Informatics in Medicine Unlocked 13 (2018) 176–185

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: www.elsevier.com/locate/imu

Breast cancer image classification using pattern-based Hyper Conceptual Sampling method

Tooba Salahuddin, Fatima Haouari, Fahad Islam, Rahma Ali, Sara Al-Rasbi, Nada Aboueata, Eman Rezk, Ali Jaoua*

Department of Computer Science and Engineering, Qatar University, Doha, Qatar

ARTICLE INFO

Keywords:
Data sampling
Formal concept analysis
Breast cancer
Machine learning

ABSTRACT

The increase in biomedical data has given rise to the need for developing data sampling techniques. With the emergence of big data and the rise of popularity of data science, sampling or reduction techniques have been assistive to significantly hasten the data analytics process. Intuitively, without sampling techniques, it would be difficult to efficiently extract useful patterns from a large dataset. However, by using sampling techniques, data analysis can effectively be performed on huge datasets, to produce a relatively small portion of data, which extracts the most representative objects from the original dataset. However, to reach effective conclusions and predictions, the samples should preserve the data behavior. In this paper, we propose a unique data sampling technique which exploits the notion of formal concept analysis. Machine learning experiments are performed on the resulting sample to evaluate quality, and the performance of our method is compared with another sampling technique proposed in the literature. The results demonstrate the effectiveness and competitiveness of the proposed approach in terms of sample size and quality, as determined by accuracy and the F1-measure.

1. Introduction

Breast cancer is a highly prevalent disease among women and the most commonly associated cause of female mortality. The American Cancer Society has predicted that about 266,000 women are likely to be diagnosed with breast cancer in 2018 in the United States and 15% of them are expected to die of the disease [31]. The survival rate of breast cancer patients can be increased by effective treatment, which can commence upon early diagnosis of the disease [15]. One automated way to diagnose breast cancer is by analyzing the processed data from histology images [3]. Data from each pixel can be classified as malignant or benign, which can assist to detect and determine the cancerous regions. However, due to the massive amount of data, the pixel classification process can become quite time consuming in time critical situations. While numerous studies can be found that concentrate on improving classification accuracy by optimizing training models [1,12,14,23,34,38,39], little attention has been directed towards optimizing the experimental setting by training a smaller sample of data instead of training the entire massive dataset. Fig. 1 demonstrates the pixel classification process by employing a sample for training.

Data sampling is a statistical approach which can be employed to select, manipulate, and analyze a representative subset of data to extract meaningful inferences. Sampling allows the user to work on a small amount of data so as to build and run analytical models relatively faster while preserving data behavior and achieving accurate results simultaneously. However, in this context, it should be noted that the preservation of data behavior is of paramount importance and it is a challenging task. By using the terminology of preserving data behavior, we mean that the sample should also contain the functional dependencies formed in the data. Likewise, an effective sampling technique will produce a high-quality sample that preserves the characteristics of the original dataset. Therefore, the technique of data sampling can act as a catalyst for the process of pixel classification.

Inspired by the research conducted in Ref. [27], this paper introduces an enhanced data sampling technique that blends existing conceptual and mathematical methods. These methods include formal concept analysis, a hyper concept algorithm [19], and a data reduction algorithm based upon high coupling to produce a sample. The novelty of this sample is that it maintains the pattern behavior of the original data and is composed of the most representative data from the original dataset. Further details of our proposed method are described in the Methodology section.

The main contributions of this paper are as follows:

* Corresponding author.
E-mail address: jaoua@qu.edu.qa (A. Jaoua).

https://doi.org/10.1016/j.imu.2018.07.002
Received 20 May 2018; Received in revised form 8 July 2018; Accepted 8 July 2018
Available online 10 July 2018
2352-9148/

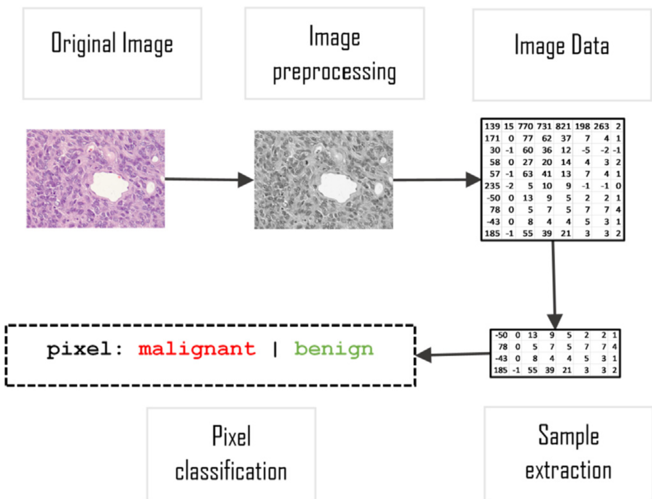

Fig. 1. From raw image to pixel classification.

1. Sampling method: We propose a novel data sampling model, Pattern-based Hyper Conceptual Sampling, which utilizes a unique combination of existing techniques. Our primary objective is to produce a sample that exhibits the most important data characteristics of the original data. We develop a formal concept analysis, a hyper context feature extraction algorithm, and reduction based on high coupling between objects.
2. Machine Learning: We perform an evaluation of the results obtained from our proposed sampling method using machine learning techniques. We also compare our experimental results with other techniques to prove the competitiveness of our sampling model.

The remainder of this paper is organized as follows: Section 3 provides succinct information on the required concepts including formal concept analysis, hyper concept and reduction using high coupling. Section 2 discusses related work on image sampling. Section 4 contains a description of our proposed solution. Experimental results from the proposed sampling technique are presented in Section 5 along with a comparison with another sampling method. Section 6 concludes the paper.

2. Related work

The most commonly applied sampling methods in statistics are simple random sampling and stratified sampling. Simple random sampling is based on arbitrary selection of items, without requiring the fulfillment of any criterion. Contrarily, stratified sampling categorizes the population into strata, based on some criteria, followed by the proportional selection of items from each stratum [5]. The categorization into strata should be such that they satisfy the requirement of being mutually exclusive and collectively exhaustive. Albeit their apparent ease of application, samples generated from this method fail to preserve the functional dependencies present in the data which is necessary to create a representative subset of the dataset. It also requires prior knowledge of the data to be able to categorize into strata. In fact, it is possible to derive meaningful insights from the sampled data and focus on the important aspects of the dataset by applying nonrandom sampling techniques [29].

Additionally, diverse image data sampling techniques are available in the literature, mainly for image rendering. Kettunen et al. [20] introduced gradient-domain sampling for image synthesis. This algorithm considers sampling based upon image gradients and pixels. Estimated gradients are calculated by computing the difference between similar path pairs, resulting in lower variance. Eventually, information from pixel values and gradients is combined using Poisson reconstruction, thus producing a sample. Along similar lines, Cho et al. [13] implemented optimized diffusion gradient sampling, also called b-value sampling, for improved analysis of breast lesions. To accelerate dynamic volumetric MRI, Feng et al. [16] used golden-angle radial sampling. Eventually, Ayech and Ziou [8] proposed an enhanced design of ranked set sampling through k-means clustering. First, optimal centers are estimated by a function on ranked set sample, and then the observed dataset is classified according to the estimated centers.

However, to the best of our knowledge, limited sampling methods for images exist in the literature which fulfill our desired criteria in a representative sample. Rezk et al. [27] proposed Pattern-based Proportional Sampling which requires the data to traverse through four different transformations to extract a sample. Initially, the processed image data is converted to binary formal context, using the pairwise tuple comparison method. In the second stage, the resulting binary relation is used for extracting patterns from the data. Patterns are calculated from the number of features in the dataset. In the third stage, proportions are mathematically calculated from the patterns and tuple instances are selected based upon the proportions. Finally, in the last stage, the objects are mapped to the original data in order to extract instances resulting in the final sample. In this way, the produced sample preserves the behavior of the original data. Yu Su et al. [32,33] also developed algorithms for efficient and effective sampling by applying indexing techniques on scientific datasets containing simulation data. The resulting samples preserve data behavior by maintaining value and spatial distributions across the sample. Firstly, bitvectors are subdivided into equal sized sectors. Next, a certain number of samples are extracted from each bitvector using random stratified sampling. This technique helps maintain value distribution across the sample. The percentage of samples is constant across the sectors. A controlled random sampling technique was proposed by Liang et al. [21] for selecting pixel samples from hyperspectral images that facilitates minimal overlap between training and testing sets from the same image. The sample possesses the property of being globally and randomly scattered across the image and well-proportioned across the classes.

3. Background

3.1. Functional dependencies

In its classical definition, functional dependency is a relationship that exists when one attribute uniquely determines another attribute. If R is a relation with attributes X and Y, a functional dependency between attributes is represented by $\mathscr{X} \rightarrow \mathscr{Y}$, which indicates Y is functionally dependent on X [7]. Suppose we have a Patient table with attributes patient_Id, patient_Name and patient_Age. In this example, patient_Id attribute uniquely identifies the patient_Name attribute, which means patient_Name is functionally dependent on patient_Id.

3.2. Formal concept analysis

Formal Concept Analysis (FCA) provides a foundation for building a conceptual mathematical framework motivated by the lattice theory [9,35]. This discipline allows a conceptual yet meaningful approach to knowledge discovery and representation [24] for practical applications in diverse domains. These domains include, but are not limited to, image mining [11], medical image interpretation [6], decision making [36], semantic web search [18], sentiment analysis [22,26], feature extraction [17] and data reduction [28].

The key expression in FCA is a formal context, which is a binary mapping between objects and attributes [25]. The transformation from the input data into binary formal concept is done in two main steps. In the first step, the input data is transformed into a binary data table which is called formal context. According to the standard definition, a formal context is represented by (G, M, I), where G and M are finite sets of objects and attributes respectively, and I is a binary relation between

Table 1
Formal context.

	Breast cancer	Malignant	Benign	Chemo Therapy	Radio Therapy	Operation
Patient 1	1	0	1	0	0	0
Patient 2	**1**	**1**	0	0	0	0
Patient 3	0	1	0	1	1	0
Patient 4	1	0	1	1	0	1
Patient 5	0	0	1	0	1	0
Patient 6	**1**	**1**	0	1	0	0

G and M. Table 1 represents a formal context where G = {Patient 1, Patient 2, Patient 3, Patient 4, Patient 5, Patient 6} and M = {Breast cancer, Malignant, Benign, Chemo Therapy, Radio Therapy, Operation} and I is the binary relation. Then in the second step, the formal concept (A, B) is built upon this context, where A and B define the extent and intent of the formal concept respectively. (A, B) = {{Patient 2, Patient 6}, {Breast cancer, Malignant}} is an example of a formal concept that can be generated from the formal context represented in Table 1, where B represents all of the attributes in M shared by the objects in A.

3.3. Hyper concept

Hyper concept, also called Hyper Rectangle, is based upon formal concept analysis [19]. Let (G, M, R) be a formal context and a is an arbitrary attribute where a ∈ M. The hyper rectangle (H_a) is a sub-relation of R such that $H_a(R) = \mathcal{I}(a.\ R^{-1})oR$ ($\mathcal{I}$ is the identity relation, $\mathcal{I}(R) = \{(e, e)|e \in R\}$, e being a tuple in R). We calculate a weight for each hyper rectangle which gives us a measure of its strength in terms of the association between its objects and attributes. The weight, w, of a hyper rectangle $H_a(R)$ is calculated by:

$$w(H_a(R)) = \frac{r}{(d^*c)}{}^*(r - (d + c)),$$

where r is the cardinality of $H_a(R)$ (i.e. the number of pairs in the binary relation $H_a(R)$), d is the cardinality of its domain and c is the cardinality of its codomain. The first table in Fig. 2 represents a full relation where $d = 6$, $c = 6$ and $r = 16$. Below this table is a list of hyper rectangles that are extracted for each attribute from the full relation, and their corresponding weights calculated are shown on the right.

The process of computing hyper rectangles associated with each attribute and their corresponding weights is exemplified in Fig. 2. The term Optimal Hyper Rectangle, $maxH(R)$, refers to the rectangle with the maximum weight extracted from the full relation. The second table in Fig. 2 represents the optimal hyper rectangle for the initial relation. The full relation is then split into the optimal hyper concept and the remaining binary relation. The remaining binary relation is then used as a new initial relation to extract the next optimal hyper rectangle. This process is repeated iteratively until full converage is achieved i.e. the full relation is covered by multiple optimal rectangles. This will produce a list of attributes qualifying for the Level 1 hyper concept. For the sake of brevity and pertinence, we exemplify Level 1 hyper concept by the example. In this example the optimal hyper rectangle and the binary relation in Fig. 2 fully converge the relation. So they represent Level 1 hyper concept with the keywords {Breast cancer, Radio Therapy}. A more detailed explanation of Hyper Concept can be found in Ref. [19].

c=6

d=6

	Breast cancer	Malignant	Benign	Chemo Therapy	Radio Therapy	Operation
t1	1	0	1	0	0	0
t2	1	1	0	0	0	0
t3	0	1	0	1	1	0
t4	1	0	1	1	0	1
t5	0	0	1	0	1	0
t6	1	1	0	1	0	0

	Breast cancer	Malignant	Benign	Chemo Therapy	Operation
t1	**1**	0	1	0	0
t2	**1**	1	0	0	0
t4	**1**	0	1	1	1
t6	**1**	1	0	1	0

w(H(R)) = 1

	Breast cancer	**Malignant**	Chemo Therapy	Radio Therapy
t2	1	**1**	0	0
t3	0	**1**	1	1
t6	1	**1**	1	0

w(H(R)) = 0.66

	Breast cancer	**Benign**	Chemo Therapy	Radio Therapy	Operation
t1	1	**1**	0	0	0
t4	1	**1**	1	0	1
t5	0	**1**	0	1	0

w(H(R)) = 0

	Breast cancer	Malignant	Benign	**Chemo Therapy**	Radio Therapy	Operation
t3	0	1	0	**1**	1	0
t4	1	0	1	**1**	0	1
t6	1	1	0	**1**	0	0

w(H(R)) = 0.55

	Malignant	Benign	Chemo Therapy	**Radio Therapy**
t3	1	0	1	**1**
t5	0	1	0	**1**

w(H(R)) = 0.625

	Breast cancer	Benign	Chemo Therapy	**Operation**
t4	1	1	1	**1**

w(H(R)) = -1

Remaining Binary Relation:

	Breast cancer	Malignant	Benign	Chemo Therapy	Radio Therapy	Operation
t3	0	1	0	1	1	0
t5	0	0	1	0	1	0

Fig. 2. Calculating weights for hyper rectangles.

3.4. Sampling methods

The most commonly applied sampling methods in statistics are simple random sampling and stratified sampling. Simple random sampling is based on arbitrary selection of items, without requiring the fulfillment of any criterion. All elements in the population have equal probability of being selected for the sample. Contrarily, stratified sampling categorizes the population into strata based on some criteria, such as age, followed by selection of items from each stratum [5]. The categorization into strata should be such that they satisfy the requirement of being mutually exclusive and collectively exhaustive. Albeit their apparent ease of application, samples generated from this method fail to preserve the functional dependencies present in the data which is necessary to create a representative subset of the dataset. It also requires prior knowledge of the data to be able to categorize into strata. In fact, it is possible to derive meaningful insights from the sampled data and focus on the important aspects of the dataset by applying nonrandom sampling techniques [29].

3.5. Coupling sampling algorithm

This data sampling technique is based upon FCA. A similarity based formal context is first generated from a given database instance. Each row in the formal context represents a pattern of 0s and 1s depending upon the similarity between attributes of the objects in the object pair. For example, in Fig. 3 the pattern for object pair ($\mathcal{O}1$, $\mathcal{O}2$) is 0110. Thus,

	A1	A2	A3	A4
O1	A	C	D	A
O2	D	C	D	C
O3	A	Y	D	A
O4	D	C	X	C
O5	A	C	D	A
O6	D	C	D	C

Conversion to pairwise FC

	A1	A2	A3	A4
(O1, O2)	0	1	1	0
(O1, O3)	1	0	1	1
(O1, O4)	0	1	0	0
…	…	…	…	…
(O5, O6)	0	1	1	0

Conversion to object pattern table

	P1	P2	P3	P4	P5	P6	P7
O1	1	1	1	1	0	0	0
O2	1	0	0	1	1	1	0
O3	0	1	0	0	1	0	1
O4	0	0	1	0	0	1	1
O5	1	1	1	1	0	0	0
O6	1	0	0	1	1	1	0

Fig. 3. Obtaining the object pattern table.

both $\mathcal{O}1$ and $\mathcal{O}2$ belong to pattern 0110. Each object pair can only have one pattern, while a pattern could belong to 0 or more object pairs, for example the pattern 0110 also belongs to the object pair ($\mathcal{O}1$, $\mathcal{O}6$), ($\mathcal{O}2$, $\mathcal{O}5$) and ($\mathcal{O}5$, $\mathcal{O}6$).

Algorithm 1 depicts the conceptual sampling algorithm, that chooses the highly coupled objects, which are the objects with the maximum total number of patterns. The algorithm starts from an object pattern table, where the rows represent the objects in the dataset and the columns represent the patterns, as shown in the last table in Fig. 3. A value of '1' indicates that an object belongs to a specific pattern.

Algorithm 1. Coupling Sampling Algorithm.

Input: ObjectPatternTable
Output: Final sample = the set of distinct objects in α
  β = set of objects analyzed until now
  repeat
    max = set of non-analyzed objects that have the maximum total number of patterns Add *max* to β
    for each $pair$ in $max \times \beta$ **do**
      find the common patterns *(CP)* shared between the two objects of the corresponding pair
      for each common pattern in *CP* **do**
        if the pair belongs to this pattern and the pattern is not covered **then**
          Set the pattern as covered
          Add the pair to (α)
        End if
      End for
    End for
until all patterns are covered

The main goal of this algorithm is dataset size reduction by choosing the most representative objects of the dataset. As an example, ($\mathcal{O}1$, $\mathcal{O}2$) and ($\mathcal{O}1$, $\mathcal{O}6$) represents pattern 0110, while ($\mathcal{O}1$, $\mathcal{O}3$) and ($\mathcal{O}3$, $\mathcal{O}5$) represents pattern 1011. In this example, the algorithm should choose the first pairs for each pattern, that is object pairs ($\mathcal{O}1$, $\mathcal{O}2$) and ($\mathcal{O}1$, $\mathcal{O}3$). By doing so, we succeed in representing 2 patterns with 3 objects rather than 5, thereby reducing the number of selected objects in the reduced dataset.

4. Methodology

Our proposed sampling method exploits conceptual methods and the Hyper Context reduction technique to build a sample that aims to preserve the functional dependency information of the original data. This technique can be applied without any prior knowledge about the data. The dataset goes through four transformations to eventually reach an acceptable sample. The first step is to convert the dataset into binary Formal Context (FC) using pairwise tuple comparisons. The generated FC objects are considered as binary patterns belonging to each object in the pair respectively. The second step results in an object/pattern table from the binary relation in the first step. Objects are mapped to their patterns, and pattern occurrences for each object are calculated. In the third step, the hyper context reduction algorithm is applied to the object/pattern table, resulting in a reduced and representative patterns list for all objects. In the fourth and last step, the reduced table from the previous step goes through the process of conceptual data sampling. This algorithm extracts a sample of objects that are highly coupled. Fig. 1 demonstrates the 4-stage sampling process with a clear explanation for each stage in the following sections.

4.1. Conversion of data to formal context

In the initial stage, data is converted to a binary FC using the pairwise tuple comparison approach. This will result in $\frac{n(n-1)}{2}$ rows in the FC table, where n is the number of rows in the original dataset. As comparison based on exact equality often leads to information loss, therefore to avoid such a loss, objects are compared based on a similarity measure calculation. This accounts for a flexible and fuzzy approach for performing comparisons which quantifies the closeness of the two objects. The similarity measure for a pair of values is computed by the formula introduced in Ref. [28]:

$$1 - \frac{n_1 - n_2}{max(n_1, n_2)},$$

where n_1 and n_2 are two numbers. Based on a fixed similarity threshold, the FC object is populated by the Boolean value '1' if the similarity measure equals or exceeds the threshold, otherwise the value is '0'. In the resulting FC table, each tuple is a binary pattern. Table 3 shows a subset of the FC output after tuple comparison from Table 2, which is used as an example to demonstrate the steps of the process. The FC in Table 3 is generated based on a 70% similarity threshold.

4.2. Formal context to object/pattern table

In the second stage, the binary data from the FC table is tabulated into an object pattern table. The binary instances from the FC table are represented as patterns by the attributes of this table. Each object from the original dataset is mapped to its patterns calculated in the FC table.

Table 2
Original database instance.

	A	B	C	D
t1	2	7	6	4
t2	4	14	9	4
t3	2	7	7	3
t4	2	16	9	6
t5	5	7	9	6
t6	1	5	5	4
t7	2	8	4	4
t8	1	6	7	5
t9	4	13	9	6
t10	6	19	18	13

Table 3
FC table.

	A	B	C	D
(t1, t2)	0	0	0	1
(t1, t3)	1	1	1	1
(t1, t4)	1	0	0	0
(t1, t5)	0	1	0	0
(t1, t6)	0	1	1	1
(t1, t7)	1	1	0	1
(t1, t8)	0	1	1	1
(t1, t9)	0	0	0	0
(t1, t10)	0	0	0	0
(t2, t3)	0	0	1	1
(t2, t4)	0	1	1	0
…	…	…	…	…
(t9, t10)	0	0	0	0

Naturally, every object may be associated to one or many patterns as a result of the pairwise comparison. The number of patterns in the object pattern table is 2^n, where n is the number of attributes in the original dataset. For each object, the frequency of its associated binary patterns is calculated. If an object belongs to a particular pattern, the value for that cell is populated as '1', otherwise the value is '0'. Consider the object pattern table presented in Table 4. The binary attributes are written in their decimal equivalent. The total number of patterns is 2^4. As seen from Table 3, the pair (t1, t2) form the binary pattern '0001'. When converted to decimal, it is represented as '1'. Therefore, for the pattern '1', t1 and t2 both have the Boolean value of true (1). The value of '0' of pattern 6 corresponding to t1 implies that t1 did not make this pattern with any other tuple during FC conversion.

4.3. Object/pattern table to hyper concept

The hyper contextual method is applied to the object pattern table, which gives us coverage of the entire table in terms of its most representative patterns. In this technique, only attributes from Level 1 hyper context are extracted. The end result is an object pattern table with fewer (and most important) patterns. When the hyper context algorithm is applied to the object pattern table (Table 4) shown in this paper, it generates the patterns 0 and 3 in Level 1. Hence, the object pattern table now contains only the patterns extracted by the Hyper Context algorithm as depicted in Table 5.

Table 4
Object pattern table.

	0	1	2	3	4	5	6	…	15
t1	1	1	0	0	1	1	0	…	1
t2	0	1	1	1	1	0	1	…	0
t3	1	0	1	1	0	1	1	…	1
t4	1	0	1	1	1	0	1	…	0
t5	1	0	1	1	1	0	1	…	0
t6	1	1	0	1	1	0	0	…	1
t7	1	1	0	1	1	1	0	…	0
t8	1	0	0	1	0	1	1	…	1
t9	1	0	1	1	0	0	1	…	0
t10	1	0	0	0	1	0	0	…	0

Table 5
Object Pattern Table after applying Hyper Context.

	0	3
t1	1	0
t2	0	1
t3	1	1
t4	1	1
t5	1	1
t6	1	1
t7	1	1
t8	1	1
t9	1	1
t10	1	0

4.4. Conceptual sampling algorithm

In the last step, a sampling algorithm focused on preserving high coupling is applied on the reduced object pattern table, which was generated as a result of the application of Hyper Context. This will produce a sample of objects that are highly coupled. Consequently, all resulting objects in the sample have a unique combination of patterns. Table 6 displays the resulting sample after applying the conceptual sampling algorithm on the slim version of object pattern table from the previous step. The sample contains 3 tuples.

5. Evaluation

5.1. Experimental setting

Our experiments were performed on microscopic images of breast cancer tissue from the MITOS 2012 dataset [30]. This dataset contains 50 images from 5 patients where each image consists of 512 × 512 pixels that are annotated as malignant or benign, based upon ground-truth data that were manually marked by experienced pathologists. These images were passed through a Maximum Response 8 (MR8) filter bank which gave us 8 texture filter responses that are used as the features for each image. The reason this was done was because Dhoha showed in her thesis [2] that the MR8 filter gave the most discriminative features, as compared to other textural features such as Gabor and Phase Gradient. Samples are generated and tested using two methods: data split and cross validation, each with different configurations.

In the data split method, the dataset is divided into two equal partitions. From the first partition, 10,000 pixels are selected randomly from 25 images (400 pixels from each image). The experiments were also repeated with 50,000 pixels in the first partition (2000 pixels from each image). This was given as input to our proposed sampling method. The resulting sample was used as a training dataset for five different machine learning algorithms. These algorithms include Naive Bayes (NB), Support Vector Machine (SVM), Pattern Net (PN), Cascade Forward Net (CFN), Feed Forward Net (FFN). The second partition with the remaining 25 images (512 × 512 pixels) was used for testing to

Table 6
Sample.

	A	B	C	D
t2	4	14	9	4
t4	2	16	9	6
t6	1	5	5	4

compare the performance of the five classifier models in terms of F1 measure and accuracy. Furthermore, the effect of the similarity measure on the quality of sample was gauged by using different similarity thresholds for FCA. These threshold configurations were set at 70%, 80% and 90%.

In cross validation, the dataset is divided into 10 folds, wherein 9 folds are supplied as a training dataset and one is used for testing. This process is repeated 10 times with a different combination of folds for the training dataset. Sample size is calculated as the average of the samples generated in all 10 iterations. We conducted the experiments on two configurations of 10-fold cross validation for the MITOS Dataset. The images are subdivided into 10 sets of 5 images. For the first configuration, a random sample of 750 rows is selected from each of the 50 images, making a total of 3750 (=750*5) rows per fold. For the second configuration, a random sample of 2500 rows is selected from each image, reaching a total of 12500 (=2500*5) rows in each fold. Similar to the data split method, the threshold configurations are also varied, and the same machine learning algorithms are used for classification. Results are compared in terms of Accuracy and F1-measure.

In addition, the results from our sampling method produced using the data split and cross validation methods are compared with the corresponding results from the pattern-based proportional sampling method proposed in Ref. [27]. The machine learning algorithms that are used for the experiments are described in the next sub-section. The results of the evaluation and comparison are presented in the following machine learning sub-section.

5.2. Machine learning

Naive Bayes is one of the most common machine learning algorithms that uses probability theory to classify objects. It is based on Bayes' theorem with the assumption of independence between attributes of data points. The most popular use cases of Naive Bayes classifier include spam filters, medical diagnosis, and others.

The Support Vector Machine (SVM) is another popular supervised machine learning technique. Due to its flexibility, it is used for classification, regression and novelty detection or outliers detection [10]. The underlying technique of SVM is a non-probabilistic binary linear classifier. Thus, given a labeled training set, an SVM trained model can assign new unseen examples to one of previously seen labels. Two different architectures of neural networks are used in this study namely, Feed Forward Net (FFN) and Cascade Forward Net (CFN). The FFN is the simplest type of network, where it consists of three layers: input, hidden and output. The data flows in one direction: forward, starting from input layer, through hidden layer and finally to the output layer. Thus, there are no cycles in FFN. Meanwhile, with the CFN, there is a connection between each layer to all following layers. For example, CFN includes two connections from the input layer to the hidden and output layer, while FFN includes one direct connection from the input layer to the hidden layer. One advantage of using a cascade-network architecture is that it can learn accurately the relationships between input and output layer by having more connections.

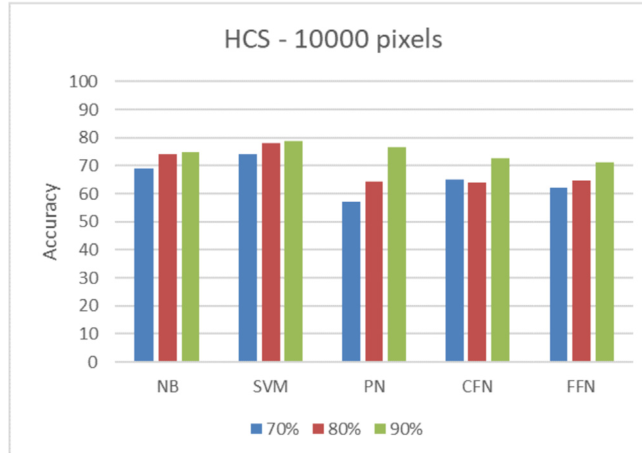

Fig. 5. Accuracy of all classifiers using Data Split (10,000 pixels).

In addition to the above, and given this is a pattern recognition classification problem, the Pattern Recognition Neural Network (PN) is also considered as a suitable technique. PN is based upon the Feed Forward Artificial Neural Networks that use the backpropagation (BP) algorithm for training (FFBPNN), and it is trained to classify inputs according to the target classes. The target data for PN consists of vectors of all zero values except for a 1 in the position where the target class is positioned [4].

5.3. Data split results

In our experiments, we observed that the sample size grows with the similarity threshold. Fig. 4 shows the sample sizes obtained from the two configurations of the training set by using three different similarity thresholds. A higher similarity threshold causes a decrease in the number of ones, thus requiring more objects in the resulting sample to preserve data characteristics. The highest number of tuples obtained in the sample for the data split method is 76 by using a similarity threshold of 90%. It is also evident that the dataset with 50000 pixels significantly increased the sample size only for 90% similarity threshold. However, it cannot be concluded that bigger datasets would generate larger samples, as it solely depends upon the functional

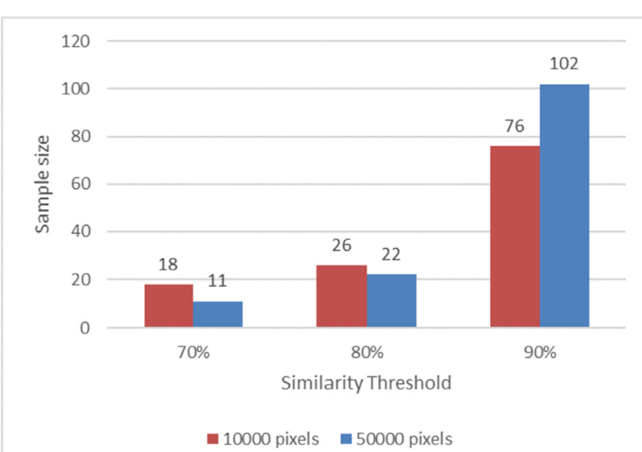

Fig. 4. Sample sizes for data split.

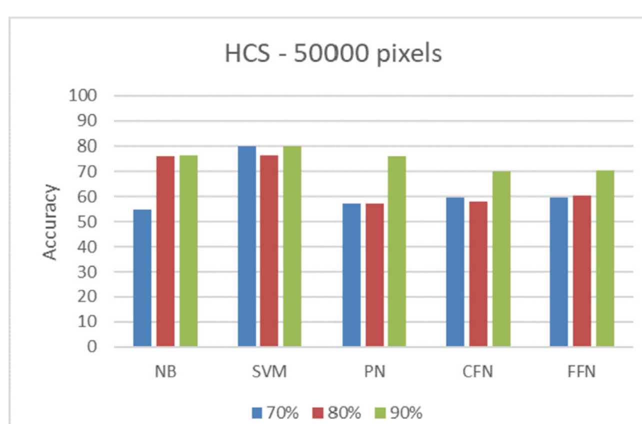

Fig. 6. Accuracy of all classifiers using Data Split (50,000 pixels).

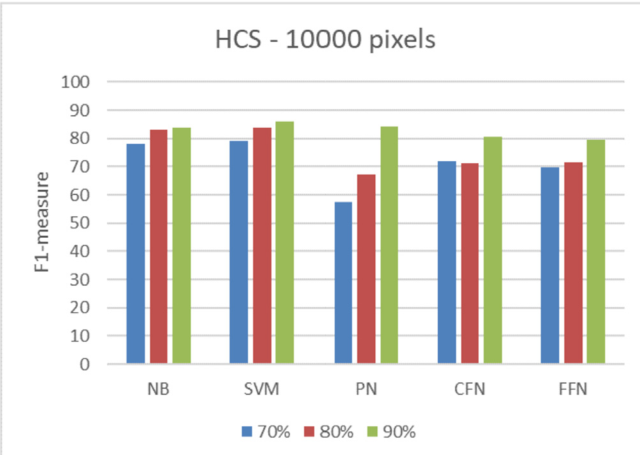

Fig. 7. F1-measure of all classifiers using Data Split (10,000 pixels).

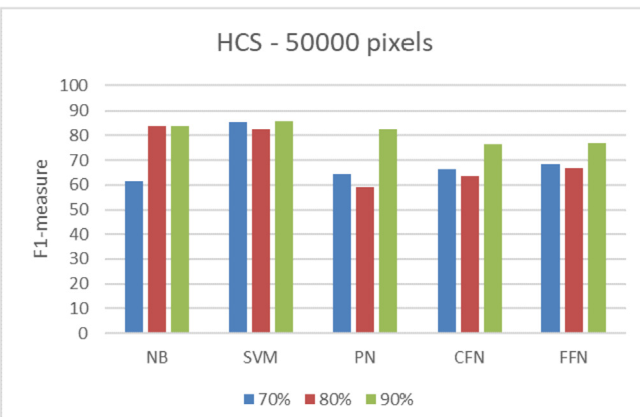

Fig. 8. F1-measure of all classifiers using Data Split (50,000 pixels).

dependencies present in the data. Tight coupling would generate a small sample, whereas loose coupling may incorporate a larger number of tuples, ultimately leading to a larger sample.

The comparison of the accuracy and F1-measure from the 5 classification algorithms using different similarity thresholds is presented in Figs. 5–8. For neural networks (PN, CFN, FFN), repeated tests on the same sample produces highly varying results. Therefore, the recorded measure is an average of 5 experiments. It can be observed from Figs. 5–8 that the results in terms of accuracy and F1-measure for the 50000-pixels dataset are consistent with the results for the 10000-pixels dataset: a higher similarity threshold (90%) yields better results than a lower one (70%). In other words, a bigger sample generates better results for all algorithms as compared with smaller samples. This is in compliance with the statistical fact that bigger sample sizes yield better results.

Moreover, among all classifiers, SVM achieves best accuracy for both 10000 and 50000 pixels. The highest accuracy for 10000 pixels is met at 79% equally for the similarity threshold configured at 80% and 90%. Results for the 50000 pixels subset provided an even better accuracy of 80% for SVM using 70% and 90% similarity threshold. In contrast, Neural Networks (FN, CFN and FFN) did not yield plausible results as they tend to perform better with larger datasets [37]. Contrarily, our samples range between 11 and 102 records only. Furthermore, SVM also performed best for F1-measure as compared to other classifiers. Among all three configurations for similarity thresholds in the data split method, optimal F1-measure by SVM is met at 86% using the 90% threshold for similarity.

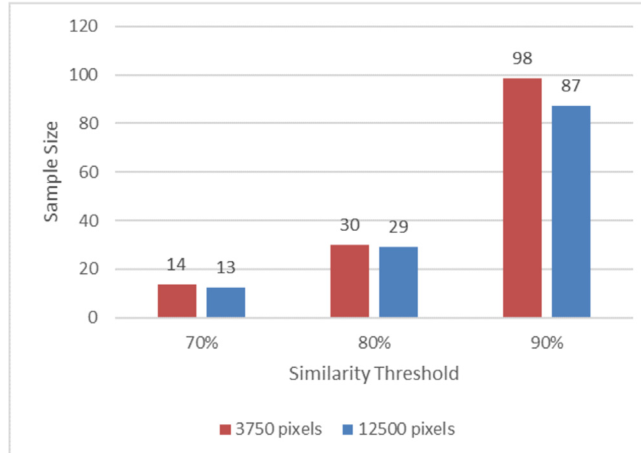

Fig. 9. Sample sizes for cross validation.

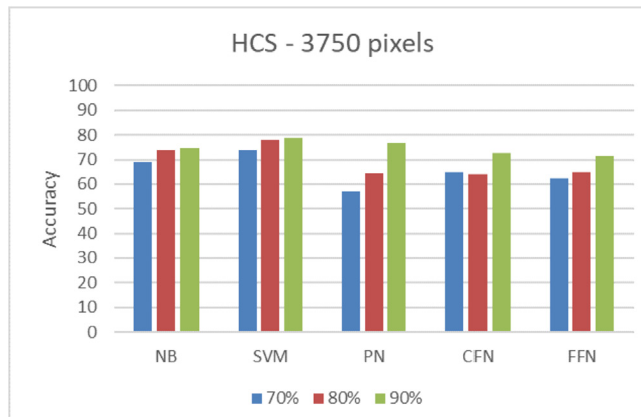

Fig. 10. Accuracy of all classifiers using Cross Validation (3750 pixels).

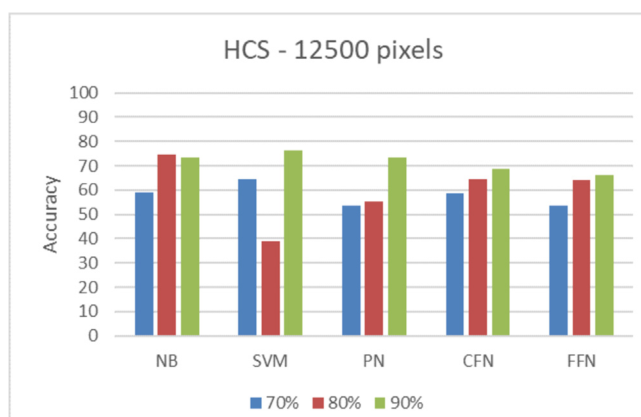

Fig. 11. Accuracy of all classifiers using Cross Validation (12,500 pixels).

5.4. Cross validation results

The experiments with cross-validation generated average sample sizes of 14, 30 and 98 for different similarity thresholds using the first configuration of 3750 pixels per fold. Fig. 9 shows the average sample sizes generated by the 10-fold cross-validation for different similarity thresholds. The sample sizes for cross validation are also consistent with those produced by the data split method.

Figs. 10 and 11 present the different measures of accuracy yielded by the 5 classifiers using cross-validation. Clearly, the samples

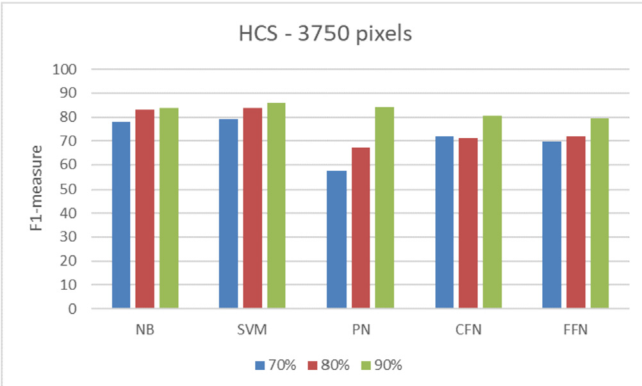

Fig. 12. F1-measure of all classifiers using Cross Validation (3750 pixels).

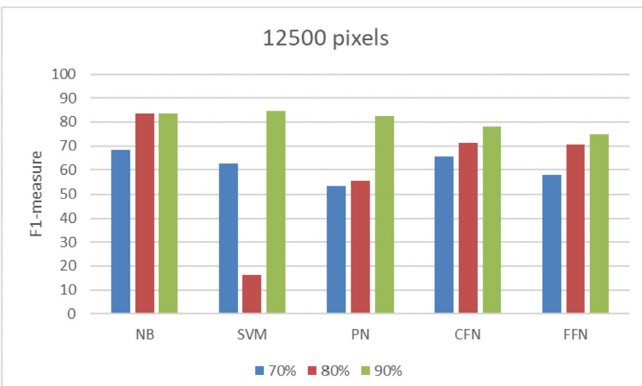

Fig. 13. F1-measure of all classifiers using Cross Validation (12,500 pixels).

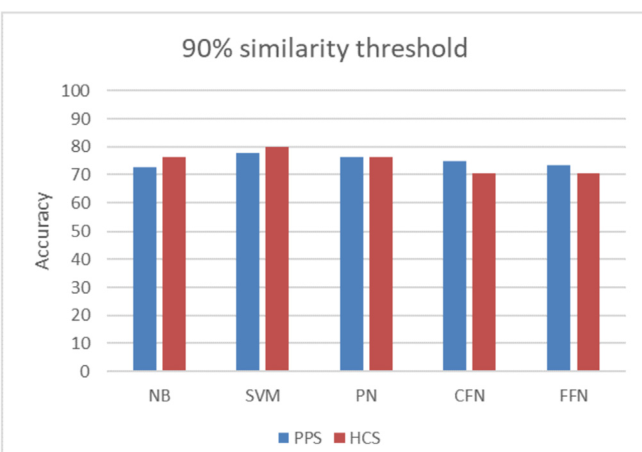

Fig. 14. Accuracy of all classifiers using Data Split for HCS and PPS.

generated using the similarity thresholds of 80% and 90% produce increased accuracy than a threshold of 70%. Moreover, SVM and NB perform better than other classifiers in all similarity threshold configurations. SVM generates the best accuracy measure of 79% for 90% similarity using the 3750 pixels configuration.

In terms of F1-measure, all classifiers produce more than 79% for the 90% similarity threshold. Overall, an optimal F1-measure of 86% is produced by SVM using the cross-validation method. Figs. 12 and 13 illustrate the results of F1-measure by all classifiers using cross-validation. Apparently, the trend is observable from the results for both subsets: a 90% similarity threshold produces better results than the other thresholds.

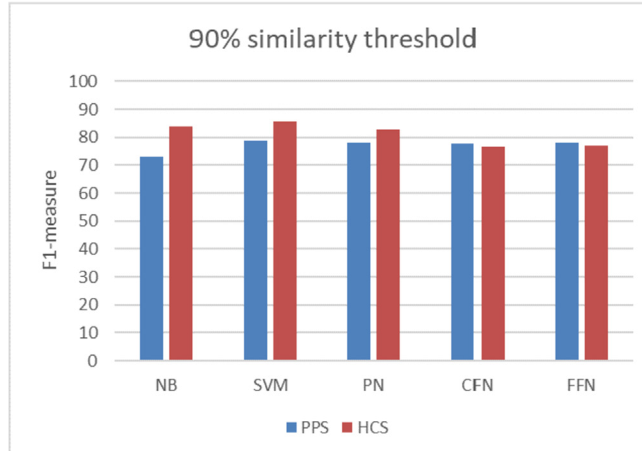

Fig. 15. F1-measure of all classifiers using Data Split for HCS and PPS.

5.5. Comparison of HCS with PPS

Our sampling method has undoubtedly exhibited remarkable performance by surpassing the highest classification accuracy obtained by PPS and reducing the sample size. Fig. 14 compares the results of HCS and PPS in terms of accuracy for a 90% similarity threshold. In this case, the differences between the accuracy are significant and also comparable. HCS obtained a best accuracy of 80%, outperforming PPS by 2% for SVM. HCS also presented improved performance compared to PPS for NB, with an accuracy of 76% as compared with 73% for PPS. It is also worth noting that even though HCS generated smaller sample sizes than PPS, it produced comparable, and in some cases, improved results over PPS. Thus, HCS proves to provide better classification accuracy by preserving functional dependencies with an even smaller sample.

Fig. 15 illustrates the comparison of results from HCS and PPS for F1-measure using the data split method. Evidently, when using a 90% similarity threshold, HCS outperforms PPS for the three algorithms NB, SVM and PN by a striking 11%, 7% and 5%, respectively. Additionally, it can also be observed that the performance of HCS is almost similar to PPS for FFN and CFN.

Similarly, using the cross-validation method and the similarity threshold configured at 90%, HCS again outperforms PPS for NB, SVM and PN in terms of both accuracy and F1-measure. The F1-measure and accuracy results for the samples generated using 90% similarity threshold by both sampling methods using cross validation are shown in Figs. 17 and 16 respectively.

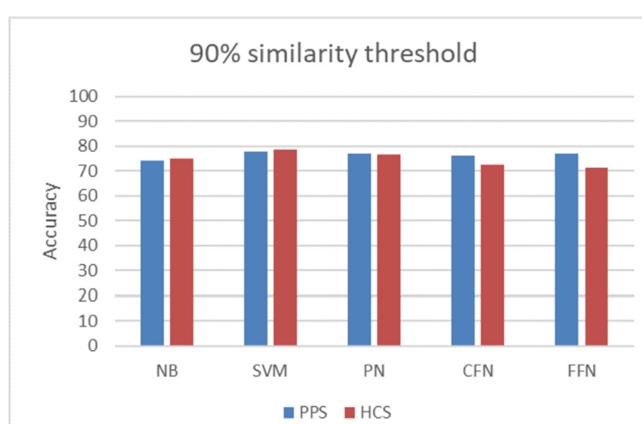

Fig. 16. Accuracy of all classifiers using Cross Validation for HCS and PPS.

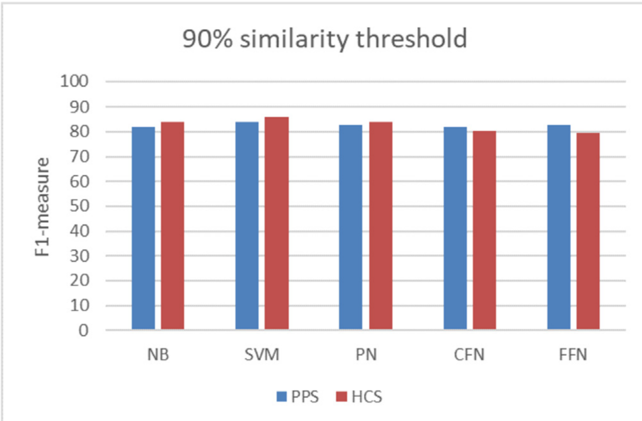

Fig. 17. F1-measure of all classifiers using Cross Validation for HCS and PPS.

5.6. Comparison with other methods

Looking at the results in the thesis of Dhoha, we can see that she worked on a sample of 137,500 pixels sampled randomly from 25 images similar to our data split method. She then used Naive Bayes (NB) and Support Vector Machine (SVM) classifiers to train on this sample and then test against the other 25 images. She managed an accuracy of 75% with NB and 78% with SVM and an F1-measure of 83% for both classifiers. Looking back at our data split method results, the best accuracy and F1-measure values for NB were 77% and 82% respectively and for SVM, they were 80% and 86% respectively. Using an Artificial Neural Network classifier, Dhoha managed to obtain a 77% accuracy and 82% F1-measure whereas from our neural network classifiers, the Pattern Recognition Neural Network had the best results with 76% accuracy and 82% accuracy. Thus, we see that our method gives similar or better results than the thesis. The more fascinating aspect is that we managed to obtain these values by just using a sample of 102 pixels from our method.

Pixel-wise classification of cancerous regions in breast cancer histopathological images using deep convolutional networks is considered state-of-the-art [12]. In terms of time efficiency, the method proposed by Ciresan et al. [14] requires 24 h to train the network with an optimized GPU, and achieves an F-score of 0.782. Improving on this, Wahab et al. [34] proposed a method which requires 15 h of training and achieves an F-score of 0.79. It seems that using the MR8 filter responses as features provides better results in general, as compared to the work done by other papers. HCS also facilitates the training process by training on a small sample instead of feeding the entire partition, thereby reducing the training time dramatically. Our experiments were carried out on a 64-bit processor with Intel(R) Core(TM) i7 CPU @ 2.30 GHz. The longest training time required by HCS is only 36 s, meanwhile improving the F-score to 0.859. This renders our approach practical and readily applicable for medical image classification.

6. Conclusion

We proposed a sampling method, HCS, consisting of a unique combination of existing techniques. The sampling procedure is rooted in Formal Concepts. It goes through a conversion to formal concept analysis followed by conversion to object pattern table. It exploits the hyper context algorithm for pattern reduction. Thereafter, the coupling sampling algorithm is applied to generate the sample. The resulting samples are measured for accuracy and F1-measure using 5 machine learning algorithms. The results were also evaluated against PPS, and HCS definitely proved to be competitive using different learning configurations by providing improved accuracy with high F1-measure.

Moreover, the resulting sample generates a very concise sample using binary distribution, which captures the functional dependencies and associations present in the original dataset. Therefore, it is able to deliver competitive results. Additionally, the patterns are generated based on all features, without the loss of any of the features. Furthermore, the sampling technique does not depend upon prior knowledge of class distribution.

Acknowledgement

This contribution was made possible by NPRP-07-794-1-145 grant from the Qatar National Research Fund (a member of Qatar foundation). The statements made herein are solely the responsibility of the authors.

References

[1] Abdel-Zaher Ahmed M, Eldeib Ayman M. Breast cancer classification using deep belief networks. Expert Syst Appl 2016;46:139–44.
[2] Abid Dhoha. Segmentation of tumor regions in microscopic images of breast cancer tissue Master's thesis Doha, Qatar: Qatar University; 2016
[3] Akay Mehmet Fatih. Support vector machines combined with feature selection for breast cancer diagnosis. Expert Syst Appl 2009;36(2):3240–7.
[4] AL-Allaf Omaima N Ahmad, AbdAlKader Shahlla A, Tamimi Abdelfatah Aref. Pattern recognition neural network for improving the performance of iris recognition system. J. Sci. Eng. Res. 2013;4.
[5] Armitage Peter, Berry Geoffrey, Matthews John Nigel Scott. Statistical methods in medical research. John Wiley & Sons; 2008.
[6] Atif Jamal, Hudelot Céline, Bloch Isabelle. Explanatory reasoning for image understanding using formal concept analysis and description logics. IEEE Trans. Syst. Man Cybern. B Cybern. 2014;44(5):552–70.
[7] Atzeni Paolo, Morfuni Nicola M. Functional dependencies and constraints on null values in database relations. Inf Contr 1986;70(1):1–31.
[8] Ayech Mohamed Walid, Ziou Djemel. Segmentation of Terahertz imaging using k-means clustering based on ranked set sampling. Expert Syst Appl 2015;42(6):2959–74.
[9] Baixeries Jaume, Kaytoue Mehdi, Napoli Amedeo. Characterizing functional dependencies in formal concept analysis with pattern structures. Ann Math Artif Intell 2014;72(1–2):129–49.
[10] Bennett Kristin P, Campbell Colin. Support vector machines: hype or hallelujah? Acm Sigkdd Explorations Newsletter 2000;2(2):1–13.
[11] Blaschke Thomas, Hay Geoffrey J, Kelly Maggi, Lang Stefan, Hofmann Peter, Addink Elisabeth, Feitosa Raul Queiroz, van der Meer Freek, van der Werff Harald, van Coillie Frieke, et al. Geographic object-based image analysis–towards a new paradigm. ISPRS J Photogrammetry Remote Sens 2014;87:180–91.
[12] Chen Hao, Dou Qi, Wang Xi, Qin Jing, Heng Pheng-Ann, et al. Mitosis detection in breast cancer histology images via deep cascaded networks. AAAI. 2016. p. 1160–6.
[13] Cho Gene Young, Moy Linda, Zhang Jeff L, Baete Steven, Lattanzi Riccardo, Moccaldi Melanie, Babb James S, Kim Sungheon, Sodickson Daniel K, Sigmund Eric E. Comparison of fitting methods and b-value sampling strategies for intravoxel incoherent motion in breast cancer. Magn Reson Med 2015;74(4):1077–85.
[14] Cireşan Dan C, Giusti Alessandro, Gambardella Luca M, Schmidhuber Jürgen. Mitosis detection in breast cancer histology images with deep neural networks. International conference on medical image computing and computer-assisted intervention. Springer; 2013. p. 411–8.
[15] E DeSantis Carol, Fedewa Stacey A, Goding Sauer Ann, Kramer Joan L, Smith Robert A, Jemal Ahmedin. Breast cancer statistics, 2015: convergence of incidence rates between black and white women. CA A Cancer J Clin 2016;66(1):31–42.
[16] Feng Li, Grimm Robert, Block Kai Tobias, Chandarana Hersh, Kim Sungheon, Xu Jian, Axel Leon, Sodickson Daniel K, Otazo Ricardo. Golden-angle radial sparse parallel MRI: combination of compressed sensing, parallel imaging, and golden-angle radial sampling for fast and flexible dynamic volumetric MRI. Magn Reson Med 2014;72(3):707–17.
[17] Ferjani Fethi, Elloumi Samir, Jaoua Ali, Ben Yahia Sadok, Ismail Sahar, Ravan Sheikha. Formal context coverage based on isolated labels: an efficient solution for text feature extraction. Inf Sci 2012;188:198–214.
[18] Formica Anna. Semantic web search based on rough sets and fuzzy formal concept analysis. Knowl Base Syst 2012;26:40–7.
[19] Hassaine Abdelaali, Mecheter Souad, Jaoua Ali. Text categorization using hyper rectangular keyword extraction: application to news articles classification. International conference on relational and algebraic methods in computer science. Springer; 2015. p. 312–25.
[20] Kettunen Markus, Manzi Marco, Aittala Miika, Lehtinen Jaakko, Durand Frédo, Zwicker Matthias. Gradient-domain path tracing. ACM Trans Graph 2015;34(4):123.
[21] Liang Jie, Zhou Jun, Qian Yuntao, Wen Lian, Bai Xiao, Gao Yongsheng. On the sampling strategy for evaluation of spectral-spatial methods in hyperspectral image classification. IEEE Trans Geosci Rem Sens 2017;55(2):862–80.
[22] Li Sheng-Tun, Tsai Fu-Ching. A fuzzy conceptualization model for text mining with application in opinion polarity classification. Knowl Base Syst 2013;39:23–33.
[23] Mert Ahmet, Kılıç Niyazi, Bilgili Erdem, Akan Aydin. Breast cancer detection with reduced feature set. Comput. Math. Meth. Med. 2015;2015.

[24] Priss Uta. Formal concept analysis in information science. Arist 2006;40(1):521–43.
[25] Qi Jianjun, Wei Ling, Yao Yiyu. Three-way formal concept analysis. International conference on rough sets and knowledge technology. Springer; 2014. p. 732–41.
[26] Ravi Kumar, Ravi Vadlamani, Prasad P Sree Rama Krishna. Fuzzy formal concept analysis based opinion mining for CRM in financial services. Appl Soft Comput 2017;60:786–807.
[27] Rezk Eman, Awan Zainab, Islam Fahad, Jaoua Ali, Al Maadeed Somaya, Zhang Nan, Das Gautam, Rajpoot Nasir. Conceptual data sampling for breast cancer histology image classification. Comput Biol Med 2017;89:59–67.
[28] Rezk Eman, Babi Syrinne, Islam Fahad, Jaoua Ali. Uncertain training data set conceptual reduction: a machine learning perspective. Fuzzy systems (FUZZ-IEEE), 2016 IEEE international conference on. IEEE; 2016. p. 1842–9.
[29] Ramos Rojas Julian A, Beth Kery Mary, Rosenthal Stephanie, Dey Anind. Sampling techniques to improve big data exploration. 2017 IEEE 7th symposium on large data analysis and visualization (LDAV). IEEE; 2017. p. 26–35.
[30] Roux Ludovic, Racoceanu Daniel, Lomenie Nicolas, Kulikova Maria, Irshad Humayun, Klossa Jacques, Capron Frederique, Genestie Catherine, Naour Gilles, Gurcan Metin. Mitosis detection in breast cancer histological images an icpr 2012 contest. J Pathol Inf 2013;4(1):8.
[31] Siegel Rebecca L, Miller Kimberly D, Jemal Ahmedin. Cancer statistics. CA A Cancer J Clin 2018;68(1):7–30. 2018.
[32] Su Yu, Agrawal Gagan, Woodring Jonathan, Myers Kary, Wendelberger Joanne, Ahrens James. Taming massive distributed datasets: data sampling using bitmap indices. Proceedings of the 22nd international symposium on High-performance parallel and distributed computing. ACM; 2013. p. 13–24.
[33] Su Yu, Agrawal Gagan, Woodring Jonathan, Myers Kary, Wendelberger Joanne, Ahrens James. Effective and efficient data sampling using bitmap indices. Cluster Comput 2014;17(4):1081–100.
[34] Wahab Noorul, Khan Asifullah, Lee Yeon Soo. Two-phase deep convolutional neural network for reducing class skewness in histopathological images based breast cancer detection. Comput Biol Med 2017;85:86–97.
[35] Wille Rudolf. Formal concept analysis as applied lattice theory. Concept lattices and their applications. Springer; 2008. p. 42–67.
[36] Ou Yang Yu-Ping, Shieh How-Ming, Tzeng Gwo-Hshiung, Yen Leon, Chan Chien-Chung. Combined rough sets with flow graph and formal concept analysis for business aviation decision-making. J Intell Inf Syst 2011;36(3):347–66.
[37] Zhang Le, Suganthan PN. A survey of randomized algorithms for training neural networks. Inf Sci 2016;364–365:146–55.
[38] Zheng Bichen, Won Yoon Sang, Lam Sarah S. Breast cancer diagnosis based on feature extraction using a hybrid of K-means and support vector machine algorithms. Expert Syst Appl 2014;41(4):1476–82.
[39] Zhu Zhe, Albadawy Ehab, Saha Ashirbani, Zhang Jun, Harowicz Michael R, Mazurowski Maciej A. Breast cancer molecular subtype classification using deep features: preliminary results. Proc SPIE 10575, Medical Imaging 2018: Computer-Aided Diagnosis. 27 February 2018:105752Xhttps://doi.org/10.1117/12.2295471.