Otto-von-Guericke Universität Magdeburg
Germany

DRESS - Project Report

Authors

Abhisar Bharti
Anustup Das
Shobhit Chaurasiya
Sumit Kundu

Supervisor

Prof. Myra Spiliopoulou
Tommy Hielscher

December 18, 2018

Contents

1 Introduction 2

2 Related Work 3

3 Fundamentals 3

4 Materials and Methodology 4

4.1 Dataset 4

4.2 Data Pre-processing 4

4.3 Our Algorithm: LW-DRESS 5

4.4 Feature Selection 6

4.4.1 Laplacian Filter 6

4.4.2 Subspace Creation using Forward Selection Method 7

4.4.3 Scoring Subspace 7

4.4.4 Clustering Algorithm 9

5 Results 10

5.1 Algorithm Variants 10

5.2 Variants under Evaluation 10

5.3 Experiments 11

5.4 Discussion 17

6 Conclusion 17

7 Future Work 18

8 Acknowledgement 18

Abstract

Recognizing the key dimensions from a dataset for accurate classification is the most important aspect to medical diagnosis today. Generally, the medical dataset is comprised of a very large number of features in which a major portions does not play any part in training the classification model. Feature selection is used to extract an optimal subset of features for classification, for which the classification accuracy is maximum. While labeled data is not easily available, unlabeled data are cheap and accessible. The algorithm - "Discovery of Relevant Example-constrained Subspaces" (DRESS)[7] proposed by Hielscher et al. aims to address this issue of semi-supervised feature selection using constrained similarity score while utilizing the density of clusters and the distances between the constraints. The objective of this project is to recreate the existing algorithm (DRESS) and attempt to optimize it in terms of performance and accuracy. We have used Laplacian Score to filter the unwanted features in the first step and modified the distance score by adding weights to it. The newly optimized algorithm - Laplacian Weighted Discovery of Related Example Constrained Subspaces (LW-DRESS) is then comparred with the original algorithm, DRESS along with other traditional feature selection methods on epidemiological study data and the findings are recorded. The report ends with a conclusion whether the proposed modification is better than the original algorithm or not.

1 Introduction

With advancement of technology and abundance of data, computers have started tackling increasingly complex learning tasks in various domains with amazing level of success. There are many applications for tasks involving supervised learning. But, it needs labeled data which is quite expensive. Two objects can be declared similar or dissimilar based on the outcome of a learning task. [7] uses as little information such as these and studies the performance on 'hepatic steatosis' using SHIP data [16] . Even though it provides great results over other algorithms as mentioned in [7], it doesn't bother about preserving the local structure of the data. It also doesn't incentivise or penalize subspaces while looking for the best subspace. In this report, we have proposed a solution in the form of an algorithm, LW-DRESS which is a modified version of DRESS.

We have divided the report into multiple sections. Next section discusses works related to our work. Our approach is based on subspace and constrained clustering.

In section - 3, we have discussed basic ideas about it. Section - 4 discusses materials and methodology used in detail. In section - 5, we have charted and described our results for different variants of our algorithm. We have written our conclusion in section - 6. The report culminates with section - 7 and 8 on future work and acknowledgement respectively.

2 Related Work

THE techniques of recognizing potential risk factors from the feature set on medical data, which plays a vital role in determining whether a person has a particular disease or not have been exhibited by [19] and [4]. The paper presented by Yoo et al. [19] uses penalized logistic regression models for relevant features selection on the data related to diabetic retinopathy from clinical health. The prediction of risk of future dengue incidence from epidemiological data using Fuzzy Association Rules is presented in [4]. Ren et al., 2008 [2] and Barkia et al., 2011 [14] used wrapper approach based on Semi-supervised Feature Importance Evaluation (SSFI) method and Forward Semi-supervised Feature Selection (FWSemiFS) method respectively. While the embedded approach for feature selection was followed by Helleputte and Dupont, 2009 [14] where Partial Supervised AROM (PS-l2-AROM) methods were applied for gene selection. Traditional feature selection methods like filter approach based on Correlation criteria and Mutual Information (MI) along with other methods like Recursive Feature Elimination techniques exists but in most cases they depend on the labeled data [5].

Work similar to the one presented in this report can be found in [10] where density-based bottom-up subspace clustering is used to evaluate a subspace irrespective of any target-concept. Along with this, [11] and [3] have also followed similar approaches to our solution where pairwise constraint score with Laplacian Score and Constrained Laplacian Score (CLS) are the focal feature selection techniques. CFS along with KNN [9] and Decision Trees [12] are used to evaluate the selected feature w.r.t. discriminative power on fatty liver.

3 Fundamentals

THE algorithm we have proposed is a modified version of [7]. Both algorithms are based on subspace clustering and constraint based clustering. The objective of subspace clustering is to identify clusters of similar objects on the subsets of the original feature set [13]. Whereas, constraint based clustering tries to find out clusters of objects that satisfy the given set of constraints [15]. Instance level constraints - must link (ML) and cannot link (NL) constraints help the clustering algorithms to find clusters that satisfies the constraints [18]. ML constraint {a, b} is said to satisfy if only if both instances are in the same cluster. On the other hand, NL constraint {x, y} is said to satisfy if only if both instances are in different clusters. Using these two types of constraints, our algorithm tries to find the best subspace in terms of cluster quality and constraint satisfaction. We seek subspaces that produce clusters having ML constraint pairs close to each other and NL constraints pairs far apart from each other.

4 Materials and Methodology

4.1 Dataset

THE data is a population based on epidemiological study, "Study of Health in Pomerania (SHIP)" [16]. This epidemiological data was collected by means of several ways which included interviews, exercise tests and laboratory analysis, ultrasound examinations and magnetic resonance tomography (MRT). In this project, we have worked with the data used by the original DRESS algorithm which is obtained from the latest study, SHIP-2. Generally the data exhibits binary classification traits where participants with a liver fat concentration value of less than 25% [less than 10] are assigned the negative class; participants with greater values [more than 10] are assigned the positive class. Out of 4308 instances, 578 are labeled [140 positive, 438 negative] and rest are unlabeled. This 578 labeled instances of study participants were segregated to male participants comprising 264 instances while the remaining 314 individuals to female participants. Out of this 264 male participants 31% have class labels as positive while 19% of the female population were labeled as positive.

4.2 Data Pre-processing

THE original dataset contains 4308 instances distributed on 405 features[16]. There are some features which had data as id and date fields and needed processing. The results of those features were irrelevant and hence, we have removed the features containing those fields. The name of such features are: *id, exdate_ship_s0, exdate_ship_s0, exdate_ship_s1, exdate_ship_s2, exdate_ship0_s0, blt_beg_s0, blt_beg_s1, blt_beg_s2*. The target field - *'mrt_liverfat_s2'* which contains numerical values for fatty liver is normalized and assigned class labels i.e. *0* and *1*. Values less than or equal to 10 are assigned label - *0* i.e. negative, and values more than 10 are assigned to label - *1* i.e. *positive*. After the removal of the fields and classifying the target feature, *?* in the dataset are replaced with *NaN* values. *?* could have been also replaced with *0* or any other value, but since it would have created a bias in the dataset, we replaced it with *NaN*. We categorized the features into two groups: one with the features having categorical values, and the other with the numerical values. Features having numerical values are normalized so that each value is between *0* and *1*, and the ones have categorical values are assigned a value for each specific category [7].

4.3 Our Algorithm: LW-DRESS

As stated earlier, LW-DRESS stands for Laplacian Weighted Discovery of Relevant Example Constrained Subspaces. It is a modified version of the original algorithm - DRESS designed by Hielscher et al. [7] which tries and finds out subspaces that separates the participants well into different categories based on the problem statement in hand, while using only a few constraints and without the need for labeled data. But unlike DRESS, our algorithm tries to preserve the local structure of the features while looking for subspace containing clusters that are homogeneous, far-apart and satisfies all constraints. LW-DRESS extracts a weighted quality score for each subspace and then expands it with more features if only if they lead to better clusters.

The workflow of LW-DRESS is mentioned in Figure 1. Firstly, LW-DRESS reduces the feature set by implementing laplacian filter and then generates a set of potential candidate subspaces. The candidate subspaces are filtered by clustering and calculating the weighted quality score for each candidate subspace. The subspace with best score is selected and features are added to generate more candidates. The algorithm terminates when there are no subspaces left with better score.

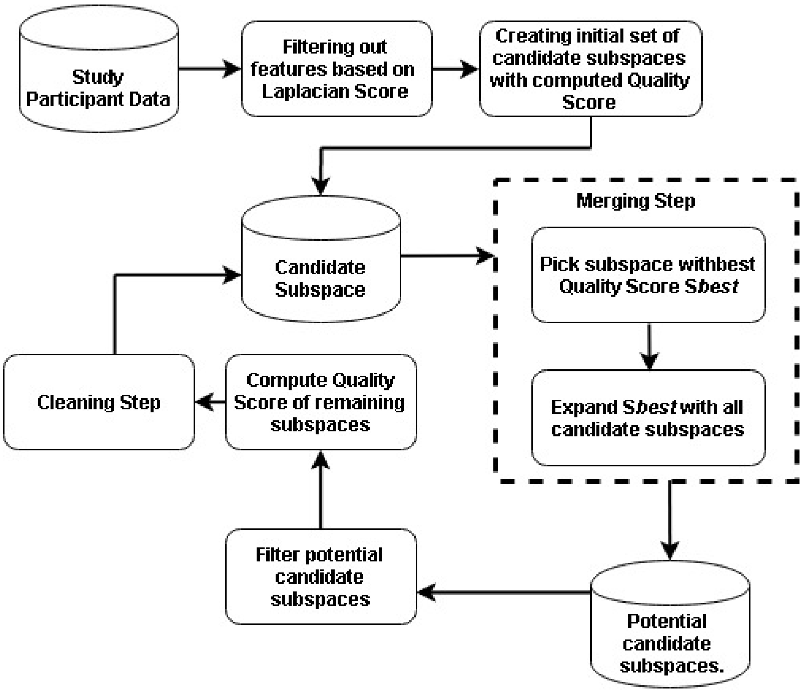
Study Participant Data
Filtering out features based on Laplacian Score
Creating initial set of candidate subspaces with computed Quality Score
Merging Step
Candidate Subspace
Pick subspace withbest Quality Score S*best*
Expand S*best* with all candidate subspaces
Cleaning Step
Compute Quality Score of remaining subspaces
Filter potential candidate subspaces
Potential candidate subspaces.

Figure 1: Modification on original DRESS workflow

4.4 Feature Selection

Feature selection is used to extract an optimal subset of features for classification, for which the classification accuracy is maximum. In order to find the optimum subspace of features from the dataset, we followed a series of steps in our algorithm.

4.4.1 Laplacian Filter

Feature selection approaches could be of various types: filter, wrapper or embedded. Here, we have used Laplacian Score (LS) [1] in a filter based approach to evaluate properties of data and features. Laplacian score is a measure of locality preserving ability of the features. It prefers features with large variances which gives more representative power. To filter out the best features, we calculated the laplacian score of each feature. The laplacian score is calculated using the sparse affinity matrix of sorted pairwise distance between each feature using HEOM as the distance metric. We selected only the features having non-NaN values as laplacian score, and and ignored the rest.

4.4.2 Subspace Creation using Forward Selection Method

Forward selection is a subset of wrapper method, which tries to learn a model with initially one subspace and based on its score keeps on adding more subspace iteratively until the performance does not improve any further. In our algorithm, we started off with each feature as an individual subspace, performed clustering, choose the subspace having best quality score $q(S_{best})$ amongst them and expand it further by combining it with remaining subspaces. and based on its quality we merged the subspace. This may lead to high computation time with increase in number of features. To avoid this computational issue, we filtered out all the subspaces having negative distance score and didn't consider them for future iterations. The iteration terminates when there are no more subspace left with score better than the current one or all the subspaces have been used up.

4.4.3 Scoring Subspace

Given a set of ML (must link) and NL (not link) constraints, LW-DRESS uses a custom function to compute the quality score of a subspace based on following two criteria: (1) constraint satisfaction (2) weighted distance between constraint objects.

We assumed (as in [7]) that similar objects appear together as dense clusters and they tend to satisfy both ML and NL constraints, i.e. ML constraint pairs appear in the same cluster and NL constraint pairs in different clusters.

For a subspace S, the constraint score is defined in [7] as:

$$q_{cons}(S) = \frac{|ML_{sat}(S)|+|NL_{sat}(S)|}{|ML|+|NL|}$$

where,

q_{cons} = constraint score

$|MLsat(S)|$ = number of satisfied ML constraints in subspace S

$|NLsat(S)|$ = number of satisfied NL constraints in subspace S

$|ML|$ = total number of ML constraints

$|NL|$ = total number of NL constraints

Even though q_{cons} assigns ranks the subspaces according to the degree of constraint satisfaction, it lead to two problems. First, it would lead to cases where a large number of subspaces satisfy the constraints and this would result in too many features being marked relevant. Second, it doesn't take into account the distance between the constraint pairs. [7] has addressed these two issues by formulating a distance score computed as the difference in average distance between ML and NL constraint pairs.

Although it leads to subspaces where ML constraint pairs are closer to each other and NL constraint pairs are far apart, it treats all the subspaces with positive distance score equally. This may lead to cases where [3] would fail to identify the ideal subspaces, the subspaces where ML constraint pairs are as close as possible to each other and NL constraint pairs are as far apart as possible from each other and miss them altogether. To avoid this, we defined the distance score, $q_{dist}(S)$ as the difference in weighted average distance between ML and NL constraint pairs. It incentivizes the subspaces where ML constraint pairs are closer than the NL constraint pairs and penalizes the subspaces having NL constraint pairs closer than ML constraint pairs.

For a subspace S, the weighted distance quality score is defined in [7] as:

$$q_{dist}(S) = d_{avg}^{NL}(S) - d_{avg}^{ML}(S)$$

$$\mathrm{d}_{avg}^{NL}(S) = [\textstyle\sum_{x,y \in NL; i,j \in ML} n(d_{x,y}^{NL} > d_{i,j}^{ML}) \backslash \mid ML \mid] \quad * \quad [\textstyle\sum_{x,y \in NL} (d(S,x,y)] \backslash \mid NL \mid]$$

$$\mathrm{d}_{avg}^{ML}(S) = [\textstyle\sum_{x,y \in ML; i,j \in NL} n(d_{i,j}^{NL} > d_{x,y}^{ML}) \backslash \mid NL \mid] \quad * \quad [\textstyle\sum_{x,y \in ML} (d(S,x,y)] \backslash \mid ML \mid]$$

where,

$q_{dist}(S)$ = weighted distance score

$d_{avg}^{NL}(S)$ = weighted average distance between NL constraint pairs

$d_{avg}^{ML}(S)$ = weighted average distance between ML constraint pairs

$n(d_{x,y}^{NL} > d_{i,j}^{ML})$ = number of instances where distance between ML constraint pairs are lesser than NL constraint pairs

$n(d_{i,j}^{NL} > d_{x,y}^{ML})$ = number of instances where distance between NL constraint pairs are greater than ML constraint pairs

$| ML |$ = total number of ML constraints

$| NL |$ = total number of NL constraints

As our data consists of missing values for both continuous as well as categorical features, we have used Heterogeneous Euclidean Overlap Metric (HEOM) as a distance function as shown in [8].

The distance function is defined as:

$$d(S, x, y) = \sqrt{\sum_{s \in S} \delta(s(x), s(y))^2}$$

where :

$$\begin{aligned} \delta(s(x), s(y)) &= 1, && \text{if s(x) = s(y) and s is nominal} \\ &= \text{ s(x) - s(y),} && \text{if s is continuous} \\ &= 0, && \text{otherwise.} \end{aligned}$$

The total quality score of a subspace is defined as [7]:

$$q(S) = q_{cons}(S) * q_{dist}(S)$$

4.4.4 Clustering Algorithm

We have used the same clustering algorithm as used in DRESS, i.e. Density-Based Spatial Clustering of Applications with Noise (DBSCAN)[6]. DBSCAN is a density based unsupervised clustering algorithm that efficiently seeks data with high density of data points and leaves out the data with low density. It also helps in identifying outliers and represents them as noise within the data during clustering. DBSCAN needs values of two parameters to perform clustering in different subspaces, namely *minPts* and *epsilon*.

- **Minimum Point (minPts) Selection**: minPts is the number of points needed within the range of epsilon for a point to be considered as core point. We have estimated it by taking the natural logarithm of the number of data points within the dataset.

  $$\text{minPts} = \text{round}(\log \mid dataset \mid)$$

- **Epsilon Selection**: Based on the *minPts* value, we estimated the value of epsilon for each subspace for each iteration of DBSCAN. For a given $m = minPts$, we computed the *m'th* nearest neighbour distance for all data points within the dataset and sorted them in ascending order[7]. We plotted *m-dist* graph with nearest neighbour distances as *Y-axis* and the corresponding points as *X-axis*. We drew a line between the two extreme points of the graph and choose the point on *Y-axis* which maximizes the shortest distance between the drawn line and the point on *m-dist* graph as the value of epsilon.

5 Results

We created three different variants of our algorithm and evaluated them on the basis of: (1) How a classifier that uses features selected by our algorithm performs compared to one that uses features selected by [7] and one that uses conventional feature selection?

5.1 Algorithm Variants

We created the following three variants of our algorithm:

- **Laplacian Weighted DRESS (LW-DRESS)**: Feature set is first reduced by using laplacian score as filter, and further reduced using DRESS while applying weighted quality score to compute score for the subspaces.
- **Laplacian DRESS (L-DRESS)**: DRESS is implemented on the features filtered out using laplacian score.
- **Weighted DRESS (W-DRESS)**: Weighted quality score is used to compute the subscore score while performing feature selection using DRESS.

5.2 Variants under Evaluation

In order to evaluate the different variants of our algorithm, we compared them against the below mentioned classifier variants:

- **kNN**: kNN classifier using all features from the dataset.
- **C4.5**: Decision tree classifier using all features from the dataset.

- **CFS + {kNN, C4.5}**: kNN and decision tree classifier using the features obtained from CFS.
- **DRESS + {kNN, C4.5}**: kNN and decision tree classifier using the features obtained from DRESS.
- **LW-DRESS + {kNN, C4.5}**: kNN and decision tree classifier using the features obtained from LW-DRESS.
- **L-DRESS + {kNN, C4.5}**: kNN and decision tree classifier using the features obtained from L-DRESS.
- **W-DRESS + {kNN, C4.5}**: kNN and decision tree classifier using the features obtained from W-DRESS.

For variants involving kNN, we set k to log(number of rows in training data) with uniform weighted voting and HEOM as distance metric. Whereas for C4.5 and its variants, we used standard parameters as provided in the python library.

5.3 Experiments

T0 evaluate the performance of our algorithm, we created a pipeline for each variant using random ML constraint pairs (10) and NL constraint pairs (10) with k-fold stratified cross validation (k = 5) which ensures that the distribution of classes from each class is preserved while creating train and test folds. We calculated accuracy, sensitivity, specificity, AUC and F-Measure for each fold, and averaged them. The python libraries used for this project are stated in Table 1. A detailed report of the selected sub-spaces are shown in Table 2 for each of the 5 folds. Table 3 summarizes the results of evaluation for all variants. The best performing algorithm amongst the feature selections algorithms shown as underlined, the best performing one of the lot is marked as bold and the baseline is marked in italics.

Task	**Library**
Dataframe Creation and Manupulation	*pandas*
Data Pre-processing	*preprocessing* from *sklearn*
Clustering	*DBSCAN* from *sklearn.cluster*
Laplacian Score Calculation	*lap_score* from *skfeature.function.similarity_based*
Sparse Matrix for Laplacian Score	*csc_matrix* from *scipy.sparse*
C4.5	*DecisionTreeClassifier* from *sklearn.tree*
CFS	*CFS* from *skfeature.function.statistical_based*
KNN Classifier	*NearestNeighbors* from *sklearn.neighbors*
Multiprocessing	*Parallel, delayed* from *joblib*
Evaluation	*metrics* from *sklearn*

Table 1: Python libraries used

Fold	Variants	Number of Constrains	Selected Feature Subspace	Number of Features
1	CFS	None	*[stea_s0, stea_alt75_s2, stea_alt75_s0, w_sample_s0, s2_frau_03_w_s2, gout_s2, quick_s2, atc_c08ca01_s2]*	9
	DRESS	ML Cons: 10 NL Cons: 10	*[hyp_s0, menopaus_yn_s2, abstain_s1, atc_c09aa02_s2, rubellai_s0, atc_a02a_s0, , quick_s2, earm_s0, atc_c01da_s0, atc_c09aa01_s0, atc_a02ba_s0]*	12
	LW-DRESS	ML Cons: 10 NL Cons: 10	*[antihyp_s0, atc_c07ab_s2, sleepp_s0, gout_s2, hypmed_s0, alcg7d_s0, menopaus_yn_s2, atc_m04a_s0, asthma_untreated_s0]*	9
	W-DRESS	ML Cons: 10 NL Cons: 10	*[stea_alt75_s0, atc_c07a_s2, atc_c07ab_s2, atc_g04c_s0, atc_r06a_s0, kidney_s0, atc_a02a_s0, angina_s0]*	8
	L-DRESS	ML Cons: 10 NL CONS: 10	*[exloc_ship0_s0, atc_c09aa_s0, gluc_s_s2, whratc_s0, kaffee_s0, asthma_untreated_s0]*	6

2	CFS	None	[stea_s0, stea_alt75_s2, stea_alt75_s0, w_sample_s0, s2_frau_03_w_s2, atc_c08ca08_s0, gout_s2, waiidf_s0]	7
	DRESS	ML Cons: 10 NL Cons: 10	[gx_rs11591741, gx_rs11597086, ffs_pattern_s0, atc_m01a_s0, atc_m01ae01_s0, abstain_s0, atc_c09aa02_s2, diphi_s0, abstain_s1, migr_s0, osteo_s0, mrt_no_measurements, gout_s2, atc_n05bp_s0, atc_r05cb_s0, atc_c03c_s0, atc_c07aa_s0, atc_c08ca05_s1, mrt_lower, mrt_mean, mrt_upper, alkligt_s1, pankr_s0, atc_c07aa_s1, inceq_s0, mcs_sf12_s0, mi_s0, asthma_untreated_s0]	28
	LW-DRESS	ML Cons: 10 NL Cons: 10	[syshyp_s0, sleepp_s0, gout_s2, whratc_s0, gluc_s_s2, cancer_s0, asthma_untreated_s0]	7
	W-DRESS	ML Cons: 10 NL Cons: 10	[knoten_s2, csmoking_s0, unitrac_s0, goiter_s2, female_s0, sex_s0, strata2_s0, angina_s0, atc_c10aa_s0, anti_hcv_s0]	9
	L-DRESS	ML Cons: 10 NL Cons: 10	[school_s0, schul_s0, atc_m01ae01_s0, atc_c09aa05_s2, atc_c09ca_s0, gout_s2, atc_c09aa02_s2, menopaus_yn_s1, atc_c03e_s0, atc_m01a_s0, asthma_untreated_s0]	11
3	CFS	None	[stea_s0 , stea_alt75_s2 ,stea_alt75_s0 ,waiidf_s0, menopaus_yn_w_s1 , gout_s2 , quick_s2 , w_sample_s0]	8
	DRESS	ML Cons: 10 NL Cons: 10	[ffs_pattern_s0, diabetes_s0, gluc_s_s2, gout_s2, knoten_s0, hyperlipid_s0, partner_s0, menopaus_yn_s2, sleepp_s2, atc_a12a_s0, atc_c08_s0, cancer_s0, earm_s2, atc_c09aa02_s2, atc_m01ae01_s0, mi_s0, pankr_s0, anti_hcv_s0]	18

	LW-DRESS	ML Cons: 10 NL Cons: 10	*[waistc_s0, partner_s0, thyr_s0, atc_c09aa01_s0, menopaus_yn_s2, atc_a09a_s0, atc_c07aa_s0, gout_s2, mi_s0, asthma_untreated_s0]*	10
	W-DRESS	ML Cons: 10 NL Cons: 10	*[stea_s0, stea_alt75_s2, asthma_untreated_s0, s2_frau_03_w_s2, gluc_s_s2, stea_alt75_s0, gout_s2, waiidf_s0]*	7
	L-DRESS	ML Cons: 10 NL Cons: 10	*[atc_c07a_s2, atc_c07ab_s2, knoten_s0, atc_c07a_s0, ncigd_s0, gluc_s_s2, node_s0, atc_a02bc_s0, asthma_untreated_s0]*	8
4	CFS	None	*[stea_s0, stea_alt75_s2, stea_alt75_s0, s2_frau_03_w_s2, angina_s0, atc_m04a_s0, waiidf_s0]*	7
	DRESS	ML Cons: 10 NL Cons: 10	*[stea_s2, atc_c07a_s0, atc_g03f_s0, gout_s2, atc_c07ab_s0, mi_s0, menopaus_yn_s2, hs_crp_s0, ggt_s_s2, asthma_untreated_s0]*	10
	LW-DRESS	ML Cons: 10 NL Cons: 10	*[waiidf_s0, smoking_s0, knoten_s0, node_s0, atc_g03a_s0, atc_c09aa05_s2, csmoking_s0, gout_s2, atc_c09ca_s0, asthma_untreated_s0]*	10
	W-DRESS	ML Cons: 10 NL Cons: 10	*[stea_alt75_s0, gluc_s_s2, csmoking_s0, atc_g04c_s0,female_s0, kidney_s0, atc_a02a_s0, angina_s0]*	8
	L-DRESS	ML Cons: 10 NL Cons: 10	*[sleepp_s0, som_huef_s2, menopaus_yn_s1, atc_g03f_s0, cancer_s0, atc_a02b_s0, atc_c09aa01_s0, gout_s2, asthma_untreated_s0]*	9
5	CFS	None	*[stea_s0, stea_alt75_s2, s2_frau_03_w_s2, stea_alt75_s0, gout_s2, waiidf_s0]*	6

	DRESS	ML Cons: 10 NL Cons: 10	[gx_rs11597390, stea_s2, physact_s2, mrt_lower, depre_s0, mrt_mean, atc_b01ac06_s0, gluc_s_s2, hemophilia_s0, abstain_s0, atc_m01ae01_s0, jodid_u_s2, mrt_upper, thrombos_s0, anti_hcv_s0, atc_c01da_s0, mrt_no_measurements, kaffee_s0, packyrs_s0, jodid_u_s0, ggt_s_s2, hdl_s_s0, zz_nr, lipo_a_s0, ggt_s_s0, crea_s_s0, hrs_s_s2, asthma_untreated_s0, atc_a03f_s0]	29
	LW-DRESS	ML Cons: 10 NL Cons: 10	[strata1_s0, exloc_ship0_s0, goiter_s2, menopaus_yn_s2, atc_r05cb_s0, psu_s0, cancer_s0, atc_a02a_s0, asthma_untreated_s0]	9
	W-Dress	ML Cons: 10 NL Cons: 10	[csmoking_s0, unitrac_s0, goiter_s2, female_s0, atc_c07a_s2, strata2_s0, angina_s0, atc_a02a_s0, anti_hcv_s0, knoten_s2]	10
	L-DRESS	ML Cons: 10 NL Cons: 10	[waistc_s0, sleeph_s0, atc_m04a_s0, knoten_s0, gluc_s_s2, som_huef_s0, som_bmi_s2, asthma_diagnosed_s0, atc_c08da01_s2, atc_c01da_s0, som_huef_s2, som_bmi_s0, genintdoc12m_s0, diab_known_s0, fib_cl_s0, atc_c03ca_s0, som_tail_s0, asthma_untreated_s0]	19

Table 2. List of selected features

The result summary from Table 3 is visualized column-wise in the form of chart in Figure 2, 3, 4, 5 and 6.

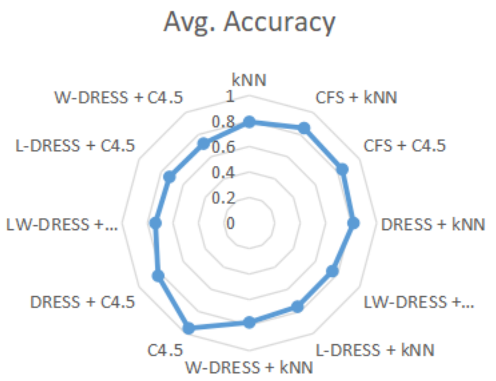

Figure 2: Average Accuracy

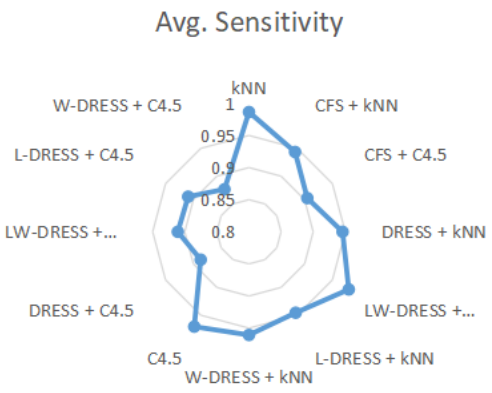

Figure 3: Average Sensitivity

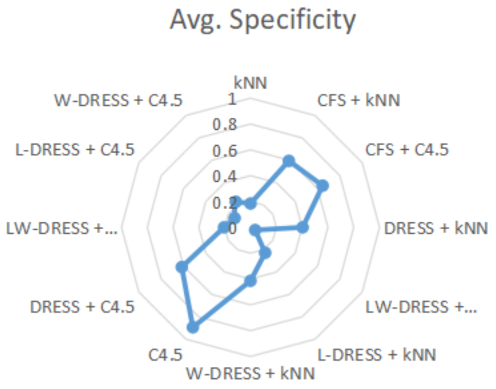

Figure 4: Average Specificity

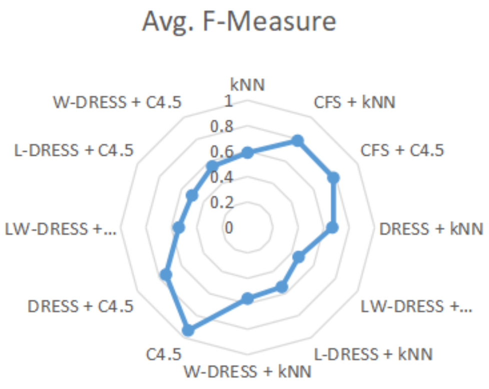

Figure 5: Average F-Measure

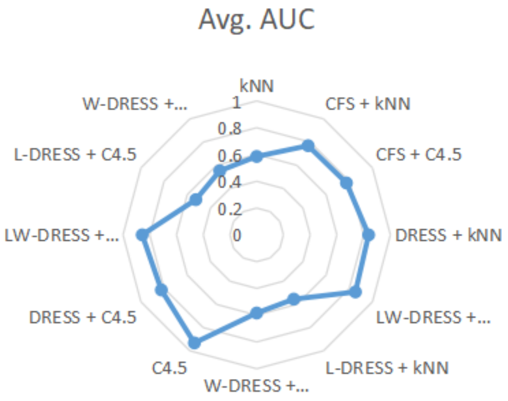

Figure 6: Average AUC Score

Variants	**Avg. Accuracy**	**Avg. Sensitivity**	**Avg. Specificity**	**Avg. AUC**	**Avg. F-Measure**
kNN	*0.79248*	***0.98646***	*0.18447*	*0.58546*	*0.58793*
CFS + kNN	**0.8592**	0.94351	**0.59457**	0.76904	**0.78794**
DRESS + kNN	0.81786	0.94576	0.40379	0.8373	0.6672
LW-DRESS + kNN	0.75300	0.9795	0.0414	**0.85273**	0.46435
L-DRESS + kNN	0.75631	0.94552	0.22732	0.55428	0.53929
W-DRESS + kNN	0.7786	0.96067	0.41379	0.58378	0.5591
C4.5	***0.95195***	***0.97052***	***0.89359***	***0.932056***	***0.93437***
CFS + C4.5	0.84221	0.90489	0.64556	0.77523	0.78073
DRESS + C4.5	0.82694	0.88684	0.6153	0.8234	0.74201
LW-DRESS + C4.5	0.73757	0.91089	0.20538	0.8564	0.54081
L-DRESS + C4.5	0.72378	0.90938	0.14236	0.525874	0.50415
W-DRESS + C4.5	0.71899	0.8764	0.22844	0.55240	0.55403

Table 3. Results of experiments

5.4 Discussion

After running experiments with randomly selected ML (10) and NL (10) constraints for different variants in a 5 fold evaluation process, it was observed that C4.5 performs the best across all measures whereas kNN performs the worst. Average accuracy and average F-Measure for DRESS is about 5-10% higher than the modified variants of it. The modified DRESS variants when combined with kNN as well as C4.5 achieves better average accuracy than CFS and DRESS, but they didn't perform well in terms of average specificity. LW-DRESS attains the best AUC Score for both combinations of kNN and C4.5.

Even though we obtained mixed results in terms of numbers obtained from different measures, we observed that number of feature selected by [7] was constantly higher than other variants namely: W-DRESS, L-DRESS and LW-DRESS. Moreover, LW-DRESS also finds the important features namely *gluc_s_s2, gout_s2, menopaus_yn_s2*, [17] etc. consistently that plays a vital role in correct classification with the SHIP data, which is also mentioned in [7].

6 Conclusion

We presented a modified version of DRESS that selects best features in terms of local structure preservation, constraint satisfaction and cluster quality. Our algorithm makes sure that only the features with most representative power are filtered out. This reduces the feature space which in turn reduces the execution time of the algorithm. The filtered subspaces are further are used to generate candidate subspaces based on constraint satisfaction and weighted distance score between the data instances. The algorithm select only the best available subspace and expands by combining with the rest untill it runs out of subspaces or stops finding subspaces with better scores. The data obtained from experiment shows mixed results but, it constantly selects lesser number of features than DRESS while also selecting the most important ones.

7 Future Work

THE future improvements include the selection of alternative to DBSCAN as clustering algorithm. There is a possibility to take up the quality function as convex optimization problem. We would like to explore other possible semi-supervised feature selection approaches to reduce the feature set instead of laplacian score. As off now, the program takes 2-3 days depending upon the random constraints selected by it. We would also like try to investigate more on reducing the run-time complexity of our algorithm.

8 Acknowledgement

WE would like to thank the IT Adminstrators - *Dipl.-Inform. Frank Zöbisch* and *Dipl.-Inform. Michael Biermann* for helping us in setting up the systems according to the requirements of our project; our guide - *M.Sc. Tommy Hielscher* for his immense support and guidance throughout the different phases of project, and *Prof. Dr. rer. nat. habil. Myra Spiliopoulou* for giving us the opportunity to work on this project.

References

[1] José Luis Balcázar et al. "Machine learning and knowledge discovery in databases". In: *Lecture Notes in Computer Science* 6323 (2010), pp. 204–218.

[2] Hasna Barkia, Haytham Elghazel, and Alex Aussem. "Semi-supervised feature importance evaluation with ensemble learning". In: *Data Mining (ICDM), 2011 IEEE 11th International Conference on.* IEEE. 2011, pp. 31–40.

[3] Khalid Benabdeslem and Mohammed Hindawi. "Constrained laplacian score for semi-supervised feature selection". In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases.* Springer. 2011, pp. 204–218.

[4] Anna L Buczak et al. "A data-driven epidemiological prediction method for dengue outbreaks using local and remote sensing data". In: *BMC medical informatics and decision making* 12.1 (2012), p. 124.

[5] Girish Chandrashekar and Ferat Sahin. "A survey on feature selection methods". In: *Computers & Electrical Engineering* 40.1 (2014), pp. 16–28.

[6] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *Kdd.* Vol. 96. 34. 1996, pp. 226–231.

[7] Tommy Hielscher et al. "Identifying relevant features for a multi-factorial disorder with constraint-based subspace clustering". In: *Computer-Based Medical Systems (CBMS), 2016 IEEE 29th International Symposium on.* IEEE. 2016, pp. 207–212.

[8] Tommy Hielscher et al. "Using participant similarity for the classification of epidemiological data on hepatic steatosis". In: *Computer-Based Medical Systems (CBMS), 2014 IEEE 27th International Symposium on.* IEEE. 2014, pp. 1–7.

[9] Wen-Jyi Hwang and Kuo-Wei Wen. "Fast kNN classification algorithm based on partial distance search". In: *Electronics letters* 34.21 (1998), pp. 2062–2063.

[10] Karin Kailing et al. "Ranking interesting subspaces for clustering high dimensional data". In: *European Conference on Principles of Data Mining and Knowledge Discovery.* Springer. 2003, pp. 241–252.

[11] Mariam Kalakech et al. "Constraint scores for semi-supervised feature selection: A comparative study". In: *Pattern Recognition Letters* 32.5 (2011), pp. 656–665.

[12] Uli Niemann et al. "Learning and inspecting classification rules from longitudinal epidemiological data to identify predictive features on hepatic steatosis". In: *Expert Systems with Applications* 41.11 (2014), pp. 5405–5415.

[13] Lance Parsons, Ehtesham Haque, and Huan Liu. "Subspace clustering for high dimensional data: a review". In: *Acm Sigkdd Explorations Newsletter* 6.1 (2004), pp. 90–105.

[14] Jiangtao Ren et al. “Forward semi-supervised feature selection”. In: *Pacific-Asia conference on knowledge discovery and data mining.* Springer. 2008, pp. 970–976.

[15] Carlos Ruiz, Myra Spiliopoulou, and Ernestina Menasalvas. “C-dbscan: Density-based clustering with constraints”. In: *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing.* Springer. 2007, pp. 216–223.

[16] Henry Völzke et al. “Cohort profile: the study of health in Pomerania”. In: *International journal of epidemiology* 40.2 (2010), pp. 294–307.

[17] Henry Völzke et al. “Menopausal status and hepatic steatosis in a general female population”. In: *Gut* 56.4 (2007), pp. 594–595.

[18] Kiri Wagstaff and Claire Cardie. “Clustering with instance-level constraints”. In: *AAAI/IAAI* 1097 (2000), pp. 577–584.

[19] Tae Keun Yoo and Eun-Cheol Park. “Diabetic retinopathy risk prediction for fundus examination using sparse learning: a cross-sectional study”. In: *BMC medical informatics and decision making* 13.1 (2013), p. 106.