, 2 \ldots, 14, 15\}$ for most experiments, different sets were used only for several experiments, in which it is explicitly noted which set of k was used.

7.1.6 Results

The results can be divided into several parts — comparison of the NCRS with the unbiased Random Subspace method, the analysis of diversity, and the analysis of convergence of NCRS.

7.1.6.1 NCRS and unbiased Random Subspace method

In the original study [10], the NCF was compared to the Random Subspace forest of Decision trees, however, our generalization NCRS allows the use of different weak classifiers in the ensemble. For this part of experiment, we have use NCRS with Decision trees (CART), Logistic Regression and Naïve Bayes Classifiers. In most tasks, the NCRS was better in terms of MCC for some values of k than the unbiased RS with the same type of weak classifiers. For each datasets, there are three possible results — NCRS was better for some vales of the parameter k (*win*, NCRS had exactly the same performance as RS for some values of k and worse for the rest (*tie*) and NCRS was worse than RS for all k (*loss*). The *tie* happens only when both classifier are perfect and have MCC equal +1. Table 7.3 displays results for different types of weak classifiers in the NCRS compared

Classifier Type	wins	ties	losses
Decision Tree	8	1	1
Logistic Regression	5	4	1
Naïve Bayes	7	1	2

Table 7.3: Performance of three different types of weak classifiers in terms of wins, ties, and losses

with the unbiased RS. The goal of this comparison is not to choose the best type of weak classifier for NCRS for this task but to determine whether the biased feature sampling process in the NCRS provides better accuracy than unbiased feature sampling in RS. The NCRS had only a minority of losses, while most of the times it was better than the RS method or both NCRS and RS made perfect classification (*tie*). However, this comparison is optimistically biased because NCRS was considered to be the winner if it was better for any value of k — in real case scenario, the parameter k could be either determined using internal cross-validation or by heuristic proposed in [10]. On the other hand, the NCRS was better in terms of MCC for any $k \in \{1, 2, \ldots, 14, 15\}$ for many tasks — the k independent results are displayed in Table 7.4 — therefore, the optimistic bias is not present in those experiments as this table only contains results that hold for any value of $k \in \{1, 2, \ldots, 14, 15\}$.

However, the results displayed in Table 7.3 and Table 7.4 are just to compare whether the NCRS with the particular type of weak classifiers is better than the random subspace (RS) method with the same type of weak classifiers, they do not compare the suitability of used weak classifiers for the task as they do not show the absolute accuracy over the datasets. Better picture is provided by Figure A.5 from Appendix A , where there are

Classifier Type	wins	ties	losses
Decision Tree	5	1	1
Logistic Regression	3	1	1
Naïve Bayes	6	1	2

Table 7.4: Performance of three different types of weak classifiers in terms of wins, ties, and losses, which were consistent for any $k \in \{1, 2, \ldots, 14, 15\}$

compared NCRSs with these three types of weak classifiers for $k \in \{1, 2, \ldots, 14, 15\}$ and also with the unbiased RS method with the same type of weak classifiers. The suitability of the particular type of weak classifiers for the NCRS differs from dataset to dataset, different types were dominating for different datasets, there is no clear winner, however, the NCRS with Logistic Regression classifier performs very well as is depicted in Table 7.5.

Task	NCRS DT	NCRS LR	NCRS NB	RS DT	RS LR	RS NB
BMBT DT5q	**0.76**	0.36	0.38	0.34	0.46	0.10
BMH ABT5q	1.00	1.00	1.00	1.00	1.00	1.00
BMH ABTnon-5q	**1.00**	**1.00**	**1.00**	0.87	0.90	0.87
BMH ADT5q	0.72	0.79	**0.81**	0.75	0.66	0.71
BMnon-5q 5qBT	**1.00**	**1.00**	0.75	0.66	0.79	0.62
PBBT DT5q	0.57	**0.79**	0.33	0.57	0.62	0.14
PBH ABT5q	0.99	**1.00**	0.82	**1.00**	**1.00**	0.84
PBH ABTnon-5q	0.83	**1.00**	0.84	0.81	**1.00**	0.65
PBH ADT5q	**1.00**	0.93	0.56	0.92	**1.00**	0.66
PBnon-5q 5qBT	0.96	**1.00**	0.82	0.86	**1.00**	0.64
Average MCC	0.88	**0.89**	0.73	0.79	0.84	0.62
Average rank	2.85	**2.20**	3.85	4.00	2.70	5.40

Table 7.5: Comparison of performance of different types of weak classifiers for both NCRS and RS ensembles. The MCC values are taken as the maximum MCC for $k \in \{1, 2, \ldots, 14, 15\}$ for given classifier

According to Table 7.5, the best classifier for these tasks is NCRS with Logistic Regression weak classifiers. However, the original NCRS with Decision Trees is also very close to the NCRS LR — both in the rank and the average MCC. These two classifiers performs similarly for most tasks but there is a significant difference in their performance for several tasks. From the unbiased RS classifiers, the RS with Logistic Regression weak classifiers performs also very well, thus it seems that this type of task is very suitable for ensembles with Logistic Regression classifiers. On the other hand, the ensembles with Naïve Bayes classifiers did not perform as well as the others, the NCRS NB was the best only in one task and the difference between the NCRS LR was only 2 percentage points in that task. However, the NCRS NB still outperformed both RS DT and RS NB in terms of ranks. Also, the RS NB was consistently the worst from tested classifiers, it seems that Naïve

Bayes ensembles are not as suitable as other ensembles for this type of tasks.

These experiments were biased because the best value of k based on the performance on the *test* set was chosen, better approach would be use internal cross-validation for determining the optimal value of k and then use this value on the *test* set, however, the dasasets are very small — from 13 to 23 samples — and another cross-validation would reduce the training or testing set even further. Even though it would still be possible, for example using Leave-one-out cross-validation, the experiments would be computationally too costly, moreover, the k is to be set using proposed heuristic in [9, 10], therefore the cross-validation would not simulate the real use of the method. The heuristic also cannot be used for comparison as it is tree specific — modification of the heuristic for other learners is part of possible future work. Furthermore, the purpose of this experiment was to show that other weak classifiers are also suitable alternative to Decision tree.

The conclusion arising from this part of experiments is clear — the NCRS method is suitable also for other types of weak classifiers than just the Decision tree. The NCRS method outperformed the RS method in most tasks for any of the three tested types of weak classifiers. In terms of absolute performance, the NCRS with Logistic Regression weak classifiers outperformed other ensemble classifiers both in ranks and average MCCs.

7.1.6.2 Analysis of diversity

The analysis of the relationships between the parameter of the walk length k and the diversity among classifiers in the ensemble is difficult because there are two main characteristics that are dependent on the parameter k — *diversity* and *weak classifiers accuracy* — and they cannot be analysed individually. For these reason, four diversity measures were chosen — two non-pairwise measures *entropy* and *Kohavi-Wolpert measure*(KW) and two pairwise measures *average Q statistics*(Q_{ave}) and *double-fault measure*(DF), all of them are further described in Section 5.2.

However, the diversity is very desirable for obtaining more accurate ensembles, yet if the diversity is gained by having less accurate weak classifiers, it is not better to have more diverse ensemble. For example, lets have an ensemble with 100 weak classifiers whose errors are not correlated, if all the weak classifiers had accuracy 0.9, the ensemble would be less diverse then if the weak classifiers had accuracy 0.6, even though the ensemble with more accurate weak classifiers would be also more accurate. Because of this reason, the double-fault pairwise measure is very important because it represents the percentage of samples that have been misclassified by both of the weak classifiers, therefore, the percentage tends to be smaller with more accurate weak classifiers and it is equal 0 in the extreme case of an ensemble with perfect weak classifiers. However, the double-fault measure has also a disadvantage — in the extreme case, it would not recognize that all weak classifiers are perfect and that therefore we do not need the computationally costly ensemble. Thus, we employ other three diversity measures as well and in same cases, also the *average MCC of weak classifiers* (AWMCC).

As proposed in [9, 10], the diversity indeed seems to decrease with the length of random walk k as the weak classifiers become less and less specialized. It is nicely shown in

figures A.1j, A.3j and A.4j, where there are plotted four different diversity measures. The measures can represent higher diversity with higher values (entropy measure) or with lower values (Kohavi-Wolpert measure, Average Q statistics, double-fault measure), thus for some measures the axis are reversed so all diversity measures show increasing diversity when they are increasing. On the other hand, the average MCC of weak classifiers is increasing with the length k in most cases. Therefore, the overall MCC of the ensemble is based on the proportion between the diversity growth and the weak classifiers accuracy growth. In many cases, the optimal value of k in terms of MCC performance lies in the extreme cases — either for low values of k or high values of k — i.e., the MCC line is either decreasing or increasing with growing $k \in \{1, 2, \ldots, 14, 15\}$. However, for several task, even the weak classifier accuracy is decreasing with increasing k, which, together with decreasing diversity, leads usually to a worse ensemble than would be the unbiased RS ensemble.

Moreover, for many task, the MCC values rocket between $k = 1$ and $k = 2$ — this happens when the seeds from which the random walk begins, have less immediate neighbours in the network than is the desirable number of features. This leads to ensemble whose weak classifiers may be fitted using less features than the ensemble generated for higher values of k — e.g., when a seed has edges only to 50 neighbouring nodes, the random walk for $k = 1$ can sample only 50 features, moreover, these features will all be the same 50 features for all walks from this particular seed. When the length is $k = 2$, the biological network is usually dense enough around the seed and all 100 features may be sampled. Therefore the weak classifier get smaller feature subspace for $k = 1$ than for $k = 2$ and higher. For $k > 1$, the feature subspace has usually the same dimension for all weak classifiers, possibly only containing different features. This effect is the most present in Figure 7.1 (Taken from A.1i) but it is visible, for example, in Figure A.3b, A.3c, A.3i or A.3j. On the other hand, this effect might be sometimes positive – as in Figure A.3d. However, this effect

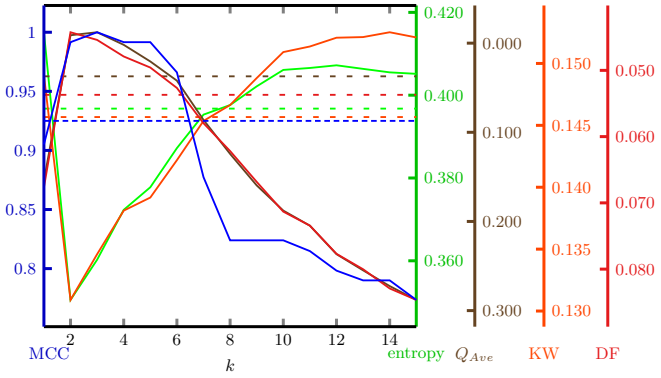


Figure 7.1: The effect caused by smaller number of obtained features for $k = 1$. The graph represents task PBH_ADT5q classified using NCRS with Decision trees weak classifiers.

is not the only one influencing the behavior around $k = 2$ — if it was the case, other characteristics such as weak classifiers' accuracy or diversity measure would change mostly between $k = 1$ and $k = 2$. In Figure A.3c other values steadily change for $k > 2$, which means that the processes inside the ensemble are more complicated and the only reason

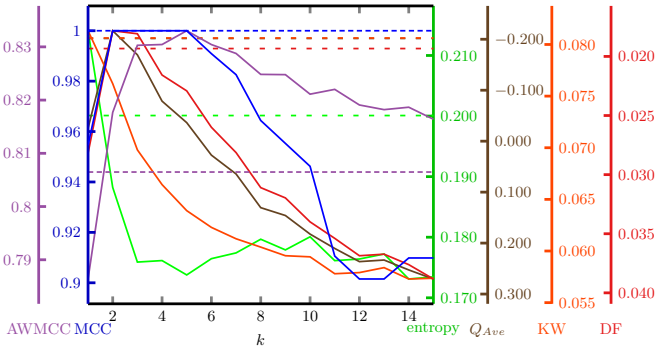
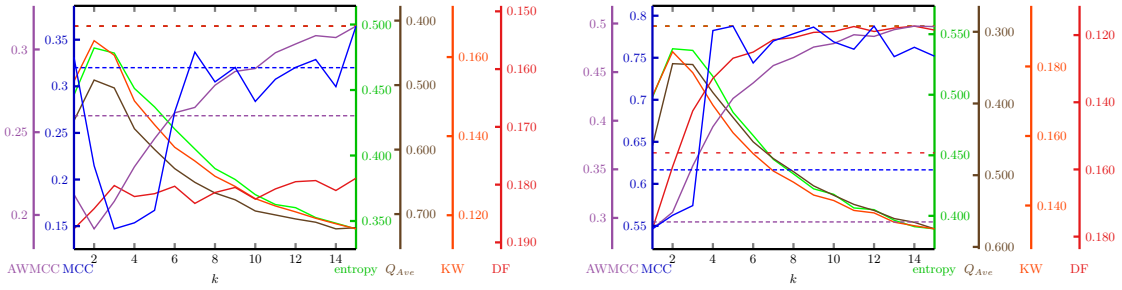


Figure 7.2: The tradeoff between the weak classifiers' diversity and the accuracy. The graph represents task BMH_ABT5q classified using NCRS with Logistic Regression weak classifiers.

they are not visible in MCC line is that the ensemble made perfect classification. On the other hand, this effect might be also visible in other measures while it is not shown in the MCC line — e.g., Figure A.3f, where the entropy and KW measures significantly increase between $k = 1$ and $k = 2$ and only then they begin their steady decrease as the weak classifiers becomes less and less diverse while there is no significant change in the MCC line between $k = 1$ and $k = 2$.

The overall MCC of the ensemble is therefore the result of proportion of its weak classifiers accuracy and diversity. This is nicely shown in Figure 7.2 where the ensemble starts with diverse weak classifiers with lower accuracy for $k = 1$, then the diversity is decreasing, however, the weak classifiers' accuracy is steeply increasing, therefore the overall MCC of the ensemble reaches 1.0 and holds there while the diversity is still decreasing and the weak classifiers' accuracy slowly increasing. However, even though the weak classifiers accuracy is increasing, they tend more and more to have correlated errors — these errors have bigger influence than the increasing accuracy and the double fault measure starts to decrease. At some point the accuracy of weak classifiers begin to slowly decrease but since the ensemble diversity is very low at this point, the overall ensemble MCC plummets — the decrease in MCC is not proportional to the decrease in the AWMCC — roughly 1.5 % for the AWMCC while about 9 % for the MCC.

Another example is in Figure 7.3. In the 7.3a, the overall diversity is decreasing while the AWMCC is increasing steeply, however, because of the loss of diversity, the weak classifiers tend to misclassify the same samples — the double fault measure is increasing only slowly in contrast with the significant increase in the average accuracy of weak classifiers, thus the overall ensemble MCC, unsurprisingly, is only a bit higher than the average weak classifiers' accuracy — 36 % MCC compared to 31 % AWMCC for $k = 15$. However, for lower values of k, it seems that the ensemble contains few accurate weak classifiers pulling the AWMCC up while the majority of weak classifiers have lower accuracy and also lower diversity, which results that the ensemble MCC can be actually lower than the AWMCC — e.g., 17 % MCC compared to 24 % AWMCC for $k = 5$. A possible explanation for this



(a) BMBT_DT5q (b) PBBT_DT5q

Figure 7.3: These are examples of two possible behavior of DF and AWMCCC.

phenomenon might be that specifically for this particular task, only several seeds are actually important while others provide too much noise, thus there are accurate weak classifiers started from several seeds but the rest is much less accurate, however, this hypothesis needs a further analysis. The situation is different in 7.3b, where the double fault measure rises more steeply than the average accuracy of weak classifiers, which causes the MCC rocket because the weak classifiers get accurate enough while having comparably high diversity. Therefore the ensemble works much better than a single weak classifier — it reaches 79 % MCC while having only 42 % AWMCC in the peak for $k = 5$. Even though this peak might be just caused by a chance, the situation is still very good for $k = 6$: 74 %MCC with 44 % AWMCC. A similar situation is in Figure A.3b, where the MCC is 100 % while the AWMCC is just 78 %, or in Figure A.3j with the overall MCC 100 % and the AWMCC 81 %.

As a whole, the NCRS algorithm manages the diversity nicely, in most cases, it starts with specialized and diverse weak classifiers and with increasing value of the parameter k, the diversity usually decreases and the average accuracy of weak classifiers increases. Tuning the random walk length k may allow to find the optimal trade-off between the diversity and the AWMCC resulting in high MCC of the whole ensemble. Only in several cases, the NCRS ends up with unexpected distribution of weak classifiers with higher AWMCC than the overall MCC, this phenomenon requires further analysis. However, it occurs only for particular combination of the dataset and the type of weak classifier, moreover it is often occurs only for particular values of k.

7.1.6.3 Analysis of the convergence of NCRS

As was described in Section 7.1.4, the NCRS converges to a stationary distribution of a random walk for $k \to \infty$ where no miRNA nodes are present in the network. The goal of this experiment was to empirically validate the convergence, therefore this experiments utilises only candidate causal genes and mRNA interactions as prior knowledge. Parameter k was chosen from $\{2, 4, 6, 8, 10, 15, 20, 30, 40, 60, 80, 100, 150, 200\}$. The NCRS algorithm was also modified in such a way that it samples the features with a probability

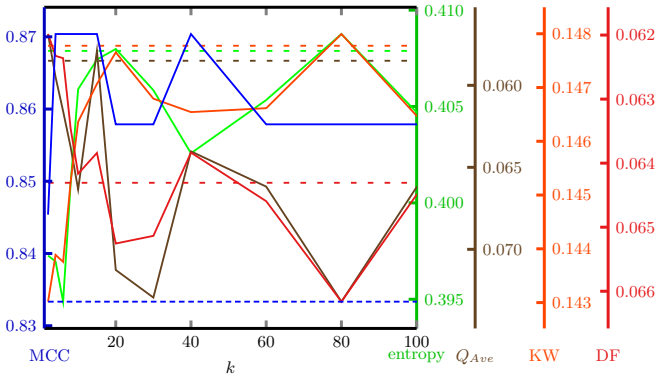


Figure 7.4: The measures do not seem to be converging to the values obtained using the modified NCRS that is sampling features with a probability proportional to their degree. However, the scales of axes are important — the chaotic behavior are just small fluctuations near the desirable values. Task BMH_ABTnon-5q with no miRNA interaction.

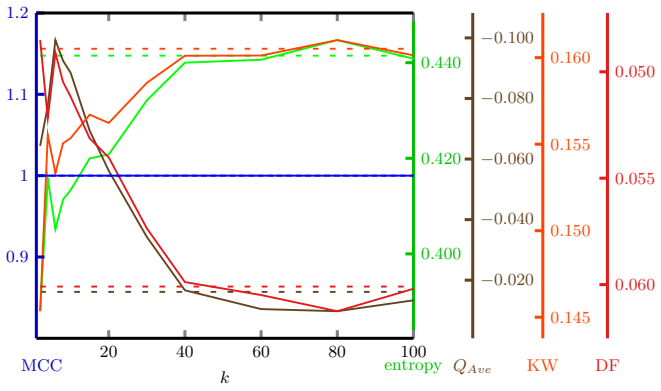


Figure 7.5: The NCRS ideally converge in this task for $k \to \infty$. Task PBH_ABT5q5q with no miRNA interaction.

$\pi^{\infty}(v) = \frac{\deg(v)}{|\mathcal{I}|}$ — i.e., the probability of feature being sampled is the proportional probability of its degree in the biological network. The results of this experiment are depicted in Figure A.6 , where the dotted lines represent values of measures for the k independent degree proportional sampling NCRS, while the full lines represent the k dependent random walk sampling NCRS. In contrast to other plots, the scale of axis is very important in analysis of the convergence — e.g., seemingly unconverging lines might be just caused by small fluctuations caused by the stochastic nature of the classifier as, for example, in Figure 7.4 (Taken from Figure A.6c), where the values seemingly do not converge for increasing values of k, however, the scales of axes are very small, therefore the observed chaotic behavior is just small fluctuations around the desirable values.

On the other hand, the convergence is ideally depicted in Figure 7.5 (Taken from Figure A.6g), where all measures nicely converge to values obtained by the modified NCRS

for higher values of k. The convergence is also shown in other task, albeit not as nicely, however, it seems that the values are converging to different values in several tasks or that the convergence is biased a bit for some reason, e.g., Figure A.6h. However even though it is possible that there is some bias, there are also other other two possible explanation for such phenomenon — first it might be just a fluctuation of the stochastic based original NCRS but secondly, more importantly, it might be caused by the stochastic nature of the modified NCRS as well. When there are changes due to the stochastic nature of the original NCRS, random fluctuation are expected as we fit the classifiers for different values of k and these fluctuations show in the smoothness of the measured points, however when dealing with stochastic nature of the modified NCRS, only one value is obtained and in spite of 10 repeated n-fold cross-validation, the obtained averaged values might still be significantly different from the hidden true expected values of the modified NCRS.

This experiment strongly suggests that the proposed convergence of NCRS (NCF) holds, even though there are still several tasks which would need further analysis as the values seem to converge to a slightly biased point — e.g., all measures converge properly in Figure A.6e except for the MCC line which seems to converge to different value but, as discussed above, this might be caused by the stochastic nature of the modified NCRS, or by low values of k that were used for the empirical verification.

7.2 Benchmark datasets

This experiments contain multiple dataset, the same as in [94]. These datasets are considered to be benchmark datasets in the field of classification using *gene expression* [94]. However, these datasets are different from the MDS datasets — they do not contain miRNA data and also the prior knowledge does not utilise candidate causal genes. The thorough description of used datasets is in [94], the following section contains only a short, simplified description.

7.2.1 Used data

The benchmark data are from a different biological domain, they do not include a miRNA data. The were obtained from two different platforms — *GPL80* and *GPL96*. The number of features used for the classification was reduced from the original number because several genes were not present in the biological network and their inclusion would give significant advantage to classifiers not utilizing prior knowledge as these genes could be important for classification, therefore only the genes present in the interaction network were used for the classification. The number of features per platform is depicted in Table 7.6.

Table 7.6: Used platforms and the number of used features

Platform	Full number	Used number
GPL80	7129	6065
GPL96	22283	19931

Table 7.7: Number of samples in individual datasets

Dataset	Platform	Number of samples
ALL/AML	GPL80	72
Gastric cancer	GPL80	30
Hypertension	GPL80	20
Smoking	GPL80	44
AML	GPL96	64
Breast cancer	GPL96	29
Glioma	GPL96	85
MGCT	GPL96	27
Prostate cancer	GPL96	20
Sarcoma/Hypoxia	GPL96	54

Furthermore, used datasets have very different sizes — from 20 samples to 85 samples. Therefore, validations of different datasets have different accuracy — e.g., one outlier in a dataset with 20 samples do much more harm than an outlier in a dataset with 80 samples. The number of samples in individual datasets is available in Table 7.7.

7.2.2 Prior knowledge

As in MDS experiments, experimentally validated protein-protein interactions are from Human Protein Reference Database [143], predicted protein-protein interactions are from [23]. However, no miRNA prior knowledge was used because these datasets do not contain miRNA expression levels. Moreover, no candidate causal genes were used as seeds for the NCRS algorithm from two main reasons — the NCF outperforms the RS DT even without the miRNAs [9] and for several datasets, e.g., *smoking*, no candidate causal genes are available.

7.2.3 Implementation

This experiment uses the same implementation as the MDS experiment, however, only three diversity measures were computed due to computation costs of pairwise measures — the entropy, the Kohavi-Wolpert measure, and the average Q statistics. Furthermore, three different parametrization of NCRS were used: NCRS 1000:100 with 1000 weak classifiers having access to 100 features, NCRS 100:100 with 100 weak classifiers accessing to 100 features each and NCRS 500:50 with 500 weak classifiers and 50 features per classifier. These k dependent classifiers were compared to RS 1000:100. All classifiers were using decision trees as weak classifiers in order to be equivalent with original NCF from [9, 10].

7.2.4 Purpose of experiments

This experiment was set up to evaluate whether the NCRS (NCF) is also suitable to this more general domain. However, this datasets do not have as extensive prior knowledge

as the MDS datasets therefore the evaluation cannot be complete, however, datasets with both mRNA, miRNA and candidate causal genes are much less frequent than just datasetes with mRNA gene expression profiles.

7.2.5 Experimental Protocol

The protocol was the same as with MDS datasets — 10 times repeated Stratified m-fold cross-validation was used, where $m := \min\{10, c\}$, where c is the number of samples in the smallest class.

7.2.6 Results

These datasets were not as suitable for NCRS algorithm as MDS datasets, the performance of NCRS was nearly k independent, therefore the NCRS algorithm mostly performed similarly as the RS method. There were only small changes in the MCC or diversity measures for any $k \in \{1, 2, \ldots, 14, 15\}$, only with exception for $k = 1$. For this value of k, the feature sampling process of NCRS might end up with less features than demanded due to limited number of neighbouring nodes.

However, the NCRS slightly, but consistently, outperformed the RS DT for several datasets — *breast cancer*, *glioma* and *prostate cancer*, moreover, it significantly outperformed RS on the *hypertension* datasets, while being worse only for one dataset — *smoking* — but not very significantly. Furthermore, to the extent of our knowledge, there are no underlying biological causes behind smoking, which makes the NCRS method less suitable for such datasets.

The NCRS method is not therefore as useful for these datasets as for the MDS datasets, however, it slightly outperforms the RS method in most cases. Moreover, the prior knowledge was not complete for these datasets —no miRNA data and candidate causal genes were used in the experiment.

Chapter 8

Conclusion

The analysis of omics domain is very important because it may allow researchers and doctors to further understand and predict the onset and progression even of heterogenous multifactorial diseases such as myelodysplastic syndrome (MDS). However, this task is very difficult and has many pitfalls. One of those pitfall is overfitting, which can be addressed in several different ways. One possible method for achieving better generalization is the use of ensemble of classifiers, however, there are many ways to combine classifier to an ensemble, therefore a brief review and description of the basic ensemble taxonomy was provided in Chapter 3. This taxonomy described many different ways of ensemble creation with many references to complicate and state-of-art methods proposed in the literature and it gives readers an introduction into the ensemble problematic. In following Chapter 4, a short description of used weak classifiers was provided. Chapter 5 provided theoretical foundations and an insight into the problematic of diversity in ensembles, showed why the diversity is crucial for ensembles and that it is the diversity that allows ensembles synergy, to be more accurate than are its individual weak classifiers. Furthermore, this chapter defined several diversity measures proposed in the literature and referenced a few more different measures.

In Chapter 6, the problem of overfitting caused by the $n \ll p$ problem in machine learning from genomics data was described and review of approaches to this problem in literature was provided. Afterwards, the chapter focused on one particular way to deal with the problem — combing ensembles with the use of prior knowledge, which represents approach that is not well researched in the literature, there were only several attempts before the NCF was proposed in [9, 10] as far as we know. The NCF was also shortly described in this chapter and on top of that we have proposed a simple generalization of NCF called *network-constrained random subspace* method (NCRS), which utilises the idea from NCF but extends it to a general ensemble approach suitable even for other weak classifiers than just *decision trees* as in original NCF. Finally, in Chapter 7, the NCRS was empirically validated using same datasets as in the original study [9, 10]. It was conclusively shown that the generalized method NCRS is suitable for different types of weak classifiers — the experiment tested *Logistic Regression* and *Naïve Bayes* classifiers as well as *Decision trees* that were used in [9, 10]. Furthermore, the NCRS with Logistic

Regression weak classifiers outperforms the originally proposed NCF (NCRS DT) on most MDS datasets. Importantly, both Naïve Bayes and Logistic regression classifiers provide insight into to problem as they allow to easily analyze feature importance — this is very important in this field and it is the reason why are such methods preferred over black-box models such as SVM [10]. Furthermore, the role of diversity in NCRS (NCF) was observed using theory and diversity measures from Chapter 5. As the feature sampling process in NCRS is parametrized by the length of random walk k, we have analysed its influence on the diversity and accuracy for the MDS datasets. Moreover, we have empirically shown that the diversity usually decreases with increasing the length k as was hinted, but not tested, in [9, 10]. The last experiments for the MDS datasets were validating the convergence of NCF (NCRS) for $k \to \infty$ proposed in [9]. After that, we have tested the behavior of NCRS on benchmark datasets from [94], which, in contrast with MDS datasets, do not have miRNA data and candidate causal genes. The NCRS performed similarly as RS on most of these datasets but it slightly outperformed the RS on several datasets and was outperformed by the RS method on only one datasets.

Chapter 9

Future work

There are many things that will be done in the future work. First, as the gene expression data together with miRNA data will be cheaper and more common, the experiments will be replicated using more data, therefore the results will be more statistically relevant and moreover, it will be possible to validate obtained results using different datasets for different tasks and on top of that it will be possible to analyse the influence of the size of the training set on the performance of NCRS compared to the unbiased RS — it is expected that with more data, the prior knowledge will be less and less important as it will be possible to obtain the knowledge from the data, however, datasets in near future are expected to have at most several hundreds sample, which, in comparison with 20,000 features, might still represent a too small training set and the prior knowledge might be still very useful. Moreover, the future work will integrate other types of data and prior knowledge into the NCRS method —- such as DNA methylation arrays — however, the datasets with complete information (GE, miRNA, DNA methylation...) are still very rare or non-existing. Moreover, the biological domain is not the only one where the prior knowledge is available in the form of networks, therefore the future work will also modify the NCRS method for other tasks, e.g., document topic prediction or click prediction, and analyse its performance on such datasets. Also future work will contain a modified heuristic for finding the optimal length of random walk k that would applicable for ensembles of general weak classifiers, not just the NCF.

And, as was stated in [97], the effect of diversity on ensembles is not yet well researched, therefore further analysis of the diversity in the NCRS will be needed.

Bibliography

[1] A. Ahmad and G. Brown. "Random Projection Random Discretization Ensembles — Ensembles of Linear Multivariate Decision Trees". In: *IEEE Transactions on Knowledge and Data Engineering* 26.5 (May 2014), pp. 1225–1239. DOI: 10.1109/tkde.2013.134. URL: http://dx.doi.org/10.1109/TKDE.2013.134.

[2] M. A. H. Akhand, M. M. H. Rahman, and K. Murase. "Decision tree ensemble construction incorporating feature values modification and random subspace method". In: *2014 International Conference on Informatics, Electronics & Vision (ICIEV)*. Institute of Electrical & Electronics Engineers (IEEE), May 2014. DOI: 10.1109/iciev.2014.6850822. URL: http://dx.doi.org/10.1109/ICIEV.2014.6850822.

[3] M. A. H. Akhand, P. C. Shill, and K. Murase. "Neural network ensembles based on Artificial Training Examples". In: *2009 12th International Conference on Computers and Information Technology*. Institute of Electrical & Electronics Engineers (IEEE), Dec. 2009. DOI: 10.1109/iccit.2009.5407262. URL: http://dx.doi.org/10.1109/ICCIT.2009.5407262.

[4] G. Alanis-Lobato, C. V. Cannistraci, and T. Ravasi. "Exploitation of genetic interaction network topology for the prediction of epistatic behavior". In: *Genomics* 102.4 (Oct. 2013), pp. 202–208. DOI: 10.1016/j.ygeno.2013.07.010. URL: http://dx.doi.org/10.1016/j.ygeno.2013.07.010.

[5] K. M. Ali and M. J. Pazzani. "Error reduction through learning multiple descriptions". In: *Machine Learning* 24.3 (Sept. 1996), pp. 173–202. DOI: 10.1007/bf00058611. URL: http://dx.doi.org/10.1007/BF00058611.

[6] A. Altmann et al. "Comparison of Classifier Fusion Methods for Predicting Response to Anti HIV-1 Therapy". In: *PLoS ONE* 3.10 (Oct. 2008). Ed. by D. Unutmaz, e3470. DOI: 10.1371/journal.pone.0003470. URL: http://dx.doi.org/10.1371/journal.pone.0003470.

[7] M. F. Amasyali and O. K. Ersoy. "Comparison of single and ensemble classifiers in terms of accuracy and execution time". In: *2011 International Symposium on Innovations in Intelligent Systems and Applications*. IEEE, June 2011. DOI: 10.1109/inista.2011.5946119. URL: http://dx.doi.org/10.1109/INISTA.2011.5946119.

[8] M. Anděl, J. Kléma, and Z. Krejčík. "Integrating mRNA and miRNA expressions with interaction knowledge to predict myelodysplatic syndromee". In: *In ITAT 2013: Information Technologies – Applications and Theory, Workshop on Bioinformatics in Genomics and Proteomics*. SCITEPRESS - Science, 2013, pp. 48–55. URL: http://ida.felk.cvut.cz/klema/publications/ITAT/itat_andel.pdf.

[9] M. Anděl, J. Kléma, and Z. Krejčík. "Network-constrained forest for regularized classification of omics data". In: *Methods* (Apr. 2015). DOI: 10.1016/j.ymeth.2015.04.006. URL: http://dx.doi.org/10.1016/j.ymeth.2015.04.006.

[10] M. Anděl, J. Kléma, and Z. Krejčík. "Network-constrained forest for regularized omics data classification". In: *2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, Nov. 2014. DOI: 10.1109/bibm.2014.6999193. URL: http://dx.doi.org/10.1109/BIBM.2014.6999193.

[11] D. Anguita, S. Ridella, and D. Sterpi. "Testing the Augmented Binary Multiclass SVM on Microarray Data". In: *The 2006 IEEE International Joint Conference on Neural Network Proceedings*. IEEE, 2006. DOI: 10.1109/ijcnn.2006.246941. URL: http://dx.doi.org/10.1109/IJCNN.2006.246941.

[12] K. J. Archer and R. V. Kimes. "Empirical characterization of random forest variable importance measures". In: *Computational Statistics & Data Analysis* 52.4 (Jan. 2008), pp. 2249–2260. DOI: 10.1016/j.csda.2007.08.015. URL: http://dx.doi.org/10.1016/j.csda.2007.08.015.

[13] I. L. Aroquiaraj and K. Thangavel. "Unsupervised Feature Selection in Digital Mammogram Image Using Tolerance Rough Set Based Quick Reduct". In: *2012 Fourth International Conference on Computational Intelligence and Communication Networks*. Institute of Electrical & Electronics Engineers (IEEE), Nov. 2012. DOI: 10.1109/cicn.2012.202. URL: http://dx.doi.org/10.1109/CICN.2012.202.

[14] L. Auret and C. Aldrich. "Empirical comparison of tree ensemble variable importance measures". In: *Chemometrics and Intelligent Laboratory Systems* 105.2 (Feb. 2011), pp. 157–170. DOI: 10.1016/j.chemolab.2010.12.004. URL: http://dx.doi.org/10.1016/j.chemolab.2010.12.004.

[15] L. Auret and C. Aldrich. "Interpretation of nonlinear relationships between process variables by use of random forests". In: *Minerals Engineering* 35 (Aug. 2012), pp. 27–42. DOI: 10.1016/j.mineng.2012.05.008. URL: http://dx.doi.org/10.1016/j.mineng.2012.05.008.

[16] M. Banerjee, M. Capozzoli, L. McSweeney, and D. Sinha. "Beyond kappa: A review of interrater agreement measures". In: *Canadian Journal of Statistics* 27.1 (Mar. 1999), pp. 3–23. DOI: 10.2307/3315487. URL: http://dx.doi.org/10.2307/3315487.

[17] R. C. Barros, M. P. Basgalupp, A. A. Freitas, and A. C. P. L. F. de Carvalho. "Evolutionary Design of Decision-Tree Algorithms Tailored to Microarray Gene Expression Data Sets". In: *IEEE Transactions on Evolutionary Computation* 18.6 (Dec. 2014), pp. 873–892. DOI: 10.1109/tevc.2013.2291813. URL: http://dx.doi.org/10.1109/TEVC.2013.2291813.

[18] R. C. Barros, M. P. Basgalupp, A. C. P. L. F. de Carvalho, and A. A. Freitas. "A Survey of Evolutionary Algorithms for Decision-Tree Induction". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.3 (May 2012), pp. 291–312. DOI: 10.1109/tsmcc.2011.2157494. URL: http://dx.doi.org/10.1109/TSMCC.2011.2157494.

[19] E. Bauer and R. Kohavi. "An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants". In: *Machine Learning* 36.1/2 (1999), pp. 105–139. DOI: 10.1023/a:1007515423169. URL: http://dx.doi.org/10.1023/A:1007515423169.

[20] S. Behnel, R. Bradshaw, C. Citro, L. Dalcin, D. S. Seljebotn, and K. Smith. "Cython: The Best of Both Worlds". In: *Computing in Science & Engineering* 13.2 (Mar. 2011), pp. 31–39. DOI: 10.1109/mcse.2010.118. URL: http://dx.doi.org/10.1109/MCSE.2010.118.

[21] M. Bhardwaj, T. Gupta, T. Grover, and V. Bhatnagar. "An efficient classifier ensemble using SVM". In: *2009 Proceeding of International Conference on Methods and Models in Computer Science (ICM2CS)*. IEEE, Dec. 2009. DOI: 10.1109/icm2cs.2009.5397955. URL: http://dx.doi.org/10.1109/ICM2CS.2009.5397955.

[22] B. Bina, O. Schulte, and H. Khosravi. "LNBC: A Link-Based Naive Bayes Classifier". In: *2009 IEEE International Conference on Data Mining Workshops*. IEEE, Dec. 2009. DOI: 10.1109/icdmw.2009.116. URL: http://dx.doi.org/10.1109/ICDMW.2009.116.

[23] A. Bossi and B. Lehner. "Tissue specificity and the human protein interaction network". In: *Mol Syst Biol* 5 (Apr. 2009). DOI: 10.1038/msb.2009.17. URL: http://dx.doi.org/10.1038/msb.2009.17.

[24] H. Bostrom, R. Johansson, and A. Karlsson. "On evidential combination rules for ensemble classifiers". In: *Information Fusion, 2008 11th International Conference on*. June 2008, pp. 1–8.

[25] L. Breiman. "Bagging predictors". In: *Machine Learning* 24.2 (1996), pp. 123–140. DOI: 10.1023/a:1018054314350. URL: http://dx.doi.org/10.1023/A:1018054314350.

[26] L. Breiman. "Randoms forests". In: *Machine Learning* 45.1 (2001), pp. 5–32. DOI: 10.1023/a:1010933404324. URL: http://dx.doi.org/10.1023/A:1010933404324.

[27] G. Brown, J. Wyatt, R. Harris, and X. Yao. "Diversity creation methods: a survey and categorisation". In: *Information Fusion* 6.1 (Mar. 2005), pp. 5–20. DOI: 10.1016/j.inffus.2004.04.004. URL: http://dx.doi.org/10.1016/j.inffus.2004.04.004.

[28] R. Cai, Z. Hao, and W. Wen. "A Novel Gene Ranking Algorithm Based on Random Subspace Method". In: *2007 International Joint Conference on Neural Networks*. Institute of Electrical & Electronics Engineers (IEEE), Aug. 2007. DOI: 10.1109/ijcnn.2007.4370958. URL: http://dx.doi.org/10.1109/IJCNN.2007.4370958.

[29] T. Can, O. Çamolu, and A. K. Singh. "Analysis of Protein-protein Interaction Networks Using Random Walks". In: *Proceedings of the 5th International Workshop on Bioinformatics*. BIOKDD '05. Chicago, Illinois: ACM, 2005, pp. 61–68. ISBN: 1-59593-213-5. DOI: 10.1145/1134030.1134042. URL: http://doi.acm.org/10.1145/1134030.1134042.

[30] A. Ch and X. Yao. *Multi-objective Ensemble Construction, Learning and Evolution*. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.9418&rep=rep1&type=pdf.

[31] A. Chandra and X. Yao. "Evolving hybrid ensembles of learning machines for better generalisation". In: *Neurocomputing* 69.7-9 (Mar. 2006), pp. 686–700. DOI: 10.1016/j.neucom.2005.12.014. URL: http://dx.doi.org/10.1016/j.neucom.2005.12.014.

[32] B. Chandra and M. Gupta. "Robust approach for estimating probabilities in Naïve–Bayes Classifier for gene expression data". In: *Expert Systems with Applications* 38.3 (Mar. 2011), pp. 1293–1298. DOI: 10.1016/j.eswa.2010.06.076. URL: http://dx.doi.org/10.1016/j.eswa.2010.06.076.

[33] B. Chandra and P. Varghese. "Fuzzy SLIQ Decision Tree Algorithm". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 38.5 (Oct. 2008), pp. 1294–1301. DOI: 10.1109/tsmcb.2008.923529. URL: http://dx.doi.org/10.1109/TSMCB.2008.923529.

[34] M. Chen and S. A. Ludwig. "Fuzzy decision tree using soft discretization and a genetic algorithm based feature selection method". In: *2013 World Congress on Nature and Biologically Inspired Computing*. IEEE, Aug. 2013. DOI: 10.1109/nabic.2013.6617869. URL: http://dx.doi.org/10.1109/NaBIC.2013.6617869.

[35] Z. Chen and W. Zhang. "Integrative Analysis Using Module-Guided Random Forests Reveals Correlated Genetic Factors Related to Mouse Weight". In: *PLoS Computational Biology* 9.3 (Mar. 2013). Ed. by F. P. Roth, e1002956. DOI: 10.1371/journal.pcbi.1002956. URL: http://dx.doi.org/10.1371/journal.pcbi.1002956.

[36] K. C. Chipman and A. K. Singh. "Predicting genetic interactions with random walks on biological networks". In: *BMC Bioinformatics* 10.1 (2009), p. 17. DOI: 10.1186/1471-2105-10-17. URL: http://dx.doi.org/10.1186/1471-2105-10-17.

[37] I. S. M. Daszykowski and B. Walczak. "Dealing with missing values and outliers in principal component analysis". In: *Talanta* 72.1 (Apr. 2007), pp. 172–178. DOI: 10.1016/j.talanta.2006.10.011. URL: http://dx.doi.org/10.1016/j.talanta.2006.10.011.

[38] E. Debie, K. Shafi, C. Lokan, and K. Merrick. "Reduct based ensemble of learning classifier system for real-valued classification problems". In: *2013 IEEE Symposium on Computational Intelligence and Ensemble Learning (CIEL)*. Institute of Electrical & Electronics Engineers (IEEE), Apr. 2013. DOI: 10.1109/ciel.2013.6613142. URL: http://dx.doi.org/10.1109/CIEL.2013.6613142.

[39] T. G. Dietterich. "An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization". In: *Mach. Learn.* 40.2 (Aug. 2000), pp. 139–157. ISSN: 0885-6125. DOI: 10.1023/A:1007607513941. URL: http://dx.doi.org/10.1023/A:1007607513941.

[40] T. G. Dietterich. "Ensemble Methods in Machine Learning". In: *Lecture Notes in Computer Science* 1857 (2000), pp. 1–15. DOI: 10.1007/3-540-45014-9_1. URL: http://web.engr.oregonstate.edu/~tgd/publications/mcs-ensembles.pdf (visited on 12/20/2014).

[41] D. J. Dittman, T. M. Khoshgoftaar, R. Wald, and A. Napolitano. "Comparison of rank-based vs. score-based aggregation for ensemble gene selection". In: *2013 IEEE 14th International Conference on Information Reuse & Integration (IRI)*. IEEE, Aug. 2013. DOI: 10.1109/iri.2013.6642476. URL: http://dx.doi.org/10.1109/IRI.2013.6642476.

[42] D. Dittman, T. Khoshgoftaar, R. Wald, and A. Napolitano. "Similarity analysis of feature ranking techniques on imbalanced DNA microarray datasets". In: *2012 IEEE International Conference on Bioinformatics and Biomedicine*. IEEE, Oct. 2012. DOI: 10.1109/bibm.2012.6392708. URL: http://dx.doi.org/10.1109/BIBM.2012.6392708.

[43] Y. Dong, H. Guo, W. Zhi, and M. Fan. "Class Imbalance Oriented Logistic Regression". In: *2014 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*. IEEE, Oct. 2014. DOI: 10.1109/cyberc.2014.42. URL: http://dx.doi.org/10.1109/CyberC.2014.42.

[44] R. P. W. Duin. "The combining classifier: to train or not to train?" In: *Object recognition supported by user interaction for service robots*. IEEE Comput. Soc, 2002. DOI: 10.1109/icpr.2002.1048415. URL: http://dx.doi.org/10.1109/ICPR.2002.1048415.

[45] H. Dweep, C. Sticht, P. Pandey, and N. Gretz. "miRWalk – Database: Prediction of possible miRNA binding sites by "walking" the genes of three genomes". In: *Journal of Biomedical Informatics* 44.5 (Oct. 2011), pp. 839–847. DOI: 10.1016/j.jbi.2011.05.002. URL: http://dx.doi.org/10.1016/j.jbi.2011.05.002.

[46] H. I. Elshazly, A. M. Elkorany, A. E. Hassanien, and A. T. Azar. "Ensemble classifiers for biomedical data: Performance evaluation". In: *2013 8th International Conference on Computer Engineering & Systems (ICCES)*. IEEE, Nov. 2013. DOI: 10.1109/icces.2013.6707198. URL: http://dx.doi.org/10.1109/ICCES.2013.6707198.

[47] H. Erdogan and M. U. Sen. "A Unifying Framework for Learning the Linear Combiners for Classifier Ensembles". In: *Proceedings of the 2010 20th International Conference on Pattern Recognition*. ICPR '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 2985–2988. ISBN: 978-0-7695-4109-9. DOI: 10.1109/ICPR.2010.731. URL: http://dx.doi.org/10.1109/ICPR.2010.731.

[48] L. Fan, K.-L. Poh, and P. Zhou. "A sequential feature extraction approach for naïve bayes classification of microarray data". In: *Expert Systems with Applications* 36.6 (Aug. 2009), pp. 9919–9923. DOI: 10.1016/j.eswa.2009.01.075. URL: http://dx.doi.org/10.1016/j.eswa.2009.01.075.

[49] Y. Freund and R. E. Schapire. "Experiments with a New Boosting Algorithm". In: *Proceedings of the Thirteenth International Conference on Machine Learning*. Morgan Kaufmann, 1996, pp. 148–156. URL: http://cseweb.ucsd.edu/~yfreund/papers/boostingexperiments.pdf.

[50] J. Friedman, T. Hastie, R. Tibshirani, et al. "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)". In: *The annals of statistics* 28.2 (2000), pp. 337–407. URL: http://people.csail.mit.edu/torralba/courses/6.869/lectures/lecture6/boosting.pdf.

[51] M. J. Gangeh, M. S. Kamel, and R. P. Duin. "Random Subspace Method in Text Categorization". In: *2010 20th International Conference on Pattern Recognition*. Institute of Electrical & Electronics Engineers (IEEE), Aug. 2010. DOI: 10.1109/icpr.2010.505. URL: http://dx.doi.org/10.1109/ICPR.2010.505.

[52] R. Genuer, J.-M. Poggi, and C. Tuleau-Malot. "Variable selection using random forests". In: *Pattern Recognition Letters* 31.14 (Oct. 2010), pp. 2225–2236. DOI: 10.1016/j.patrec.2010.03.014. URL: http://dx.doi.org/10.1016/j.patrec.2010.03.014.

[53] G. Giacinto and F. Roli. "Design of effective neural network ensembles for image classification purposes". In: *Image and Vision Computing* 19.9-10 (Aug. 2001), pp. 699–707. DOI: 10.1016/s0262-8856(01)00045-2. URL: http://dx.doi.org/10.1016/S0262-8856(01)00045-2.

[54] K. Gkirtzou, P. Tsakalides, and P. Poirazi. "Mature miRNA identification via the use of a Naive Bayes classifier". In: *2008 8th IEEE International Conference on BioInformatics and BioEngineering*. IEEE, Oct. 2008. DOI: 10.1109/bibe.2008.4696697. URL: http://dx.doi.org/10.1109/BIBE.2008.4696697.

[55] P. Guan, D. Huang, M. He, and B. Zhou. "Lung cancer gene expression database analysis incorporating prior knowledge with support vector machine-based classification method". In: *Journal of Experimental & Clinical Cancer Research* 28.1 (2009), p. 103. DOI: 10.1186/1756-9966-28-103. URL: http://dx.doi.org/10.1186/1756-9966-28-103.

[56] Y. Guan, C.-T. Li, and Y. Hu. "Random Subspace Method for Gait Recognition". In: *2012 IEEE International Conference on Multimedia and Expo Workshops*. Institute of Electrical & Electronics Engineers (IEEE), July 2012. DOI: 10.1109/icmew.2012.55. URL: http://dx.doi.org/10.1109/ICMEW.2012.55.

[57] P. A. Gutiérrez, C. Hervás-Martínez, and F. J. Martínez-Estudillo. "Logistic Regression by Means of Evolutionary Radial Basis Function Neural Networks". In: *IEEE Trans. Neural Netw.* 22.2 (Feb. 2011), pp. 246–263. DOI: 10.1109/tnn.2010.2093537. URL: http://dx.doi.org/10.1109/TNN.2010.2093537.

[58] D. Haizhou and M. Chong. "Study on Constructing Generalized Decision Tree by Using DNA Coding Genetic Algorithm". In: *2009 International Conference on Web Information Systems and Mining*. IEEE, Nov. 2009. DOI: 10.1109/wism.2009.41. URL: http://dx.doi.org/10.1109/WISM.2009.41.

[59] S. L. Hamilton and J. R. Hamilton. "Predicting in-hospital-death and mortality percentage using logistic regression". In: *Computing in Cardiology (CinC), 2012*. IEEE, Sept. 2012, pp. 489–492. ISBN: 978-1-4673-2076-4. URL: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6420437.

[60] L. Hansen and P. Salamon. "Neural network ensembles". In: *IEEE Trans. Pattern Anal. Machine Intell.* 12.10 (1990), pp. 993–1001. DOI: 10.1109/34.58871. URL: http://dx.doi.org/10.1109/34.58871.

[61] A. Hapfelmeier and K. Ulm. "A new variable selection approach using Random Forests". In: *Computational Statistics & Data Analysis* 60 (Apr. 2013), pp. 50–69. DOI: 10.1016/j.csda.2012.09.020. URL: http://dx.doi.org/10.1016/j.csda.2012.09.020.

[62] A. Hapfelmeier and K. Ulm. "Variable selection by Random Forests using data with missing values". In: *Computational Statistics & Data Analysis* 80 (Dec. 2014), pp. 129–139. DOI: 10.1016/j.csda.2014.06.017. URL: http://dx.doi.org/10.1016/j.csda.2014.06.017.

[63] M. T. Harandi, M. N. Ahmadabadi, B. N. Araabi, A. Bigdeli, and B. C. Lovell. "Directed Random Subspace Method for Face Recognition". In: *2010 20th International Conference on Pattern Recognition*. Institute of Electrical & Electronics Engineers (IEEE), Aug. 2010. DOI: 10.1109/icpr.2010.659. URL: http://dx.doi.org/10.1109/ICPR.2010.659.

[64] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.

[65] T. K. Ho. "The random subspace method for constructing decision forests". In: *IEEE Trans. Pattern Anal. Machine Intell.* 20.8 (1998), pp. 832–844. DOI: 10.1109/34.709601. URL: http://dx.doi.org/10.1109/34.709601.

[66] M. Holec, J. Kléma, F. Železný, J. Bělohradský, and J. Tolar. "Cross-genome knowledge-based expression data fusion". In: *BCBGC 2009: International Conference on Bioinformatics, Computational Biology, Genomics and Chemoinformatics 2009*. 2009, pp. 43–50. URL: http://ida.felk.cvut.cz/klema/publications/BCBGC/bcbgc09.pdf.

[67] M. Holec, V. Gologuzov, and J. Kléma. "miXGENE Tool for Learning from Heterogeneous Gene Expression Data Using Prior Knowledge". In: *2014 IEEE 27th International Symposium on Computer-Based Medical Systems*. IEEE, May 2014. DOI: 10.1109/cbms.2014.8. URL: http://dx.doi.org/10.1109/CBMS.2014.8.

[68] M. Holec, J. Kléma, F. Železný, and J. Tolar. "Comparative evaluation of set-level techniques in predictive classification of gene expression samples". In: *BMC Bioinformatics* 13.Suppl 10 (2011), S15. DOI: 10.1186/1471-2105-13-s10-s15. URL: http://dx.doi.org/10.1186/1471-2105-13-S10-S15.

[69] M. Holec, J. Kléma, F. Železný, and J. Tolar. "Comparative evaluation of set-level techniques in predictive classification of gene expression samples". In: *BMC Bioinformatics* 13.Suppl 10 (2012), S15. DOI: 10.1186/1471-2105-13-s10-s15. URL: http://dx.doi.org/10.1186/1471-2105-13-S10-S15.

[70] M. Holec, F. Železný, J. Kléma, and J. Tolar. "Integrating Multiple-Platform Expression Data through Gene Set Features". In: *Bioinformatics Research and Applications*. Springer Science Business Media, 2009, pp. 5–17. DOI: 10.1007/978-3-642-01551-9_2. URL: http://dx.doi.org/10.1007/978-3-642-01551-9_2.

[71] J.-H. Hong and S.-B. Cho. "The classification of cancer based on DNA microarray data that uses diverse ensemble genetic programming". In: *Artificial Intelligence in Medicine* 36.1 (Jan. 2006), pp. 43–58. DOI: 10.1016/j.artmed.2005.06.002. URL: http://dx.doi.org/10.1016/j.artmed.2005.06.002.

[72] J. D. Hunter. "Matplotlib: A 2D Graphics Environment". In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/mcse.2007.55. URL: http://dx.doi.org/10.1109/MCSE.2007.55.

[73] H. M. Hussain, K. Benkrid, and H. Seker. "Reconfiguration-based implementation of SVM classifier on FPGA for Classifying Microarray data". In: *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, July 2013. DOI: 10.1109/embc.2013.6610186. URL: http://dx.doi.org/10.1109/EMBC.2013.6610186.

[74] J. A. Iglesias, A. Ledezma, and A. Sanchis. "An ensemble method based on evolving classifiers: eStacking". In: *2014 IEEE Symposium on Evolving and Autonomous Learning Systems (EALS)*. IEEE, Dec. 2014. DOI: 10.1109/eals.2014.7009513. URL: http://dx.doi.org/10.1109/EALS.2014.7009513.

[75] S. Imoto, T. Higuchi, T. Goto, K. Tashiro, S. Kuhara, and S. Miyano. "Combining microarrays and biological knowledge for estimating gene networks via Bayesian networks". In: *Computational Systems Bioinformatics. CSB2003. Proceedings of the 2003 IEEE Bioinformatics Conference. CSB2003.* IEEE Comput. Soc, 2003. DOI: 10.1109/csb.2003.1227309. URL: http://dx.doi.org/10.1109/CSB.2003.1227309.

[76] N. Ishii, I. Torii, Y. Bao, and H. Tanaka. "Modified Reduct: Nearest Neighbor Classification". In: *2012 IEEE/ACIS 11th International Conference on Computer and Information Science.* Institute of Electrical & Electronics Engineers (IEEE), May 2012. DOI: 10.1109/icis.2012.72. URL: http://dx.doi.org/10.1109/ICIS.2012.72.

[77] M. M. Islam, X. Yao, and K. Murase. "A constructive algorithm for training cooperative neural network ensembles". In: *IEEE Trans. Neural Netw.* 14.4 (July 2003), pp. 820–834. DOI: 10.1109/tnn.2003.813832. URL: http://dx.doi.org/10.1109/TNN.2003.813832.

[78] J. Jedrzejowicz and P. Jedrzejowicz. "Constructing Ensemble Classifiers from Expression Trees". In: *2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing.* IEEE, Nov. 2010. DOI: 10.1109/3pgcic.2010.48. URL: http://dx.doi.org/10.1109/3PGCIC.2010.48.

[79] L. Jiang, Z. Cai, H. Zhang, and D. Wang. "Not so greedy: Randomly Selected Naive Bayes". In: *Expert Systems with Applications* 39.12 (Sept. 2012), pp. 11022–11028. DOI: 10.1016/j.eswa.2012.03.022. URL: http://dx.doi.org/10.1016/j.eswa.2012.03.022.

[80] W. Jiang, M. Li, H. Zhang, and J. Zhou. "Relevance feedback using random subspace method". In: *2004 IEEE International Symposium on Circuits and Systems (IEEE Cat. No.04CH37512).* Institute of Electrical & Electronics Engineers (IEEE), 2004. DOI: 10.1109/iscas.2004.1329203. URL: http://dx.doi.org/10.1109/ISCAS.2004.1329203.

[81] M. Johannes, H. Frohlich, H. Sultmann, and T. Beissbarth. "pathClass: an R-package for integration of pathway knowledge into support vector machines for biomarker discovery". In: *Bioinformatics* 27.10 (Mar. 2011), pp. 1442–1443. DOI: 10.1093/bioinformatics/btr157. URL: http://dx.doi.org/10.1093/bioinformatics/btr157.

[82] E. Jones, T. Oliphant, P. Peterson, et al. *SciPy: Open source scientific tools for Python.* [Online; accessed 2015-05-12]. 2001–. URL: http://www.scipy.org/.

[83] A. V. Kelarev, A. Stranieri, J. L. Yearwood, and H. F. Jelinek. "Empirical Study of Decision Trees and Ensemble Classifiers for Monitoring of Diabetes Patients in Pervasive Healthcare". In: *2012 15th International Conference on Network-Based Information Systems.* IEEE, Sept. 2012. DOI: 10.1109/nbis.2012.20. URL: http://dx.doi.org/10.1109/NBiS.2012.20.

[84] A. Khemphila and V. Boonjing. "Comparing performances of logistic regression, decision trees, and neural networks for classifying heart disease patients". In: *2010 International Conference on Computer Information Systems and Industrial Management Applications (CISIM)*. IEEE, Oct. 2010. DOI: 10.1109/cisim.2010.5643666. URL: http://dx.doi.org/10.1109/CISIM.2010.5643666.

[85] T. Khoshgoftaar, N. Seliya, and Y. Liu. "Genetic programming-based decision trees for software quality classification". In: *Proceedings. 15th IEEE International Conference on Tools with Artificial Intelligence*. IEEE Comput. Soc, 2003. DOI: 10.1109/tai.2003.1250214. URL: http://dx.doi.org/10.1109/TAI.2003.1250214.

[86] F. Klawonn and P. Angelov. "Evolving Extended Naive Bayes Classifiers". In: *Sixth IEEE International Conference on Data Mining - Workshops (ICDMW'06)*. IEEE, 2006. DOI: 10.1109/icdmw.2006.74. URL: http://dx.doi.org/10.1109/ICDMW.2006.74.

[87] J. Kléma, J. Zahálka, M. Anděl, and Z. Krejčík. "Knowledge-based Subtractive Integration of mRNA and miRNA Expression Profiles to Differentiate Myelodysplastic Syndrome". In: *Proceedings of the International Conference on Bioinformatics Models, Methods and Algorithms*. SCITEPRESS - Science, 2014, pp. 31–39. DOI: 10.5220/0004752200310039. URL: http://ida.felk.cvut.cz/klema/publications/Biotex/Bioinformatics2014_final.pdf.

[88] R. Kohavi and R. Quinlan. "Decision Tree Discovery". In: *IN HANDBOOK OF DATA MINING AND KNOWLEDGE DISCOVERY*. University Press, 1999, pp. 267–276. URL: http://ai.stanford.edu/users/ronnyk/treesHB.pdf.

[89] R. Kohavi and D. H. Wolpert. "Bias Plus Variance Decomposition for Zero-One Loss Functions". In: *MACHINE LEARNING: PROCEEDINGS OF THE THIRTEENTH INTERNATIONAL*. Morgan Kaufmann Publishers, 1996, pp. 275–283.

[90] S. B. Kotsiantis. "Decision trees: a recent overview". In: *Artificial Intelligence Review* 39.4 (June 2011), pp. 261–283. DOI: 10.1007/s10462-011-9272-4. URL: http://dx.doi.org/10.1007/s10462-011-9272-4.

[91] S. Kotsiantis. "Combining bagging, boosting, rotation forest and random subspace methods". In: *Artificial Intelligence Review* 35.3 (Dec. 2010), pp. 223–240. DOI: 10.1007/s10462-010-9192-8. URL: http://dx.doi.org/10.1007/s10462-010-9192-8.

[92] H. C. Kraemer, V. S. Periyakoil, and A. Noda. "Kappa coefficients in medical research". In: *Statist. Med.* 21.14 (2002), pp. 2109–2129. DOI: 10.1002/sim.1180. URL: http://dx.doi.org/10.1002/sim.1180.

[93] P. Krahwinkler, J. Rossmann, and B. Sondermann. "Support vector machine based decision tree for very high resolution multispectral forest mapping". In: *2011 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, July 2011. DOI: 10.1109/igarss.2011.6048893. URL: http://dx.doi.org/10.1109/IGARSS.2011.6048893.

[94] M. Krejník and J. Kléma. "Empirical Evidence of the Applicability of Functional Clustering through Gene Expression Classification". In: *IEEE/ACM Trans. Comput. Biol. and Bioinf.* 9.3 (May 2012), pp. 788–798. DOI: 10.1109/tcbb.2012.23. URL: http://dx.doi.org/10.1109/TCBB.2012.23.

[95] L. I. Kuncheva. *Combining Pattern Classifiers*. John Wiley & Sons, Inc., July 2004. DOI: 10.1002/0471660264. URL: http://dx.doi.org/10.1002/0471660264.

[96] L. I. Kuncheva and J. J. Rodríguez. "An Experimental Study on Rotation Forest Ensembles". In: *Multiple Classifier Systems*. Springer Science Business Media, 2007, pp. 459–468. DOI: 10.1007/978-3-540-72523-7_46. URL: http://dx.doi.org/10.1007/978-3-540-72523-7_46.

[97] L. I. Kuncheva and C. J. Whitaker. "Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy". In: *Mach. Learn.* 51.2 (May 2003), pp. 181–207. ISSN: 0885-6125. DOI: 10.1023/A:1022859003006. URL: http://dx.doi.org/10.1023/A:1022859003006.

[98] B.-C. Kuo, H.-C. Liu, Y.-C. Hsieh, and R.-M. Chao. "A random subspace method with automatic dimensionality selection for hyperspectral image classification". In: *Proceedings. 2005 IEEE International Geoscience and Remote Sensing Symposium, 2005. IGARSS '05.* Institute of Electrical & Electronics Engineers (IEEE), 2005. DOI: 10.1109/igarss.2005.1526131. URL: http://dx.doi.org/10.1109/IGARSS.2005.1526131.

[99] L. Lam, C. Y. Suen, and F. Ieee. "C Y. Application of majority voting to pattern recognition: an analysis of its behavior and performance". In: *IEEE Trans Syst, Man and Cybernet* (1997), pp. 553–568. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.468.3399.

[100] L. Lan and S. Vucetic. "A Multi-task Feature Selection Filter for Microarray Classification". In: *2009 IEEE International Conference on Bioinformatics and Biomedicine*. IEEE, Nov. 2009. DOI: 10.1109/bibm.2009.79. URL: http://dx.doi.org/10.1109/BIBM.2009.79.

[101] O. Lavi, G. Dror, and R. Shamir. "Network-Induced Classification Kernels for Gene Expression Profile Analysis". In: *Journal of Computational Biology* 19.6 (June 2012), pp. 694–709. DOI: 10.1089/cmb.2012.0065. URL: http://dx.doi.org/10.1089/cmb.2012.0065.

[102] S. Lee, M.-S. Kwon, I.-S. Huh, and T. Park. "CUDA-LR: CUDA-accelerated logistic regression analysis tool for gene-gene interaction for genome-wide association study". In: *2011 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW)*. IEEE, Nov. 2011. DOI: 10.1109/bibmw.2011.6112454. URL: http://dx.doi.org/10.1109/BIBMW.2011.6112454.

[103] C. Leng, S. Wang, and H. Wang. "Learning Naive Bayes Classifiers with Incomplete Data". In: *2009 International Conference on Artificial Intelligence and Computational Intelligence*. IEEE, 2009. DOI: 10.1109/aici.2009.402. URL: http://dx.doi.org/10.1109/AICI.2009.402.

[104] C. Li and H. Li. "Network-constrained regularization and variable selection for analysis of genomic data". In: *Bioinformatics* 24.9 (Mar. 2008), pp. 1175–1182. DOI: 10.1093/bioinformatics/btn081. URL: http://dx.doi.org/10.1093/bioinformatics/btn081.

[105] C. Li, W. Zhi, and H. Li. "Network-Based Empirical Bayes Methods for Linear Models with Applications to Genomic Data". In: *Journal of Biopharmaceutical Statistics* 20.2 (Mar. 2010), pp. 209–222. DOI: 10.1080/10543400903572712. URL: http://dx.doi.org/10.1080/10543400903572712.

[106] F.-C. Li, J. Su, and X.-Z. Wang. "Analysis on the fuzzy filter in fuzzy decision trees". In: *Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.03EX693)*. IEEE, 2003. DOI: 10.1109/icmlc.2003.1259723. URL: http://dx.doi.org/10.1109/ICMLC.2003.1259723.

[107] G.-Z. Li, X.-Q. Zeng, J. Y. Yang, and M. Q. Yang. "Partial Least Squares Based Dimension Reduction with Gene Selection for Tumor Classification". In: *2007 IEEE 7th International Symposium on BioInformatics and BioEngineering*. IEEE, Oct. 2007. DOI: 10.1109/bibe.2007.4375763. URL: http://dx.doi.org/10.1109/BIBE.2007.4375763.

[108] K. Li and Y. Han. "Study of selective ensemble learning method and its diversity based on decision tree and neural network". In: *2010 Chinese Control and Decision Conference*. Institute of Electrical & Electronics Engineers (IEEE), May 2010. DOI: 10.1109/ccdc.2010.5498179. URL: http://dx.doi.org/10.1109/CCDC.2010.5498179.

[109] K. Li and L. Hao. "Naïve Bayes ensemble learning based on oracle selection". In: *2009 Chinese Control and Decision Conference*. IEEE, June 2009. DOI: 10.1109/ccdc.2009.5194867. URL: http://dx.doi.org/10.1109/CCDC.2009.5194867.

[110] Y. Li, L. Cui, and K. Li. "Creating Diversity in Ensembles using Clustering Method from Libraries of Models". In: *2006 International Conference on Machine Learning and Cybernetics*. Institute of Electrical & Electronics Engineers (IEEE), 2006. DOI: 10.1109/icmlc.2006.258656. URL: http://dx.doi.org/10.1109/ICMLC.2006.258656.

[111] D. Liu, Y.-y. Huang, and C.-x. Ma. "Feature Extraction and Classification of Proteomics Data Using Stationary Wavelet Transform and Naive Bayes Classifier". In: *2010 4th International Conference on Bioinformatics and Biomedical Engineering*. IEEE, June 2010. DOI: 10.1109/icbbe.2010.5516610. URL: http://dx.doi.org/10.1109/ICBBE.2010.5516610.

[112] H.-C. Liu, Y.-D. Jheng, G.-S. Chen, and B.-C. Jeng. "A new classification algorithm combining Choquet integral and logistic regression". In: *2008 International Conference on Machine Learning and Cybernetics*. IEEE, July 2008. DOI: 10.1109/icmlc.2008.4620936. URL: http://dx.doi.org/10.1109/ICMLC.2008.4620936.

[113] H.-C. Liu, S.-W. Liu, P.-C. Chang, W.-C. Huang, and C.-H. Liao. "A novel classifier for influenza a viruses based on SVM and logistic regression". In: *2008 International Conference on Wavelet Analysis and Pattern Recognition*. IEEE, Aug. 2008. DOI: 10.1109/icwapr.2008.4635791. URL: http://dx.doi.org/10.1109/ICWAPR.2008.4635791.

[114] H.-C. Liu, P.-C. Peng, T.-C. Hsieh, T.-C. Yeh, C.-J. Lin, C.-Y. Chen, J.-Y. Hou, L.-Y. Shih, and D.-C. Liang. "Comparison of Feature Selection Methods for Cross-Laboratory Microarray Analysis". In: *IEEE/ACM Trans. Comput. Biol. and Bioinf.* 10.3 (May 2013), pp. 593–604. DOI: 10.1109/tcbb.2013.70. URL: http://dx.doi.org/10.1109/TCBB.2013.70.

[115] N. E. M. Llerena, L. Berton, and A. de Andrade Lopes. "Graph-based cross-validated committees ensembles". In: *2012 Fourth International Conference on Computational Aspects of Social Networks (CASoN)*. Institute of Electrical & Electronics Engineers (IEEE), Nov. 2012. DOI: 10.1109/cason.2012.6412381. URL: http://dx.doi.org/10.1109/CASoN.2012.6412381.

[116] W.-Y. Loh. "Classification and regression trees". In: *WIREs Data Mining Knowl Discov* 1.1 (Jan. 2011), pp. 14–23. DOI: 10.1002/widm.8. URL: http://dx.doi.org/10.1002/widm.8.

[117] H. H. Manap, N. M. Tahir, and R. Abdullah. "Anomalous gait detection using Naive Bayes classifier". In: *2012 IEEE Symposium on Industrial Electronics and Applications*. IEEE, Sept. 2012. DOI: 10.1109/isiea.2012.6496664. URL: http://dx.doi.org/10.1109/ISIEA.2012.6496664.

[118] R. Manavalan and K. Thangavel. "Quick Reduct-ACO based feature selection for TRUS prostate cancer image classification". In: *International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME-2012)*. Institute of Electrical & Electronics Engineers (IEEE), Mar. 2012. DOI: 10.1109/icprime.2012.6208367. URL: http://dx.doi.org/10.1109/ICPRIME.2012.6208367.

[119] A. Margoosian and J. Abouei. "Ensemble-based classifiers for cancer classification using human tumor microarray data". In: *2013 21st Iranian Conference on Electrical Engineering (ICEE)*. IEEE, May 2013. DOI: 10.1109/iraniancee.2013.6599553. URL: http://dx.doi.org/10.1109/IranianCEE.2013.6599553.

[120] W. McKinney. "Data Structures for Statistical Computing in Python". In: *Proceedings of the 9th Python in Science Conference*. Ed. by S. van der Walt and J. Millman. 2010, pp. 51–56.

[121] Q. Meng, Q. He, N. Li, X. Du, and L. Su. "Crisp Decision Tree Induction Based on Fuzzy Decision Tree Algorithm". In: *2009 First International Conference on Information Science and Engineering*. IEEE, 2009. DOI: 10.1109/icise.2009.440. URL: http://dx.doi.org/10.1109/ICISE.2009.440.

[122] L. L. Minku and X. Yao. "An analysis of multi-objective evolutionary algorithms for training ensemble models based on different performance measures in software effort estimation". In: *Proceedings of the 9th International Conference on Predictive Models in Software Engineering - PROMISE '13*. Association for Computing Machinery (ACM), 2013. DOI: 10.1145/2499393.2499396. URL: http://dx.doi.org/10.1145/2499393.2499396.

[123] J. Mohr, S. Seo, and K. Obermayer. "Automated Microarray Classification Based on P-SVM Gene Selection". In: *2008 Seventh International Conference on Machine Learning and Applications*. IEEE, 2008. DOI: 10.1109/icmla.2008.75. URL: http://dx.doi.org/10.1109/ICMLA.2008.75.

[124] B. Momin, S. Mitra, and R. Gupta. "Reduct Generation and Classification of Gene Expression Data". In: *2006 International Conference on Hybrid Information Technology*. Institute of Electrical & Electronics Engineers (IEEE), Nov. 2006. DOI: 10.1109/ichit.2006.253568. URL: http://dx.doi.org/10.1109/ICHIT.2006.253568.

[125] X. Mu, J. Lu, P. Watta, and M. H. Hassoun. "Weighted voting-based ensemble classifiers with application to human face recognition and voice recognition". In: *2009 International Joint Conference on Neural Networks*. IEEE, June 2009. DOI: 10.1109/ijcnn.2009.5178708. URL: http://dx.doi.org/10.1109/IJCNN.2009.5178708.

[126] R. Musehane, F. Netshiongolwe, F. V. Nelwamondo, L. M. Masisi, and T. Marwala. "Relationship between Diversity and Perfomance of Multiple Classifiers for Decision Support". In: *CoRR* abs/0810.3865 (2008). URL: http://arxiv.org/abs/0810.3865.

[127] L. Nanni and A. Lumini. "FuzzyBagging: A novel ensemble of classifiers". In: *Pattern Recognition* 39.3 (Mar. 2006), pp. 488–490. DOI: 10.1016/j.patcog.2005.10.002. URL: http://dx.doi.org/10.1016/j.patcog.2005.10.002.

[128] A. Navada, A. N. Ansari, S. Patil, and B. A. Sonkamble. "Overview of use of decision tree algorithms in machine learning". In: *2011 IEEE Control and System Graduate Research Colloquium*. IEEE, June 2011. DOI: 10.1109/icsgrc.2011.5991826. URL: http://dx.doi.org/10.1109/ICSGRC.2011.5991826.

[129] S. Nikolić, M. Knežević, V. Ivančević, and I. Luković. "Building an Ensemble from a Single Naive Bayes Classifier in the Analysis of Key Risk Factors for Polish State Fire Service". In: *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems*. IEEE, Sept. 2014. DOI: 10.15439/2014f499. URL: http://dx.doi.org/10.15439/2014F499.

[130] A. Nurnberger, C. Borgelt, and A. Klose. "Improving naive Bayes classifiers using neuro-fuzzy learning". In: *ICONIP'99. ANZIIS'99 & ANNES'99 & ACNN'99. 6th International Conference on Neural Information Processing. Proceedings (Cat. No.99EX378)*. IEEE, 1999. DOI: 10.1109/iconip.1999.843978. URL: http://dx.doi.org/10.1109/ICONIP.1999.843978.

[131] S. Nygård, T. Reitan, T. Clancy, V. Nygaard, J. Bjørnstad, B. Skrbic, T. Tønnessen, G. Christensen, and E. Hovig. "Identifying pathogenic processes by integrating microarray data with prior knowledge". In: *BMC Bioinformatics* 15.1 (2014), p. 115. DOI: 10.1186/1471-2105-15-115. URL: http://dx.doi.org/10.1186/1471-2105-15-115.

[132] D. W. Opitz. "Feature Selection for Ensembles". In: *Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*. AAAI '99/IAAI '99. Orlando, Florida, USA: American Association for Artificial Intelligence, 1999, pp. 379–384. ISBN: 0-262-51106-1. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.500.9132.

[133] D. W. Opitz and J. W. Shavlik. "Generating Accurate and Diverse Members of a Neural-Network Ensemble". In: *Advances in Neural Information Processing Systems*. MIT Press, 1996, pp. 535–541. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.56.69.

[134] D. Opitz and R. Maclin. "Popular ensemble methods: An empirical study". In: *Journal of Artificial Intelligence Research* 11 (1999), pp. 169–198. URL: http://www.d.umn.edu/~rmaclin/publications/opitz-jair99.pdf.

[135] P. Panov and S. Džeroski. "Combining Bagging and Random Subspaces to Create Better Ensembles". In: *Advances in Intelligent Data Analysis VII*. Springer Science Business Media, 2007, pp. 118–129. DOI: 10.1007/978-3-540-74825-0_11. URL: http://dx.doi.org/10.1007/978-3-540-74825-0_11.

[136] D. Partridge and W. Krzanowski. "Software diversity: practical statistics for its measurement and exploitation". In: *Information and Software Technology* 39.10 (Jan. 1997), pp. 707–717. DOI: 10.1016/s0950-5849(97)00023-2. URL: http://dx.doi.org/10.1016/S0950-5849(97)00023-2.

[137] D. Partridge and W. B. Yates. "Engineering Multiversion Neural-Net Systems". In: *Neural Computation* 8.4 (May 1996), pp. 869–893. DOI: 10.1162/neco.1996.8.4.869. URL: http://dx.doi.org/10.1162/neco.1996.8.4.869.

[138] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830. URL: http://arxiv.org/pdf/1201.0490v2.pdf.

[139] S. Peng, X. Liu, J. Yu, Z. Wan, and X. Peng. "A New Implementation of Recursive Feature Elimination Algorithm for Gene Selection from Microarray Data". In: *2009 WRI World Congress on Computer Science and Information Engineering*. IEEE, 2009. DOI: 10.1109/csie.2009.75. URL: http://dx.doi.org/10.1109/CSIE.2009.75.

[140] F. Perez and B. E. Granger. "IPython: A System for Interactive Scientific Computing". In: *Computing in Science & Engineering* 9.3 (2007), pp. 21–29. DOI: 10.1109/mcse.2007.53. URL: http://dx.doi.org/10.1109/MCSE.2007.53.

[141] R. Polikar. "Ensemble based systems in decision making". In: *IEEE Circuits Syst. Mag.* 6.3 (2006), pp. 21–45. DOI: 10.1109/mcas.2006.1688199. URL: http://dx.doi.org/10.1109/MCAS.2006.1688199.

[142] C. Porzelius, M. Johannes, H. Binder, and T. Beißbarth. "Leveraging external knowledge on molecular interactions in classification methods for risk prediction of patients". In: *Biometrical Journal* 53.2 (Feb. 2011), pp. 190–201. DOI: 10.1002/bimj.201000155. URL: http://dx.doi.org/10.1002/bimj.201000155.

[143] T. S. K. Prasad et al. "Human Protein Reference Database–2009 update". In: *Nucleic Acids Research* 37.Database (Jan. 2009), pp. D767–D772. DOI: 10.1093/nar/gkn892. URL: http://dx.doi.org/10.1093/nar/gkn892.

[144] F. Provost and V. Kolluri. "A Survey of Methods for Scaling Up Inductive Algorithms". In: *Data Min. Knowl. Discov.* 3.2 (June 1999), pp. 131–169. ISSN: 1384-5810. DOI: 10.1023/A:1009876119989. URL: http://dx.doi.org/10.1023/A:1009876119989.

[145] J. R. Quinlan. "Bagging, Boosting, and C4.S". In: *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 1*. AAAI'96. Portland, Oregon: AAAI Press, 1996, pp. 725–730. ISBN: 0-262-51091-X. URL: http://dl.acm.org/citation.cfm?id=1892875.1892983.

[146] J. R. Quinlan. "Induction of decision trees". In: *Machine Learning* 1.1 (Mar. 1986), pp. 81–106. DOI: 10.1007/bf00116251. URL: http://dx.doi.org/10.1007/BF00116251.

[147] P. Rani and V. Pudi. "RBNBC: Repeat Based Naive Bayes Classifier for Biological Sequences". In: *2008 Eighth IEEE International Conference on Data Mining*. IEEE, Dec. 2008. DOI: 10.1109/icdm.2008.66. URL: http://dx.doi.org/10.1109/ICDM.2008.66.

[148] F. Rapaport, A. Zinovyev, M. Dutreix, E. Barillot, and J.-P. Vert. "Classification of microarray data using gene networks". In: *BMC Bioinformatics* 8.1 (2007), p. 35. DOI: 10.1186/1471-2105-8-35. URL: http://dx.doi.org/10.1186/1471-2105-8-35.

[149] P. Reshetova, A. K. Smilde, A. H. C. van Kampen, and J. A. Westerhuis. "Use of prior knowledge for the analysis of high-throughput transcriptomics and metabolomics data". In: *BMC Syst Biol* 8.Suppl 2 (2014), S2. DOI: 10.1186/1752-0509-8-s2-s2. URL: http://dx.doi.org/10.1186/1752-0509-8-S2-S2.

[150] I. Rish. *An empirical study of the Naive Bayes classifier*. Tech. rep. Thomas J. Watson Research Center, 2001. URL: http://www.cc.gatech.edu/home/isbell/classes/reading/papers/Rish.pdf.

[151] G. Ritschard. "CHAID and Earlier Supervised Tree Methods". In: *Cahiers du Département d'économétrie*. 2010, pp. 3–6. URL: http://www.unige.ch/ses/metri/cahiers/2010_02.pdf.

[152] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso. "Rotation Forest: A New Classifier Ensemble Method". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.10 (Oct. 2006), pp. 1619–1630. DOI: 10.1109/tpami.2006.211. URL: http://dx.doi.org/10.1109/TPAMI.2006.211.

[153] J. J. Rodríguez and L. I. Kuncheva. "Naïve Bayes Ensembles with a Random Oracle". In: *Multiple Classifier Systems*. Springer Science Business Media, 2007, pp. 450–458. DOI: 10.1007/978-3-540-72523-7_45. URL: http://dx.doi.org/10.1007/978-3-540-72523-7_45.

[154] L. Rokach. "Ensemble-based Classifiers". In: *Artif. Intell. Rev.* 33.1-2 (Feb. 2010), pp. 1–39. ISSN: 0269-2821. DOI: 10.1007/s10462-009-9124-7. URL: http://dx.doi.org/10.1007/s10462-009-9124-7.

[155] L. Rokach. "Taxonomy for characterizing ensemble methods in classification tasks: A review and annotated bibliography". In: *Computational Statistics & Data Analysis* 53.12 (Oct. 2009), pp. 4046–4072. DOI: 10.1016/j.csda.2009.07.017. URL: http://dx.doi.org/10.1016/j.csda.2009.07.017.

[156] E. Ryeng and B. K. Alsberg. "Microarray data classification using inductive logic programming and gene ontology background information". In: *Journal of Chemometrics* 24.5 (Feb. 2010), pp. 231–240. DOI: 10.1002/cem.1263. URL: http://dx.doi.org/10.1002/cem.1263.

[157] R. E. Schapire. "The strength of weak learnability". In: *Machine Learning* 5.2 (June 1990), pp. 197–227. DOI: 10.1007/bf00116037. URL: http://dx.doi.org/10.1007/BF00116037.

[158] A. Scherbart and T. W. Nattkemper. "The Diversity of Regression Ensembles Combining Bagging and Random Subspace Method". In: *Advances in Neuro-Information Processing*. Springer Science Business Media, 2009, pp. 911–918. DOI: 10.1007/978-3-642-03040-6_111. URL: http://dx.doi.org/10.1007/978-3-642-03040-6_111.

[159] D. B. Skalak. "The Sources of Increased Accuracy for Two Proposed Boosting Algorithms". In: *In Proc. American Association for Arti Intelligence, AAAI-96, Integrating Multiple Learned Models Workshop*. 1996, pp. 120–125. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.40.2269.

[160] D. P. Snustad and M. J. Simmons. *Genetika*. 1st ed. Masarykova univerzita, 2009. ISBN: 978-80-210-4852-2.

[161] D. P. Snustad and M. J. Simmons. *Principles of Genetics*. 5th ed. Wiley, Dec. 2008. ISBN: 978-0-470-38825-9.

[162] M. Song and S. Rajasekaran. "A Greedy Correlation-Incorporated SVM-Based Algorithm for Gene Selection". In: *21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07)*. IEEE, 2007. DOI: 10.1109/ainaw.2007.25. URL: http://dx.doi.org/10.1109/AINAW.2007.25.

[163] A. Statnikov, C. F. Aliferis, I. Tsamardinos, D. Hardin, and S. Levy. "A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis". In: *Bioinformatics* 21.5 (Sept. 2004), pp. 631–643. DOI: 10.1093/bioinformatics/bti033. URL: http://dx.doi.org/10.1093/bioinformatics/bti033.

[164] N. Stepenosky, D. Green, J. Kounios, C. M. Clark, and R. Polikar. "Majority Vote and Decision Template Based Ensemble Classifiers Trained on Event Related Potentials for Early Diagnosis of Alzheimer's Disease". In: *2006 IEEE International Conference on Acoustics Speed and Signal Processing Proceedings*. IEEE, 2006. DOI: 10.1109/icassp.2006.1661422. URL: http://dx.doi.org/10.1109/ICASSP.2006.1661422.

[165] F. C. Stingo and M. Vannucci. "Variable selection for discriminant analysis with Markov random field priors for the analysis of microarray data". In: *Bioinformatics* 27.4 (Dec. 2010), pp. 495–501. DOI: 10.1093/bioinformatics/btq690. URL: http://dx.doi.org/10.1093/bioinformatics/btq690.

[166] J. Sun and X.-Z. Wang. "An initial comparison on noise resisting between crisp and fuzzy decision trees". In: *2005 International Conference on Machine Learning and Cybernetics*. IEEE, 2005. DOI: 10.1109/icmlc.2005.1527372. URL: http://dx.doi.org/10.1109/ICMLC.2005.1527372.

[167] F. Tai and W. Pan. "Incorporating prior knowledge of gene functional groups into regularized discriminant analysis of microarray data". In: *Bioinformatics* 23.23 (Oct. 2007), pp. 3170–3177. DOI: 10.1093/bioinformatics/btm488. URL: http://dx.doi.org/10.1093/bioinformatics/btm488.

[168] E. K. Tang, P. N. Suganthan, and X. Yao. "Feature Selection for Microarray Data Using Least Squares SVM and Particle Swarm Optimization". In: *2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*. IEEE, 2005. DOI: 10.1109/cibcb.2005.1594892. URL: http://dx.doi.org/10.1109/CIBCB.2005.1594892.

[169] Y. Tang, W. Pan, H. Li, and Y. Xu. "Fuzzy Naive Bayes classifier based on fuzzy clustering". In: *IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 2002. DOI: 10.1109/icsmc.2002.1176401. URL: http://dx.doi.org/10.1109/ICSMC.2002.1176401.

[170] D. Tao and X. Tang. "SVM-based relevance feedback using random subspace method". In: *2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No.04TH8763)*. Institute of Electrical & Electronics Engineers (IEEE), 2004. DOI: 10.1109/icme.2004.1394177. URL: http://dx.doi.org/10.1109/ICME.2004.1394177.

[171] D. Tao, X. Tang, X. Li, and X. Wu. "Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.7 (July 2006), pp. 1088–

1099. DOI: 10.1109/tpami.2006.134. URL: http://dx.doi.org/10.1109/TPAMI.2006.134.

[172] A. K. Tiwari and R. Srivastava. "A Survey of Computational Intelligence Techniques in Protein Function Prediction". In: *International Journal of Proteomics* 2014 (2014), pp. 1–22. DOI: 10.1155/2014/845479. URL: http://dx.doi.org/10.1155/2014/845479.

[173] J. Torres-Sospedra, C. Hernandez-Espinosa, and M. Fernandez-Redondo. "Designing a Multilayer Feedforward Ensembles with Cross Validated Boosting Algorithm". In: *The 2006 IEEE International Joint Conference on Neural Network Proceedings*. Institute of Electrical & Electronics Engineers (IEEE), 2006. DOI: 10.1109/ijcnn.2006.246839. URL: http://dx.doi.org/10.1109/IJCNN.2006.246839.

[174] I. Trajkovski, F. Železný, N. Lavrač, and J. Tolar. "Learning Relational Descriptions of Differentially Expressed Gene Groups". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 38.1 (Jan. 2008), pp. 16–25. DOI: 10.1109/tsmcc.2007.906059. URL: http://dx.doi.org/10.1109/TSMCC.2007.906059.

[175] A. Tsymbal, M. Pechenizkiy, and P. Cunningham. "Sequential Genetic Search for Ensemble Feature Selection". In: *Proceedings of the 19th International Joint Conference on Artificial Intelligence*. IJCAI'05. Edinburgh, Scotland: Morgan Kaufmann Publishers Inc., 2005, pp. 877–882. URL: http://dl.acm.org/citation.cfm?id=1642293.1642434.

[176] K. Tumer and J. Ghosh. "Error Correlation and Error Reduction in Ensemble Classifiers". In: *Connection Science* 8.3-4 (Dec. 1996), pp. 385–404. DOI: 10.1080/095400996116839. URL: http://dx.doi.org/10.1080/095400996116839.

[177] G. Valentini, M. Muselli, and F. Ruffino. "Bagged ensembles of Support Vector Machines for gene expression data analysis". In: *Proceedings of the International Joint Conference on Neural Networks, 2003.* IEEE, 2003. DOI: 10.1109/ijcnn.2003.1223688. URL: http://dx.doi.org/10.1109/IJCNN.2003.1223688.

[178] C. Valle, R. Ñanculef, H. Allende, and C. Moraga. "Two Bagging Algorithms with Coupled Learners to Encourage Diversity". In: *Advances in Intelligent Data Analysis VII*. Springer Science Business Media, 2007, pp. 130–139. DOI: 10.1007/978-3-540-74825-0_12. URL: http://dx.doi.org/10.1007/978-3-540-74825-0_12.

[179] T. Vergoulis et al. "TarBase 6.0: capturing the exponential growth of miRNA targets with experimental support". In: *Nucleic Acids Research* 40.D1 (Dec. 2011), pp. D222–D229. DOI: 10.1093/nar/gkr1161. URL: http://dx.doi.org/10.1093/nar/gkr1161.

[180] S. van der Walt, S. C. Colbert, and G. Varoquaux. "The NumPy Array: A Structure for Efficient Numerical Computation". In: *Computing in Science & Engineering* 13.2 (Mar. 2011), pp. 22–30. DOI: 10.1109/mcse.2011.37. URL: http://dx.doi.org/10.1109/MCSE.2011.37.

[181] A. Wang, N. An, Y. Xia, L. Li, and G. Chen. "A Logistic Regression and Artificial Neural Network-Based Approach for Chronic Disease Prediction: A Case Study of Hypertension". In: *2014 IEEE International Conference on Internet of Things(iThings), and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom)*. IEEE, Sept. 2014. DOI: 10.1109/ithings.2014.16. URL: http://dx.doi.org/10.1109/iThings.2014.16.

[182] J. Wang, S. Luo, and X. Zeng. "A random subspace method for co-training". In: *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. Institute of Electrical & Electronics Engineers (IEEE), June 2008. DOI: 10.1109/ijcnn.2008.4633789. URL: http://dx.doi.org/10.1109/IJCNN.2008.4633789.

[183] X. Wang, Z. Shi, C. Wu, and W. Wang. "An Improved Algorithm for Decision-Tree-Based SVM". In: *2006 6th World Congress on Intelligent Control and Automation*. IEEE, 2006. DOI: 10.1109/wcica.2006.1713173. URL: http://dx.doi.org/10.1109/WCICA.2006.1713173.

[184] Z. Wang, W. Xu, F. A. S. Lucas, and Y. Liu. "Incorporating prior knowledge into Gene Network Study". In: *Bioinformatics* 29.20 (Aug. 2013), pp. 2633–2640. DOI: 10.1093/bioinformatics/btt443. URL: http://dx.doi.org/10.1093/bioinformatics/btt443.

[185] G. I. Webb. "MultiBoosting: A Technique for Combining Boosting and Wagging". In: *Machine Learning* 40.2 (2000), pp. 159–196. DOI: 10.1023/a:1007659514849. URL: http://dx.doi.org/10.1023/A:1007659514849.

[186] H. Wei, X. Lin, X. Xu, L. Li, W. Zhang, and X. Wang. "A novel ensemble classifier based on multiple diverse classification methods". In: *2014 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*. IEEE, Aug. 2014. DOI: 10.1109/fskd.2014.6980850. URL: http://dx.doi.org/10.1109/FSKD.2014.6980850.

[187] S. L. Wong et al. "Combining biological networks to predict genetic interactions". In: *Proceedings of the National Academy of Sciences* 101.44 (Oct. 2004), pp. 15682–15687. DOI: 10.1073/pnas.0406614101. URL: http://dx.doi.org/10.1073/pnas.0406614101.

[188] J. Wooldridge. *Introductory econometrics : a modern approach*. Mason, OH: South Western, Cengage Learning, 2009. ISBN: 978-0-324-66054-8.

[189] Q. Wu, Y. Ye, S.-S. Ho, and S. Zhou. "Semi-supervised multi-label collective classification ensemble for functional genomics". In: *BMC Genomics* 15.Suppl 9 (2014), S17. DOI: 10.1186/1471-2164-15-s9-s17. URL: http://dx.doi.org/10.1186/1471-2164-15-S9-S17.

[190] B. Yang, J. Tan, N. Deng, and L. Jing. "Network Kernel SVM for microarray classification and gene sets selection". In: *2012 IEEE 6th International Conference on Systems Biology (ISB)*. IEEE, Aug. 2012. DOI: 10.1109/isb.2012.6314120. URL: http://dx.doi.org/10.1109/ISB.2012.6314120.

[191] H. Yang and S. Fong. "Optimized very fast decision tree with balanced classification accuracy and compact tree size". In: *ICMIA 2011 proceedings : the 3rd International Conference on Data Mining and Intelligent Information Technology Applications*. IEEE, October 2011. ISBN: 978-1-4673-0231-9. URL: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=6108399.

[192] P. Yang, Y. H. Yang, B. B. Zhou, and A. Y. Zomaya. "A Review of Ensemble Methods in Bioinformatics". In: *CBIO* 5.4 (Dec. 2010), pp. 296–308. DOI: 10.2174/157489310794072508. URL: http://dx.doi.org/10.2174/157489310794072508.

[193] Z. Yang and M. Zhou. "Kappa statistic for clustered physician–patients polytomous data". In: *Computational Statistics & Data Analysis* 87 (July 2015), pp. 1–17. DOI: 10.1016/j.csda.2015.01.007. URL: http://dx.doi.org/10.1016/j.csda.2015.01.007.

[194] Z. Yang and M. Zhou. "Weighted kappa statistic for clustered matched-pair ordinal data". In: *Computational Statistics & Data Analysis* 82 (Feb. 2015), pp. 1–18. DOI: 10.1016/j.csda.2014.08.004. URL: http://dx.doi.org/10.1016/j.csda.2014.08.004.

[195] R. Ye and P. Suganthan. "Empirical comparison of bagging-based ensemble classifiers". In: *Information Fusion (FUSION), 2012 15th International Conference on*. July 2012, pp. 917–924. ISBN: 978-1-4673-0417-7.

[196] P. Yildirim and D. Birant. "Naive Bayes classifier for continuous variables using novel method (NBC4D) and distributions". In: *2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings*. IEEE, June 2014. DOI: 10.1109/inista.2014.6873605. URL: http://dx.doi.org/10.1109/INISTA.2014.6873605.

[197] H. Yu and S. Xu. "Simple rule-based ensemble classifiers for cancer DNA microarray data classification". In: *2011 International Conference on Computer Science and Service System (CSSS)*. IEEE, June 2011. DOI: 10.1109/csss.2011.5974135. URL: http://dx.doi.org/10.1109/CSSS.2011.5974135.

[198] J. Yu, J. Yu, A. A. Almal, S. M. Dhanasekaran, D. Ghosh, W. P. Worzel, and A. M. Chinnaiyan. "Feature Selection and Molecular Classification of Cancer Using Genetic Programming". In: *Neoplasia* 9.4 (Apr. 2007), pp. 292–303. DOI: 10.1593/neo.07121. URL: http://dx.doi.org/10.1593/neo.07121.

[199] J. Yu, J. Yu, A. A. Almal, S. M. Dhanasekaran, D. Ghosh, W. P. Worzel, and A. M. Chinnaiyan. "Feature Selection and Molecular Classification of Cancer Using Genetic Programming". In: *Computational and Mathematical Methods in Medicine* 2015 (Apr. 2015), pp. 1–11. DOI: 10.1155/2015/193406. URL: http://dx.doi.org/10.1155/2015/193406.

[200] W. Yu, M. Clyne, M. J. Khoury, and M. Gwinn. "Phenopedia and Genopedia: disease-centered and gene-centered views of the evolving knowledge of human genetic associations". In: *Bioinformatics* 26.1 (Oct. 2009), pp. 145–146. DOI: 10.1093/bioinformatics/btp618. URL: http://dx.doi.org/10.1093/bioinformatics/btp618.

[201] G. U. Yule. "On the Association of Attributes in Statistics: With Illustrations from the Material of the Childhood Society, &c". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 194.252-261 (Jan. 1900), pp. 257–319. DOI: 10.1098/rsta.1900.0019. URL: http://dx.doi.org/10.1098/rsta.1900.0019.

[202] Y. Zhang, B. Zhang, F. Coenenz, and W. Lu. "Highly reliable breast cancer diagnosis with cascaded ensemble classifiers". In: *The 2012 International Joint Conference on Neural Networks (IJCNN)*. IEEE, June 2012. DOI: 10.1109/ijcnn.2012.6252547. URL: http://dx.doi.org/10.1109/IJCNN.2012.6252547.

[203] W. Zhi and L. Hongzhe. "A Markov random field model for network-based analysis of genomic data". In: *Bioinformatics* 23.12 (May 2007), pp. 1537–1544. DOI: 10.1093/bioinformatics/btm129. URL: http://dx.doi.org/10.1093/bioinformatics/btm129.

[204] W. Zhi, J. Minturn, E. Rappaport, G. Brodeur, and H. Li. "Network-Based Analysis of Multivariate Gene Expression Data". In: *Methods in Molecular Biology*. Springer Science Business Media, 2013, pp. 121–139. DOI: 10.1007/978-1-60327-337-4_8. URL: http://dx.doi.org/10.1007/978-1-60327-337-4_8.

[205] X. Zhou, X. Y. Wu, K. Z. Mao, and D. P. Tuck. "Fast Gene Selection for Microarray Data Using SVM-Based Evaluation Criterion". In: *2008 IEEE International Conference on Bioinformatics and Biomedicine*. IEEE, 2008. DOI: 10.1109/bibm.2008.57. URL: http://dx.doi.org/10.1109/BIBM.2008.57.

[206] X. Zhou, J. Liu, X. Ye, W. Wang, and J. Xiong. "Ensemble classifier based on context specific miRNA regulation modules: a new method for cancer outcome prediction". In: *BMC Bioinformatics* 14.Suppl 12 (2013), S6. DOI: 10.1186/1471-2105-14-s12-s6. URL: http://dx.doi.org/10.1186/1471-2105-14-S12-S6.

[207] Y. Zhu, X. Shen, and W. Pan. "Network-based support vector machine for classification of microarray samples". In: *BMC Bioinformatics* 10.Suppl 1 (2009), S21. DOI: 10.1186/1471-2105-10-s1-s21. URL: http://dx.doi.org/10.1186/1471-2105-10-S1-S21.

[208] Y. Zhu. "Random subspace method based on Canonical Correlation Analysis". In: *2010 3rd International Congress on Image and Signal Processing*. Institute of Electrical & Electronics Engineers (IEEE), Oct. 2010. DOI: 10.1109/cisp.2010.5648017. URL: http://dx.doi.org/10.1109/CISP.2010.5648017.

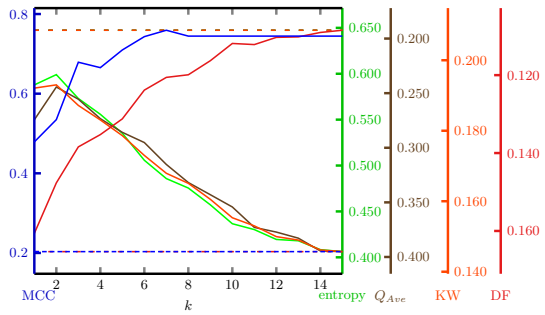
Appendices

Appendix A

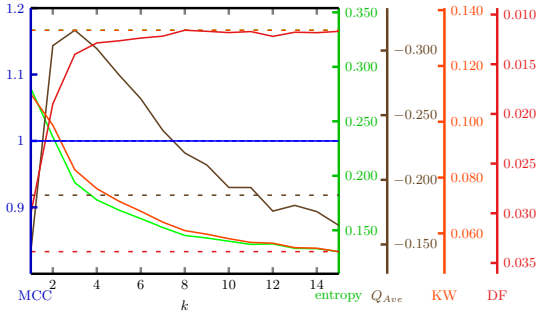
MDS experiment plots

All these plots depict MCC and several diversity measures, however, even though they are depicted for discrete values of k, they are depicted by a line without markers because the graphs would become even less readable that way. Moreover, all diversity measures are drawn in a such way that if the line that represents them is decreasing, the diversity is also decreasing, which results in reversed axes for several measures but it enhance the readability of the plot. Also, axes are not in the same scale in order to show the relationship in change between various diversity measures and the accuracy.

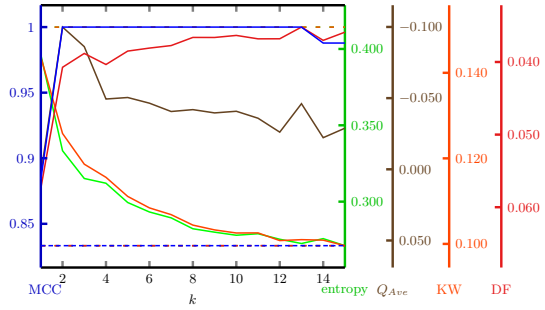
A.1 NCSR with Decision Trees



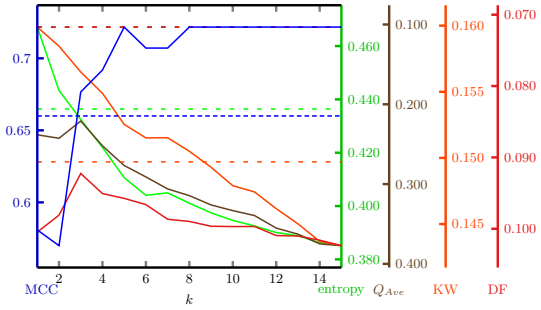
(a) BMBT_DT5q



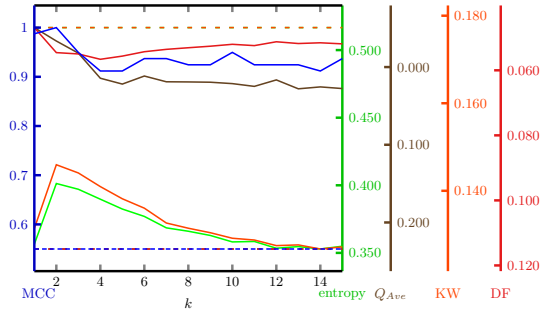
(b) BMH_ABT5q



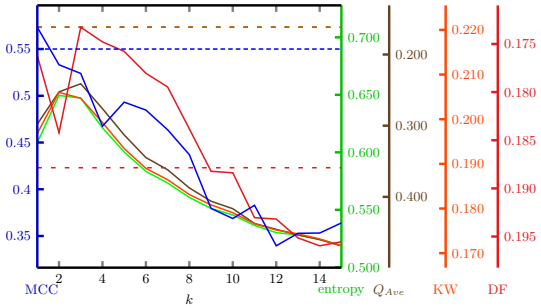
(c) BMH_ABTnon-5q



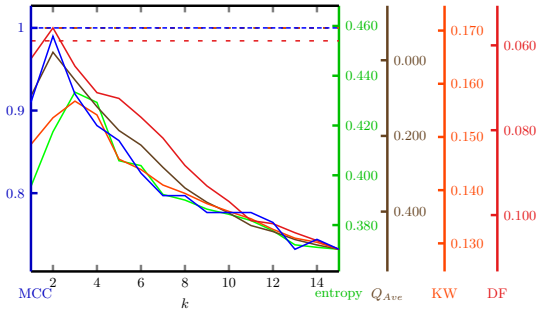
(d) BMH_ADT5q



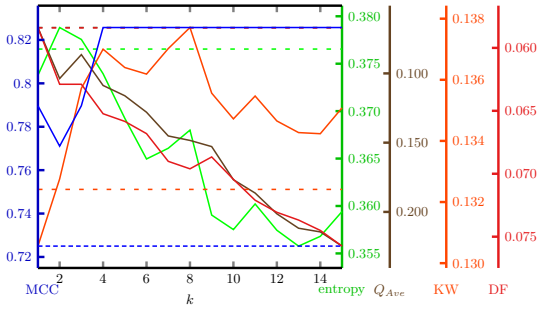
(e) BMnon-5q_5qBT



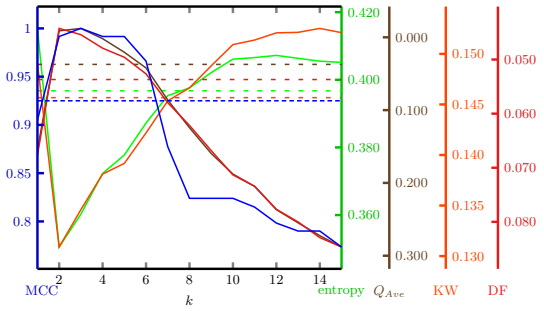
(f) PBBT_DT5q



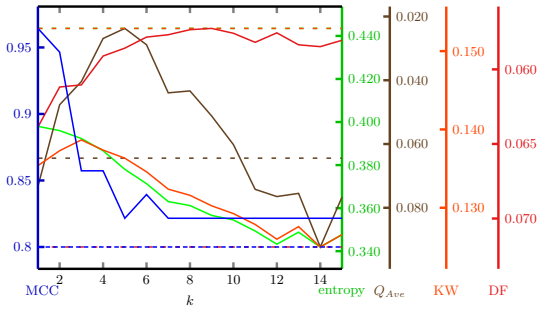
(g) PBH_ABT5q



(h) PBH_ABTnon-5q

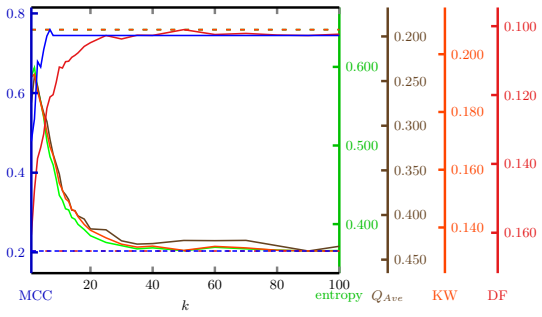


(i) PBH_ADT5q

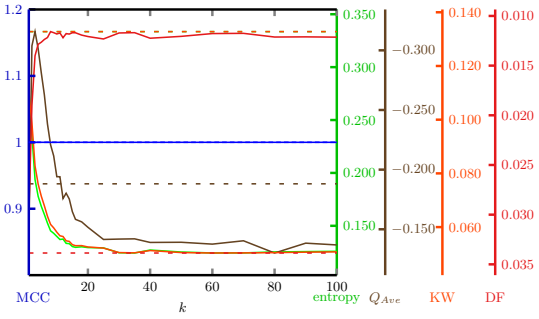


(j) PBnon-5q_5qBT

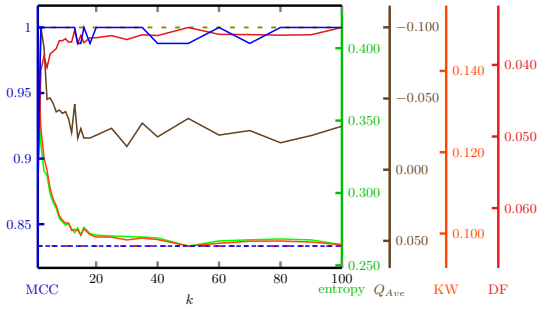
Figure A.1: Relationships between Matthew's correlation coefficient and 4 diversity measures - entropy, average Q statistics, Kohawi-Wolpert, and Double Fault measure — for NCRS with Decision trees. The dotted line represents values obtained from Random Subspace method with Decision trees. Computed for $k \in \{1, 2, \ldots, 14, 15\}$.



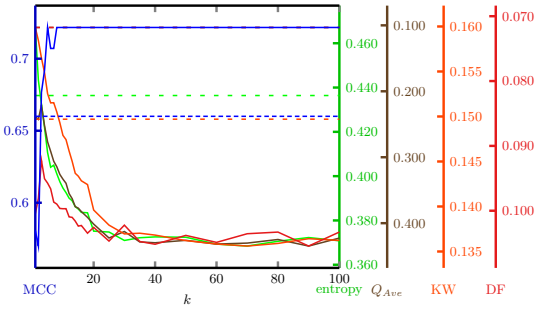
(a) BMBT_DT5q



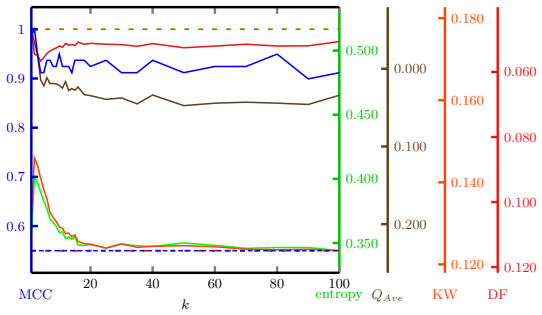
(b) BMH_ABT5q



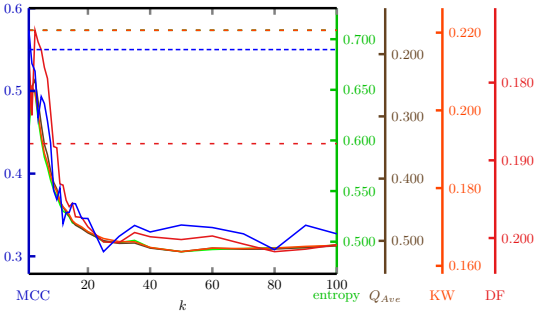
(c) BMH_ABTnon-5q



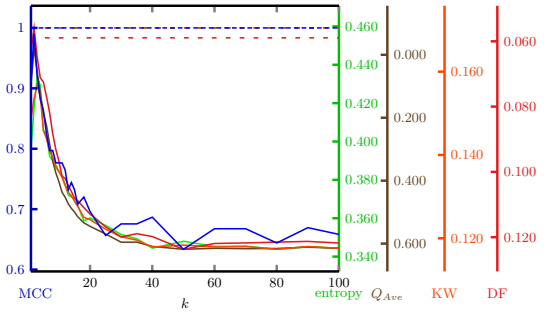
(d) BMH_ADT5q



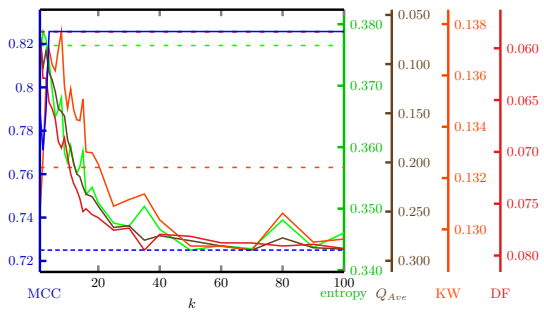
(e) BMnon-5q_5qBT



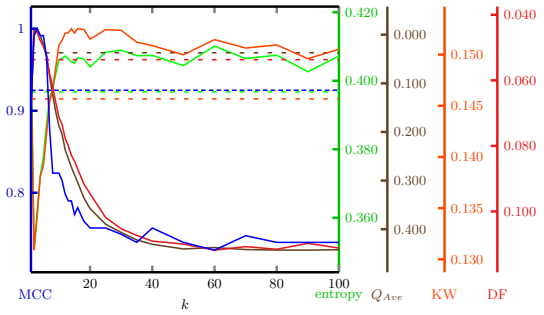
(f) PBBT_DT5q



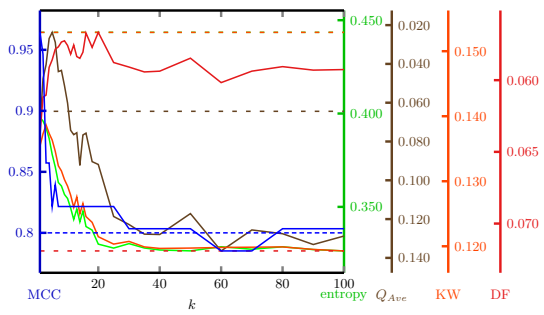
(g) PBH_ABT5q



(h) PBH_ABTnon-5q



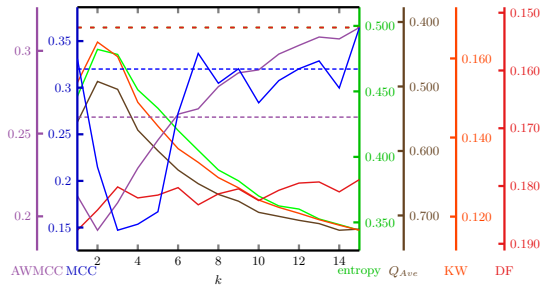
(i) PBH_ADT5q



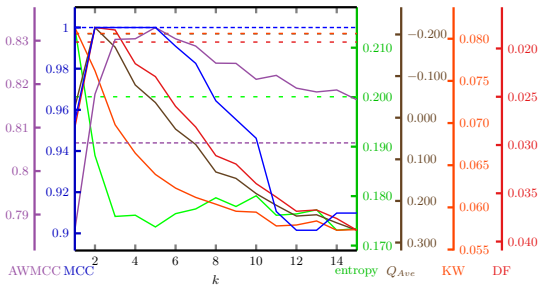
(j) PBnon-5q_5qBT

Figure A.2: Relationships between Matthew's correlation coefficient and 4 diversity measures - entropy, average Q statistics, Kohawi-Wolpert, and Double Fault measure — for NCRS with Decision trees. The dotted line represents values obtained from Random Subspace method with Decision trees. Computed for $k \in \{1, 2, \ldots, 15, 16, 18, 20, 25, 30, 35, 40, 50, 60, 70, 80, 90, 100\}$.

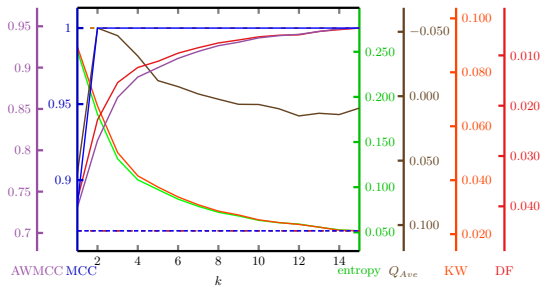
A.2 NCSR with Logistic Regression



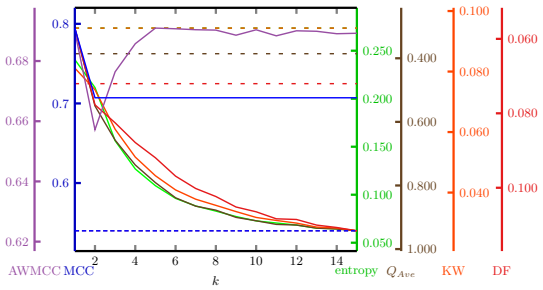
(a) BMBT_DT5q



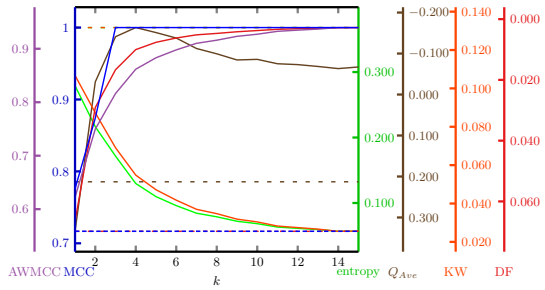
(b) BMH_ABT5q



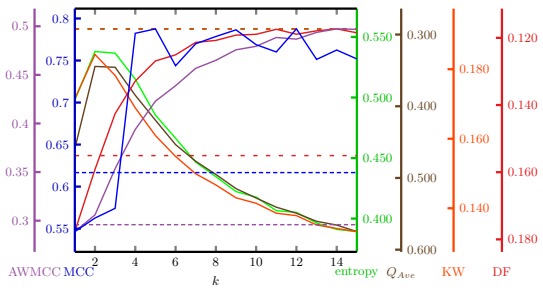
(c) BMH_ABTnon-5q



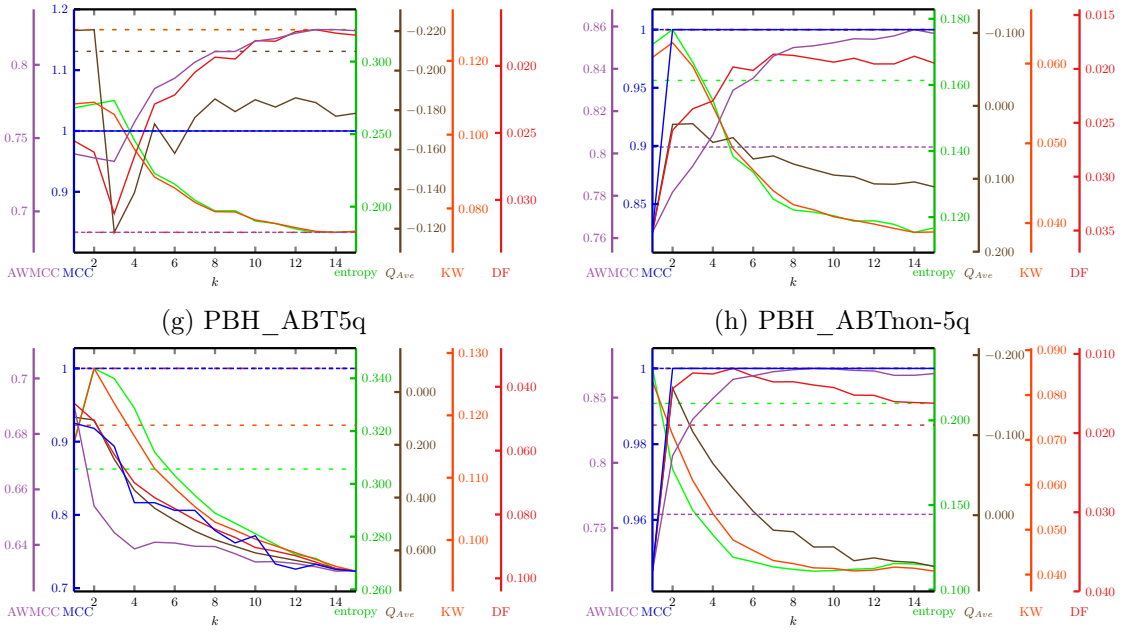
(d) BMH_ADT5q



(e) BMnon-5q_5qBT



(f) PBBT_DT5q

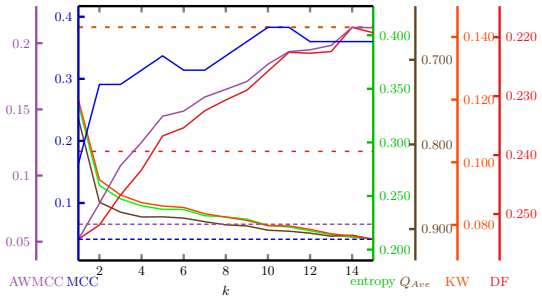


(g) PBH_ABT5q (h) PBH_ABTnon-5q

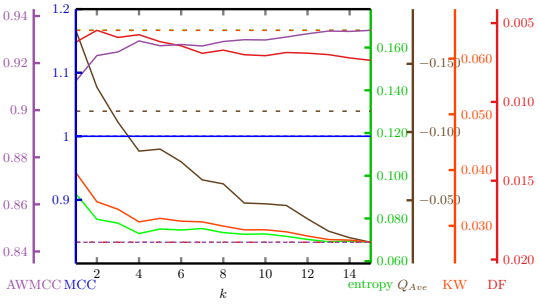
(i) PBH_ADT5q (j) PBnon-5q_5qBT

Figure A.3: Relationships between Matthew's correlation coefficient and 4 diversity measures - entropy, average Q statistics, Kohawi-Wolpert, and Double Fault measure — for NCRS with Logistic regression weak classifiers. The dotted line represents values obtained from Random Subspace method with Logistic regression weak classifiers. Computed for $k \in \{1, 2, \ldots, 14, 15\}$.

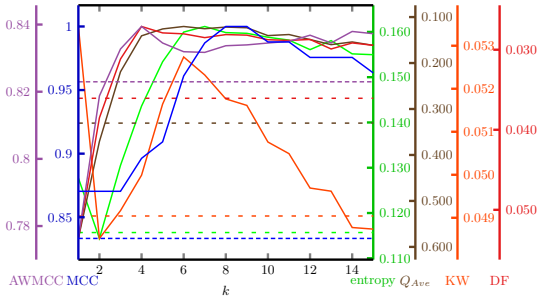
A.3 NCSR with Naïve Bayes



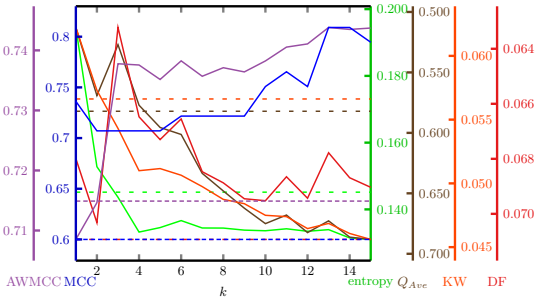
(a) BMBT_DT5q



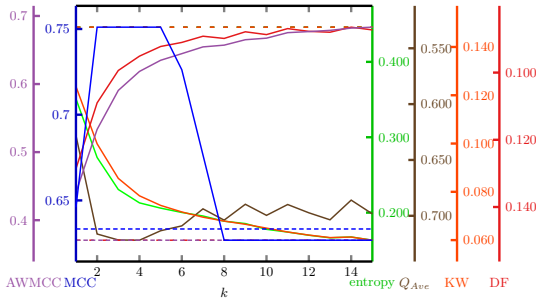
(b) BMH_ABT5q



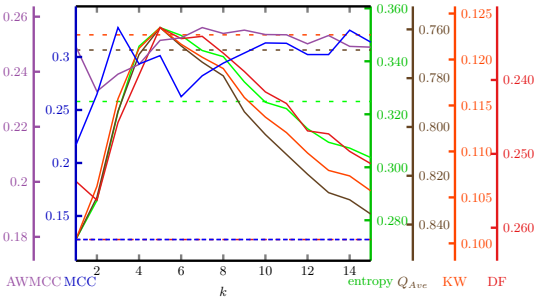
(c) BMH_ABTnon-5q



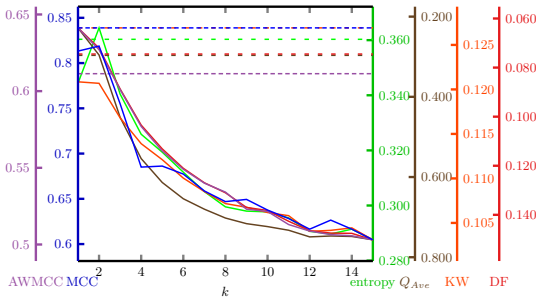
(d) BMH_ADT5q



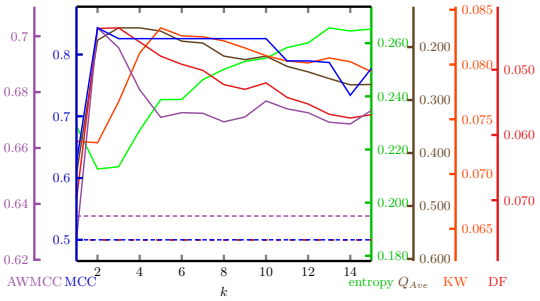
(e) BMnon-5q_5qBT



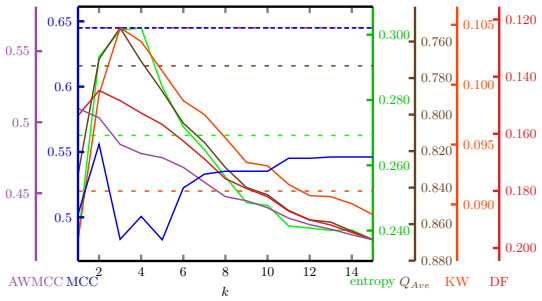
(f) PBBT_DT5q



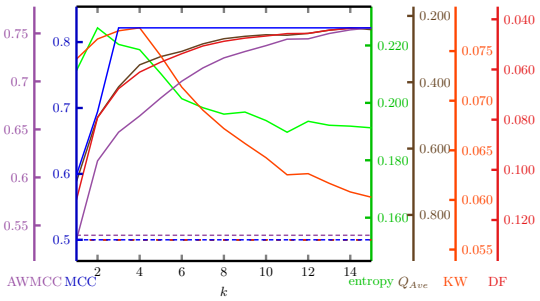
(g) PBH_ABT5q



(h) PBH_ABTnon-5q



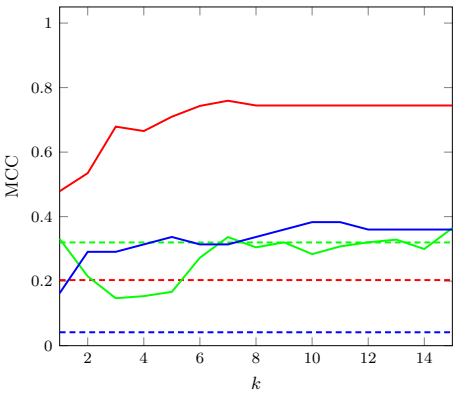
(i) PBH_ADT5q



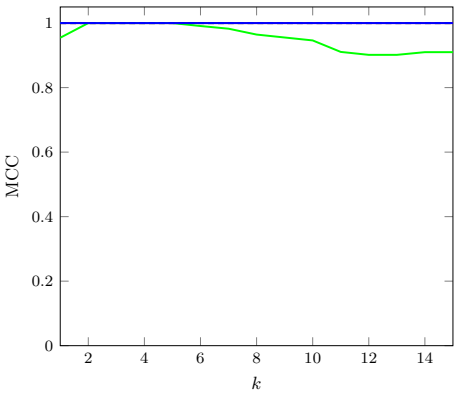
(j) PBnon-5q_5qBT

Figure A.4: Relationships between Matthew's correlation coefficient and 4 diversity measures - entropy, average Q statistics, Kohawi-Wolpert, and Double Fault measure — for NCRS with Naïve Bayes weak classifiers. The dotted line represents values obtained from Random Subspace method with Naïve Bayes weak classifiers. Computed for $k \in \{1, 2, \ldots, 14, 15\}$.

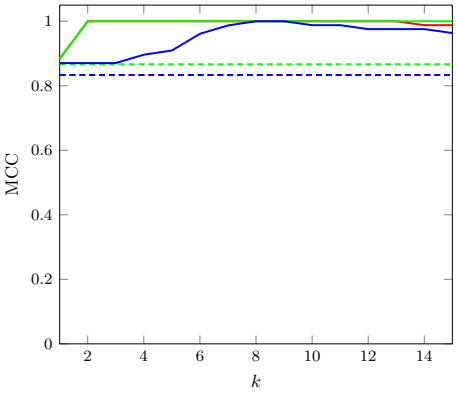
A.4 Comparison of different types of weak classifiers



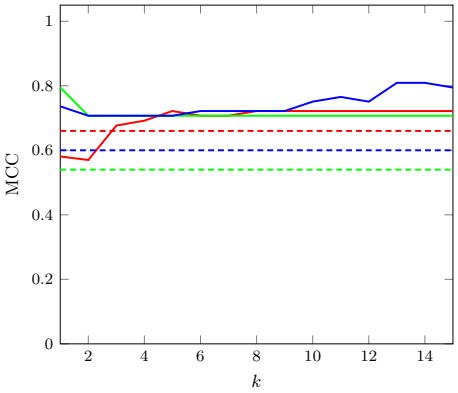
(a) BMBT_DT5q



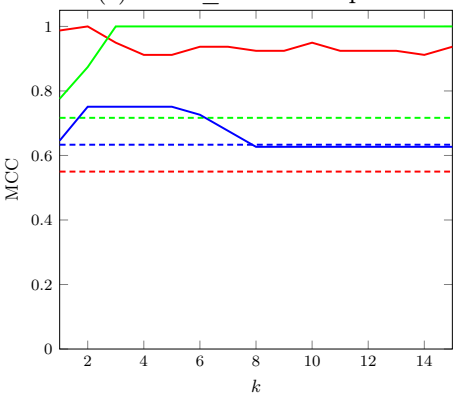
(b) BMH_ABT5q



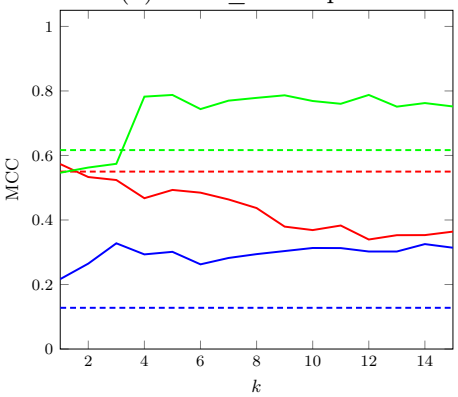
(c) BMH_ABTnon-5q



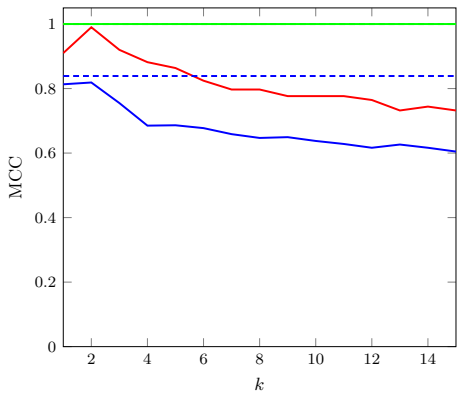
(d) BMH_ADT5q



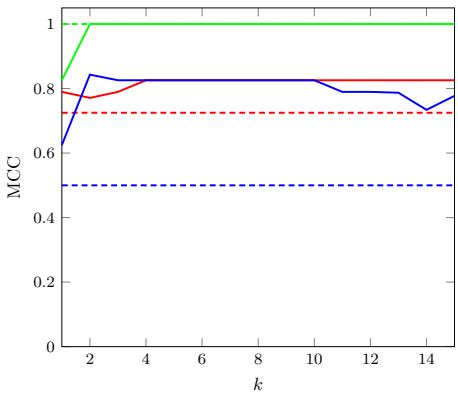
(e) BMnon-5q_5qBT



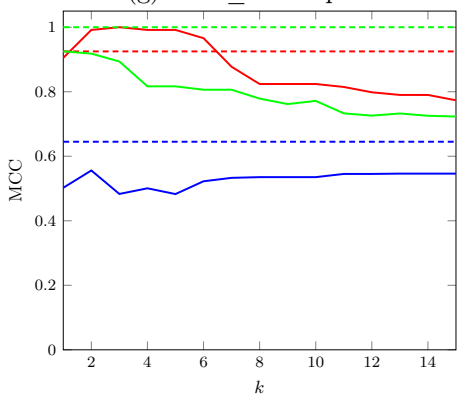
(f) PBBT_DT5q



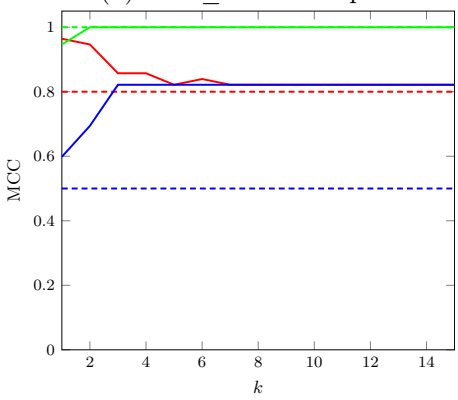
(g) PBH_ABT5q



(h) PBH_ABTnon-5q



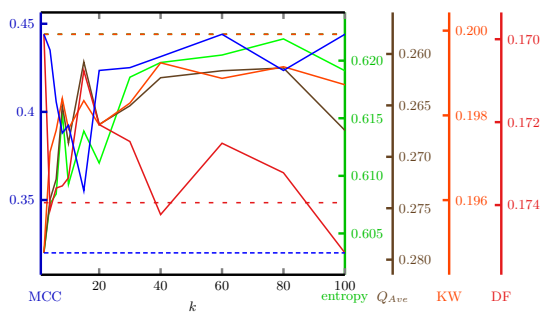
(i) PBH_ADT5q



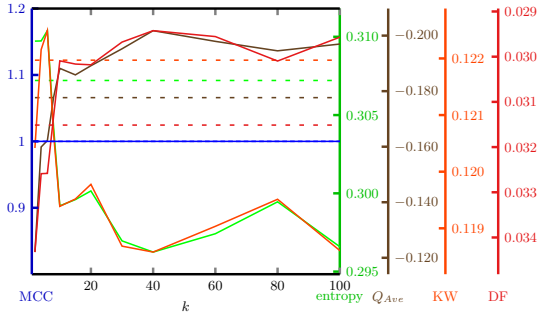
(j) PBnon-5q_5qBT

Figure A.5: Comparison of MCC performance of NCRS with three different types of weak classifiers. Values taken from Figure A.1,Figure A.3, and Figure A.4 (i.e., with Decision trees (red), Logistic Regression (green) and Naïve Nayes (blue) weak classifiers).

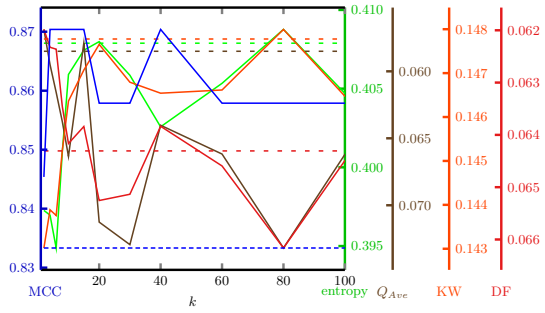
A.5 NCSR without miRNAs prior knowledge



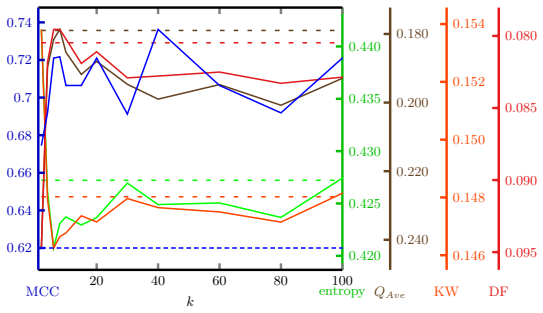
(a) BMBT_DT5q



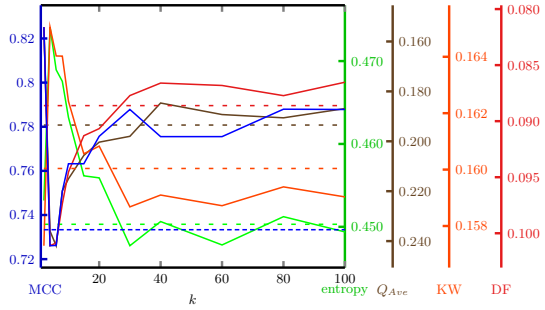
(b) BMH_ABT5q



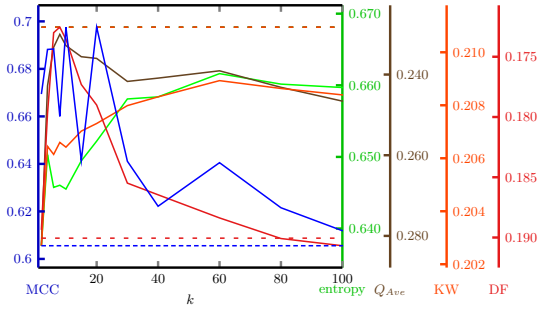
(c) BMH_ABTnon-5q



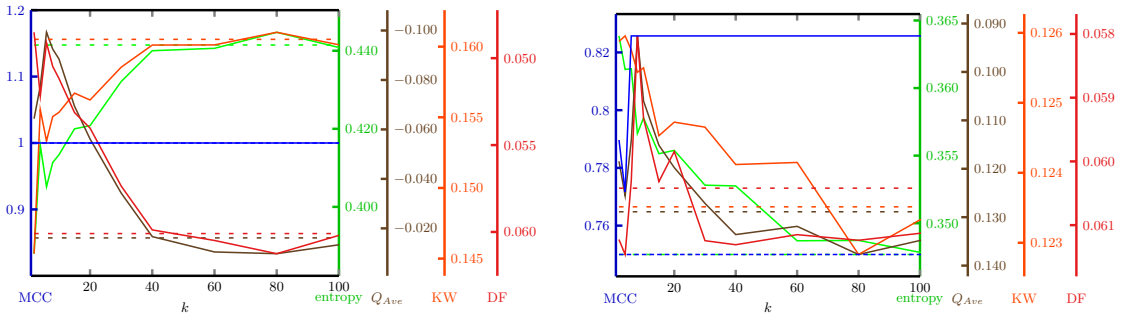
(d) BMH_ADT5q



(e) BMnon-5q_5qBT

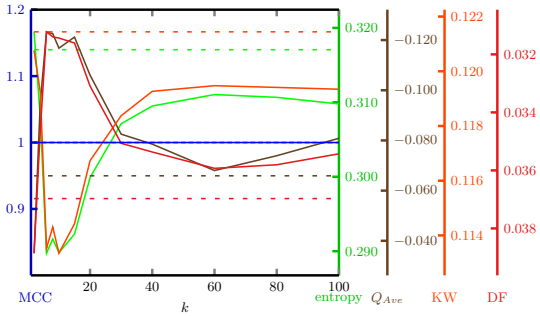


(f) PBBT_DT5q

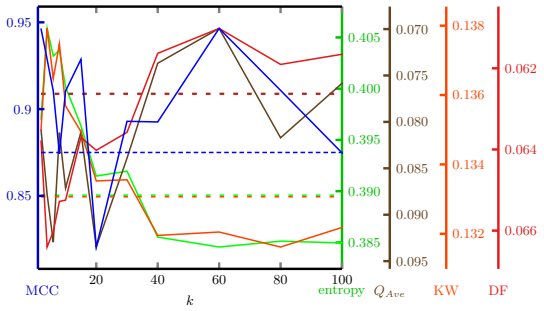


(g) PBH_ABT5q

(h) PBH_ABTnon-5q



(i) PBH_ADT5q



(j) PBnon-5q_5qBT

Figure A.6: Relationships between Matthew's correlation coefficient and 4 diversity measures - entropy, average Q statistics, Kohawi-Wolpert, and Double Fault measure — for NCRS with Decision trees and without miRNAs prior knowledge. The dotted line represents values obtained from NCRS with modified sampling function — samples features with probability proportional to their degree in the feature network. Computed for $k \in \{2, 4, 6, 8, 10, 15, 20, 30, 40, 60, 80, 100, 150, 200\}$.

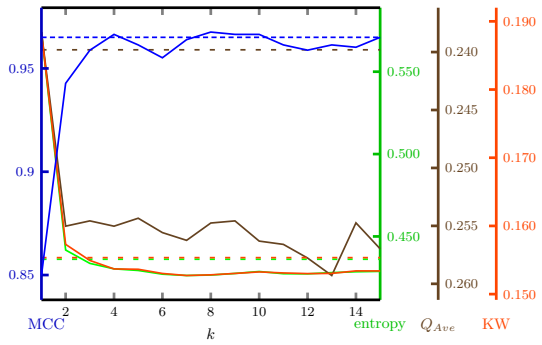
Appendix B

Benchmark datasets

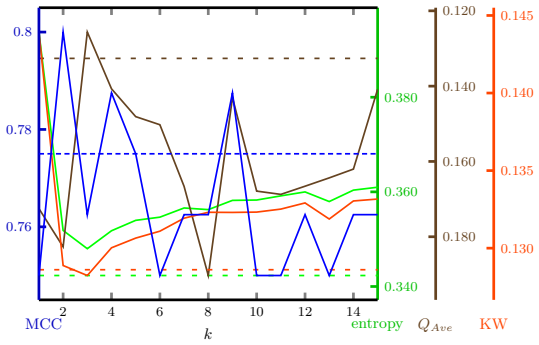
These are results for several benchmark datasets for *gene expression* profiles. Unlike the MDS datasets, they do not contain miRNA and the NCRS was trained without miRNA prior knowledge or candidate causal genes. Table B.1 contains the information about platform type and number of samples for each dataset, more about datasets is available in Section 7.2 Benchmark datasets.

Table B.1: Number of samples in individual datasets

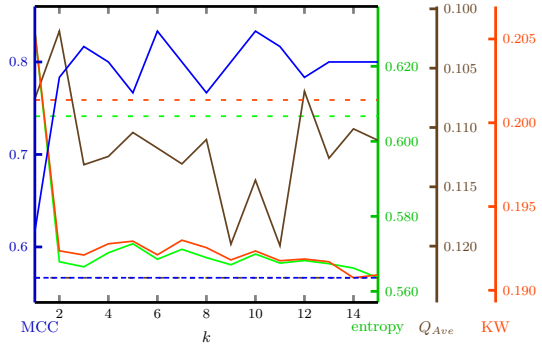
Dataset	Platform	Number of samples
ALL/AML	GPL80	72
Gastric cancer	GPL80	30
Hypertension	GPL80	20
Smoking	GPL80	44
AML	GPL96	64
Breast cancer	GPL96	29
Glioma	GPL96	85
MGCT	GPL96	27
Prostate cancer	GPL96	20
Sarcoma/Hypoxia	GPL96	54



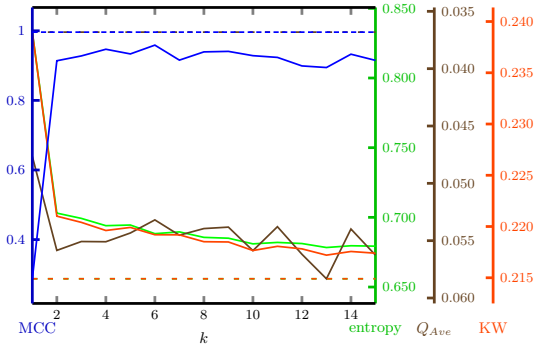
(a) ALL/AML



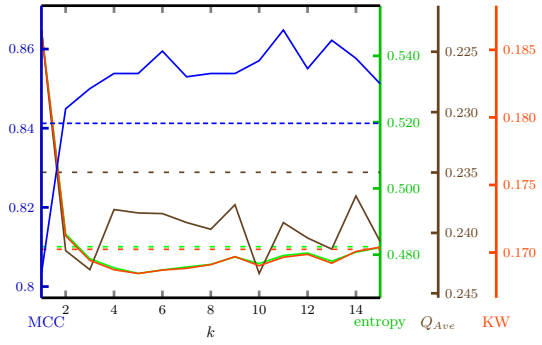
(b) Gastric cancer



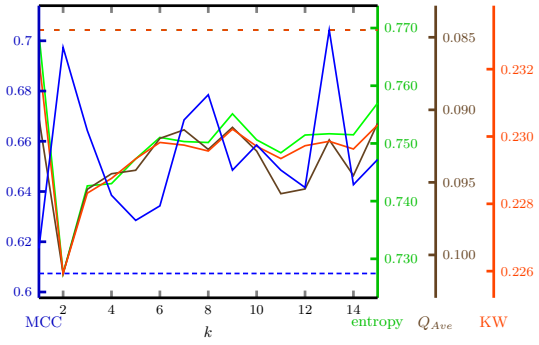
(c) Hypertension



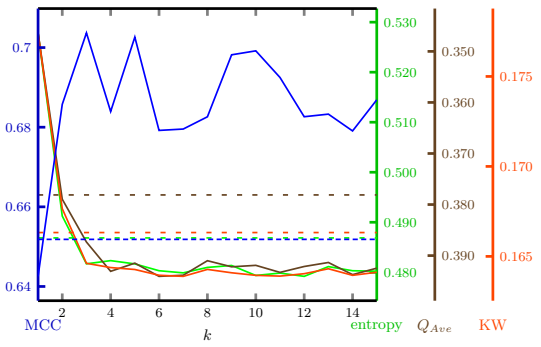
(d) Smoking



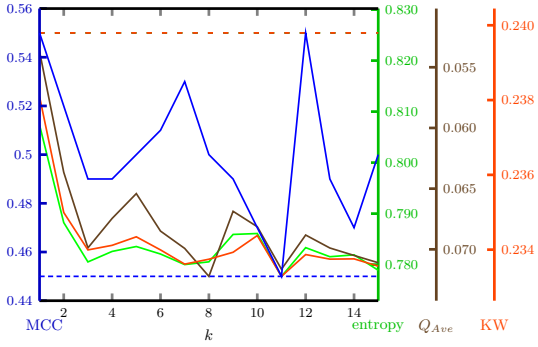
(e) AML



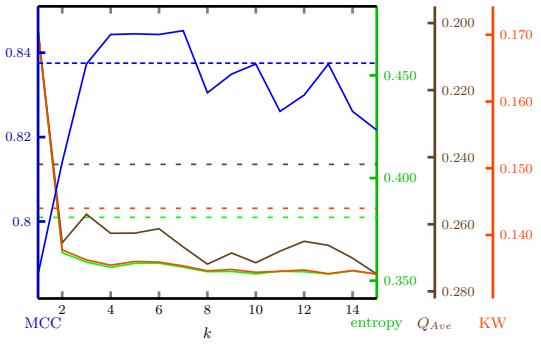
(f) Breast cancer



(g) Glioma



(h) Prostate cancer



(i) Sarcoma/Hypoxia

Figure B.1: Relationships between Matthew's correlation coefficient and 4 diversity measures — entropy, average Q statistics, Kohawi-Wolpert — for NCRS with decision trees compared to RS DT

Appendix C

Content of the CD

The attached CD contains the electronic version of this work, all graphs from this work, implementation of the NCRS and original datasets.

- `/Codes/` contains codes for both the implemented NCRS and the original NCF
  - `/Codes/NCF/` contains the code of the original NCF proposed by Anděl and Kléma
  - `/Codes/NCRS/` contains the implementation of NCRS together with several scripts that were used for the experiments
    - `/Codes/NCRS/example.py` contains simple example of usage of the NCRS general classifier
    - `/Codes/NCRS/meas.py` contains implementation of diversity metrics used in Chapter 7
    - `/Codes/NCRS/ncrs.py` the actual implementation of the NCRS ensemble. The implementation is based on the code of NCF and BaggingClassifier from [138]
- `/Data/` contains datasets for experiments
  - `/Data/Benchmark/` contains 10 benchmark datasets
  - `/Data/MDS/` contains the data related to the myelodysplastic syndrome (MDS)
    - `/Data/MDS/datasets/` contains the actual datasets together with candidate causal genes
    - `/Data/MDS/interactions/` contains the prior knowledge in the form of interaction networks
- `/Graphs/` contains all graphs from this work
- `/Raw_Results/` contains raw results returned by scripts in the form of log files
- `/kuncvlad_BP_2015.pdf` the electronic version of this work