

# DISC: Data-Intensive Similarity Measure for Categorical Data

Aditya Desai, Himanshu Singh and Vikram Pudi

International Institute of Information Technology-Hyderabad,  
Hyderabad, India

{aditya.desai,himanshu.singh}@research.iiit.ac.in, vikram@iiit.ac.in  
<http://iiit.ac.in>

**Abstract.** The concept of similarity is fundamentally important in almost every scientific field. Clustering, distance-based outlier detection, classification, regression and search are major data mining techniques which compute the similarities between instances and hence the choice of a particular similarity measure can turn out to be a major cause of success or failure of the algorithm. The notion of similarity or distance for categorical data is not as straightforward as for continuous data and hence, is a major challenge. This is due to the fact that different values taken by a categorical attribute are not inherently ordered and hence a notion of direct comparison between two categorical values is not possible. In addition, the notion of similarity can differ depending on the particular domain, dataset, or task at hand. In this paper we present a new similarity measure for categorical data DISC - Data-Intensive Similarity Measure for Categorical Data. DISC captures the semantics of the data without any help from domain expert for defining the similarity. In addition to these, it is generic and simple to implement. These desirable features make it a very attractive alternative to existing approaches. Our experimental study compares it with 14 other similarity measures on 24 standard real datasets, out of which 12 are used for classification and 12 for regression, and shows that it is more accurate than all its competitors.

**Keywords:** Categorical similarity measures, cosine similarity, knowledge discovery, classification, regression

## 1 Introduction

The concept of similarity is fundamentally important in almost every scientific field. Clustering, distance-based outlier detection, classification and regression are major data mining techniques which compute the similarities between instances and hence choice of a particular similarity measure can turn out to be a major cause of success or failure of the algorithm. For these tasks, the choice of a similarity measure can be as important as the choice of data representation or feature selection. Most algorithms typically treat the similarity computation as an orthogonal step and can make use of any measure. Similarity measures can be broadly divided in two categories: similarity measures for continuous data and categorical data.

The notion of similarity measure for continuous data is straightforward due to inherent numerical ordering. *Minkowski* distance and its special case, the

*Euclidean* distance are the two most widely used distance measures for continuous data. However, the notion of similarity or distance for categorical data is not as straightforward as for continuous data and hence is a major challenge. This is due to the fact that the different values that a categorical attribute takes are not inherently ordered and hence a notion of direct comparison between two categorical values is not possible. In addition, the notion of similarity can differ depending on the particular domain, dataset, or task at hand.

Although there is no inherent ordering in categorical data, there are other factors like co-occurrence statistics that can be effectively used to define what should be considered more similar and vice-versa. This observation has motivated researchers to come up with data-driven similarity measures for categorical attributes. Such measures take into account the frequency distribution of different attribute values in a given data set but most of these algorithms fail to capture any other feature in the dataset apart from frequency distribution of different attribute values in a given data set. One solution to the problem is to build a common repository of similarity measures for all commonly occurring concepts. As an example, let the similarity values for the concept “color” be determined. Now, consider 3 colors red, pink and black. Consider the two domains as follows:

- Domain I: The domain is say determining the response of cones of the eye to the color, then it is obvious that the cones behave largely similarly to red and pink as compared to black. Hence similarity between red and pink must be high compared to the similarity between red and black or pink and black.
- Domain II: Consider another domain, for example the car sales data. In such a domain, it may be known that the pink cars are extremely rare as compared to red and black cars and hence the similarity between red and black must be larger than that between red and pink or black and pink in this case.

Thus, the notion of similarity varies from one domain to another and hence the assignment of similarity must involve a thorough understanding of the domain. Ideally, the similarity notion is defined by a domain expert who understands the domain concepts well. However, in many applications domain expertise is not available and the users don’t understand the interconnections between objects well enough to formulate exact definitions of similarity or distance.

In the absence of domain expertise it is conceptually very hard to come up with a domain independent solution for similarity. This makes it necessary to define a similarity measure based on latent knowledge available from data instead of a fit-to-all measure and is the major motivation for this paper.

In this paper we present a new similarity measure for categorical data, DISC – Data-Intensive Similarity Measure for Categorical Data. DISC captures the semantics of the data without any help from domain experts for defining similarity. It achieves this by capturing the relationships that are inherent in the data itself, thus making the similarity measure “data-intensive”. In addition, it is generic and simple to implement.

The remainder of the paper is organized as follows. In Section 2 we discuss related work and problem formulation in Section 3. We present the DISC algo-

rithm in Section 4 followed by experimental evaluation and results in Section 5. Finally, in Section 6, we summarize the conclusions of our study and identify future work.

### 1.1 Key Contributions

- Introducing a notion of similarity between two values of a categorical attribute based on co-occurrence statistics.
- Defining a valid similarity measure for capturing such a notion which can be used out-of-the-box for any generic domain.
- Experimentally validating that such a similarity measure provides a significant improvement in accuracy when applied to classification and regression on a wide array of dataset domains. The experimental validation is especially significant since it demonstrates a reasonably large improvement in accuracy by changing only the similarity measure while keeping the algorithm and its parameters constant.

## 2 Related Work

Determining similarity measures for categorical data is a much studied field as there is no explicit notion of ordering among categorical values. Sneath and Sokal were among the first to put together and discuss many of the categorical similarity measures and discuss this in detail in their book [2] on numerical taxonomy.

The specific problem of clustering categorical data has been actively studied. There are several books [3–5] on cluster analysis that discuss the problem of determining similarity between categorical attributes. The problem has also been studied recently in [17, 18]. However, most of these approaches do not offer solutions to the problem discussed in this paper, and the usual recommendation is to “binarize” the data and then use similarity measures designed for binary attributes. Most work has been carried out on development of clustering algorithms and not similarity functions. Hence these works are only marginally or peripherally related to our work. Wilson and Martinez [6] performed a detailed study of heterogeneous distance functions (for categorical and continuous attributes) for instance based learning. The measures in their study are based upon a supervised approach where each data instance has class information in addition to a set of categorical/continuous attributes.

There have been a number of new data mining techniques for categorical data that have been proposed recently. Some of them use notions of similarity which are neighborhood-based [7–9], or incorporate the similarity computation into the learning algorithm [10, 11]. These measures are useful to compute the neighborhood of a point and neighborhood-based measures but not for calculating similarity between a pair of data instances. In the area of information retrieval, Jones et al. [12] and Noreault et. al [13] have studied several similarity measures. Another comparative empirical evaluation for determining similarity between fuzzy sets was performed by Zwick et al. [14], followed by several others [15, 16].

In our experiments we have compared our approach with the methods discussed in [1]; which provides a recent exhaustive comparison of similarity measure for categorical data.

### 3 Problem Formulation

In this section we discuss the necessary conditions for a valid similarity measure. Later, in Section 4.5 we describe how DISC satisfies these requirements and prove the validity of our algorithm. The following conditions need to hold for a distance metric “ $d$ ” to be valid where  $d(x, y)$  is the distance between  $x$  and  $y$ .

1.  $d(x, y) \geq 0$
2.  $d(x, y) = 0$  if and only if  $x=y$
3.  $d(x, y) = d(y, x)$
4.  $d(x, z) \leq d(x, y) + d(y, z)$

In order to come up with conditions for a valid similarity measure we use  $sim = \frac{1}{1+dist}$ , a distance-similarity mapping used in [1]. Based on this mapping we come up with the following definitions for valid similarity measures:

1.  $0 \leq Sim(x, y) \leq 1$
2.  $Sim(x, y) = 1$  if and only if  $x = y$
3.  $Sim(x, y) = Sim(y, x)$
4.  $\frac{1}{Sim(x, y)} + \frac{1}{Sim(y, z)} \geq 1 + \frac{1}{Sim(x, z)}$

Where,  $Sim(x, y)$  is the similarity between  $x$  and  $y$ :

## 4 DISC Algorithm

In this section we present the DISC algorithm. First in Section 4.1 we present the motivation for our algorithm followed by data-structure description in Section 4.2 and a brief overview of the algorithm in Section 4.3. We then describe the algorithm for similarity matrix computation in Section 4.4. Finally in Section 4.5 we validate our similarity measure.

### 4.1 Motivation and Design

As can be seen from the related work, current similarity (distance) measures for categorical data only examine the number of distinct categories and their counts without looking at co-occurrence statistics with other dimensions in the data. Thus, there is a high possibility that, the latent information that comes along is lost during the process of assigning similarities. Consider the example in Table 1, let there be a 3-column dataset where the *Brand* of a car and *Color* are independent variables and the *Price* of the car is a dependant variable. Now there are three brands  $a, b, c$  with average price 49.33, 32.33, 45.66. It can be intuitively said that based on the available information that similarity between  $a$  and  $c$  is greater than that between the categories  $a$  and  $b$ . This is true in real life where  $a, c, b$  may represent low, medium and high end cars and hence the similarity between a low-end and a medium-end car will be more than the similarity between a low-end and a high-end car. Now the other independent variable is *Color*. The average prices corresponding to the three colors namely

Brand	Color	Price
a	red	50
a	green	48
a	blue	50
b	red	32
b	green	30
b	blue	35
c	red	47
c	green	45
c	blue	45

**Table 1.** Illustration

*red*, *green* and *blue* are 43, 41, 43.33. As can be seen, there is a small difference in their prices which is in line with the fact that the cost of the car is very loosely related to its color.

It is important to note that a notion of similarity for categorical variables has a cognitive component to it and as such each one is debatable. However, the above explained notion of similarity is the one that best exploits the latent information for assigning similarity and will hence give predictors of high accuracy. This claim is validated by the experimental results. Extracting these underlying semantics by studying co-occurrence data forms the motivation for the algorithm presented in this section.

## 4.2 Data Structure Description

We first construct a data structure called the categorical information table (CI). The function of the CI table is to provide a quick-lookup for information related to the co-occurrence statistics. Thus, for the above example  $CI[\text{Brand:a}][\text{Color:red}] = 1$ , as for only a single instance, *Brand:a* co-occurs with *Color:red*. For a categorical-numeric pair, e.g.  $CI[\text{Brand:a}][\text{Price}] = 49.33$  the value is the mean value of the attribute *Price* for instances whose value for *Brand* is *a*.

Now, for every value  $v$  of each categorical attribute  $A_k$  a representative point  $\tau(A_k : v)$  is defined. The representative point is a vector consisting of the means of all attributes other than  $A_k$  for instances where attribute  $A_k$  takes value  $v$ :

$$\tau(A_k : v) = \langle \mu(A_k : v, A_1), \dots, \mu(A_k : v, A_d) \rangle \quad (1)$$

It may be noted that the term  $\mu(A_k : v, A_k)$  is skipped in the above expression. As there is no standard notion of *mean* for categorical attributes we define it as

$$\mu(A_k : v, A_i) = \langle CI[A_k : v][A_i : v_{i1}], \dots, CI[A_k : v][A_i : v_{in}] \rangle \quad (2)$$

where  $\text{domain}(A_i) = \{v_{i1}, \dots, v_{in}\}$ . It can thus be seen that, the mean itself is a point in a  $n$ -dimensional space having dimensions as  $v_{i1}, \dots, v_{in}$  with magnitudes:  $\langle CI[A_k : v][A_i : v_{i1}], \dots, CI[A_k : v][A_i : v_{in}] \rangle$ .

Initially all distinct values belonging to the same attribute are conceptually vectors perpendicular to each other and hence the similarity between them is 0.

For, the given example, the mean for dimension *Color* when *Brand : a* is denoted as  $\mu(\text{Brand : a}, \text{Color})$ . As defined above, the mean in a categorical

dimension is itself a point in a  $n$ -dimensional space and hence, the dimensions of mean for the attribute *Color* are *red*, *blue*, *green* and hence

$$\mu(\text{Brand} : a, \text{Color}) = \{CI[\text{Brand} : a][\text{Color} : \text{red}], CI[\text{Brand} : a][\text{Color} : \text{blue}], CI[\text{Brand} : a][\text{Color} : \text{green}]\}$$

$$\text{Similarly, } \mu(\text{Brand} : a, \text{Price}) = \{CI[\text{Brand} : a][\text{Price}]\}$$

Thus the representative point for the value  $a$  of attribute *Brand* is given by,  $\tau(\text{Brand} : a) = \langle \mu(\text{Brand} : a, \text{Color}), \mu(\text{Brand} : a, \text{Price}) \rangle$

### 4.3 Algorithm Overview

Initially we calculate the representative points for all values of all attributes. We then initialize similarity in a manner similar to the overlap similarity measure where matches are assigned similarity value 1 and the mismatches are assigned similarity value 0. Using the representative points calculated above, we assign a new similarity between each pair of values  $v, v'$  belonging to the same attribute  $A_k$  as equal to the average of cosine similarity between their means for each dimension. Now the cosine similarity between  $v$  and  $v'$  in dimension  $A_i$  is denoted by  $CS(v : v', A_i)$  and is equal to the cosine similarity between vectors  $\mu(A_k : v, A_i)$  and  $\mu(A_k : v', A_i)$ . Thus, similarity between  $A_k : v$  and  $A_k : v'$  is:

$$\frac{\sum_{l=0, l \neq i}^d CS(v : v', A_l)}{d - 1}$$

Thus, for the above example, the similarity between *Brand:a* and *Brand:b* is the average of cosine similarity between their respective means in dimensions *Color* and *Price*. Thus  $Sim(a, b)$  is given as:  $\frac{CS(a:b, \text{Color}) + CS(a:b, \text{Price})}{2}$

An iteration is said to have been completed, when similarity between all pairs of values belonging to the same attribute (for all attributes) are computed using the above methodology. These, new values are used for cosine similarity computation in the next iteration.

### 4.4 DISC Computation

In this section, we describe the DISC algorithm and hence the similarity matrix construction. The similarity matrix construction using DISC is described as follows:

1. The similarity matrix is initialized in a manner similar to overlap similarity measure where  $\forall_{i,j,k} Sim(v_{ij}, v_{ik}) = 1$ , if  $v_{ij} = v_{ik}$  and  $Sim(v_{ij}, v_{ik}) = 0$ , if  $v_{ij} \neq v_{ik}$
2. Consider a training dataset to be consisting of  $n$  tuples. The value of the feature variable  $A_j$  corresponding to the  $i^{th}$  tuple is given as  $Train_{ij}$ . We construct a data-structure ‘‘Categorical Information’’ which for any categorical value ( $v_{il}$ ) of attribute  $A_i$  returns number of co-occurrences of value  $v_{jk}$  taken by feature variable  $A_j$  if  $A_j$  is categorical and returns the mean value of feature variable  $A_j$  for the corresponding set of instances if it is numeric. Let this data-structure be denoted by  $CI$ . The value corresponding to the number of co-occurrences of categorical value  $v_{jk}$  when feature variable  $A_i$  takes value  $v_{il}$  is given by  $CI[A_i, v_{il}][A_j, v_{jk}]$  when  $A_j$  is categorical. Also, when  $A_j$  is numeric,  $CI[A_i, v_{il}][A_j]$  corresponds to the mean of values taken by attribute  $A_j$  when  $A_i$  takes value  $v_{il}$ .

3. The  $\text{Sim}(v_{ij}, v_{ik})$  (Similarity between categorical values  $v_{ij}$  and  $v_{ik}$ ) is now calculated as the average of the per-attribute cosine similarity between their means ( $\text{Similarity}_m$ ), where the means have a form as described above. The complicated nature of cosine product arises due to the fact that, the transformed space after the first iteration has dimensions which are no longer orthogonal (i.e.  $\text{Sim}(v_{ij}, v_{ik})$  is no longer 0).

---


$$\text{Similarity}_m = \begin{cases} 1 - \frac{|CI[A_i:v_{ij}][A_m] - CI[A_i:v_{ik}][A_m]|}{\text{Max}[A_m] - \text{Min}[A_m]}; & \text{if } A_m \text{ is Numeric} \\ CS(\mu(A_i : v_{ij}, A_m), \mu(A_i : v_{ik}, A_m)); & \text{if } A_m \text{ is Categorical} \end{cases}$$

where  $CS(\mu(A_i : v_{ij}, A_m), \mu(A_i : v_{ik}, A_m))$  is defined as follows :

$$\frac{\sum_{v_{ml}, v_{m\bar{l}} \in A_m} CI[A_i : v_{ij}][A_m : v_{ml}] * CI[A_i : v_{ik}][A_m : v_{m\bar{l}}] * \text{Sim}(v_{m\bar{l}}, v_{ml})}{\text{NormalVector1} * \text{NormalVector2}}$$

$$\begin{aligned} \text{NormalVector1} &= (\sum_{v_{ml}, v_{m\bar{l}} \in A_m} CI[A_i : v_{ij}][A_m : v_{ml}] * CI[A_i : v_{ij}][A_m, v_{m\bar{l}}] * \text{Sim}(v_{ml}, v_{m\bar{l}}))^{1/2} \\ \text{NormalVector2} &= (\sum_{v_{ml}, v_{m\bar{l}} \in A_m} CI[A_i : v_{ik}][A_m : v_{ml}] * CI[A_i : v_{ik}][A_m, v_{m\bar{l}}] * \text{Sim}(v_{ml}, v_{m\bar{l}}))^{1/2} \\ \text{Sim}(v_{ij}, v_{ik}) &= \frac{1}{d-1} \sum_{m=1, m \neq i}^d \text{Similarity}_m \end{aligned}$$


---

**Table 2.** Cosine Similarity computation between  $v_{ij}, v_{ik}$

4. The matrix is populated using the above equation for all combinations  $\forall_{i,j,k} \text{Sim}(v_{ij}, v_{ik})$ . To test the effectiveness of the similarity matrix, we plug the similarity values in a classifier (the nearest neighbour classifier in our case) in case of classification and compute its accuracy on a validation set. If the problem domain is regression we plug in the similarity values into a regressor (the nearest neighbour regressor in our case) and compute the corresponding root mean square error. Thus, such an execution of 3 followed by 4 is termed an iteration.
5. The step 3 is iterated on again using the new similarity values until the accuracy parameter stops increasing. The matrix obtained at this iteration is the final similarity matrix that is used for testing. (In case of regression we stop when the root mean square error increases.)
- In addition, the authors have observed that, most of the improvement takes place in the first iteration and hence in domains like clustering (unsupervised) or in domains with tight limits on training time the algorithm can be halted after the first iteration.

#### 4.5 Validity of Similarity Measure

The similarity measure proposed in this paper is basically a mean of cosine similarities derived for individual dimensions in non-orthogonal spaces. The validity of the similarity measure can now be argued as follows:

1. As the similarity measure is a mean of cosine similarities which have a range from 0-1, it is implied that the range of values output by the similarity measure will be between 0-1 thus satisfying the first constraint.
2. For the proposed similarity measure  $Sim(X, Y) = 1$ , if and only if  $Sim_k(X_k, Y_k) = 1$  for all feature variables  $A_k$ . Now, constraint 2 will be violated if  $X \neq Y$  and  $Sim(X, Y) = 1$ . This implies that there exists an  $X_k, Y_k$  such that  $X_k \neq Y_k$  and for which  $Sim(X_k, Y_k) = 1$ . Now for  $Sim(X_k, Y_k) = 1$  implies cosine product of  $CI[A_k : X_k][A_m]$  and  $CI[A_k : Y_k][A_m]$  is 1 for all  $A_m$  which implies that  $CI[A_k : X_k][A_m]$ ,  $CI[A_k : Y_k][A_m]$  are parallel and hence can be considered to be equivalent with respect to the training data.
3. As cosine product is commutative, the third property holds implicitly.
4. It may be noted that the resultant similarity is a mean of similarities computed for each dimension. Also, the similarity for each dimension is in essence a cosine product and hence, the triangle inequality holds for each component of the sum. Thus the fourth property is satisfied.

## 5 Experimental Study

In this section, we describe the pre-processing steps and the datasets used in Section 5.1 followed by experimental results in Section 5.2. Finally in Section 5.3 we provide a discussion on the experimental results.

### 5.1 Pre-Processing and Experimental Settings

For our experiments we used 24 datasets out of which 12 were used for classification and 12 for regression. We compare our approach with the approaches discussed in [1], which provides a recent exhaustive comparison of similarity measures for categorical data.

Eleven of the datasets used for classification were purely categorical and one was numeric (Iris). Different methods can be used to handle numeric attributes in datasets like discretizing the numeric variables using the concept of minimum description length [20] or equi-width binning. Another possible way to handle a mixture of attributes is to compute the similarity for continuous and categorical attributes separately, and then do a weighted aggregation. For our experiments we used MDL for discretizing numeric variables for classification datasets.

Nine of the datasets used for regression were purely numeric, two (Abalone and Auto Mpg) were mixed while one (Servo) was purely categorical. It may be noted that the datasets used for regression were discretized using equi-width binning using the following weka setting: *“weka.filters.unsupervised.attribute.Discretize - B10 - M - 1.0 - Rfirst - last”* The  $k$ -Nearest Neighbours ( $kNN$ ) was implemented with number of neighbours 10. The weight associated with each neighbour was the similarity between the neighbour and the input tuple. The class with the highest weighted votes was the output class for classification while the output for regression was a weighted sum of the individual responses.

The results have been presented for 10-folds cross-validation. Also, for our experiments we used the entire train set as the validation set. The numbers in brackets indicate the rank of DISC versus all other competing similarity measures. For classification, the values indicate the accuracy of the classifier where a

high value corresponds to high percentage accuracy and hence such a similarity measure is assigned a better (higher) rank. On the other hand, for regression Root Mean Square Error (RMSE) value has been presented where a comparatively low value indicates lower error and better performance of the predictor and hence such a similarity measure is assigned a better rank. It may be noted that a rank of 1 indicates best performance with the relative performance being poorer for lower ranks.

## 5.2 Experimental Results

The experimental results for classification and regression are presented in Table 3, 4 and Table 5, 6 respectively. In these tables each row represents competing similarity measure and the column represents different datasets. In Table 3 and 4, each cell represents the accuracy for the corresponding dataset and similarity measure respectively. In Table 5 and 6, each cell represents the root mean square error (RMSE) for the corresponding dataset and similarity measure respectively.

Dataset Sim. Measure	Balance	Breast Cancer	Car Evaluation	Hayes Roth	Iris	Lymphog- raphy
DISC	90.4(1)	76.89(1)	96.46(1)	77.27(1)	96.66(1)	85.13(3)
Overlap	81.92	75.81	92.7	64.39	96.66	81.75
Eskin	81.92	73.28	91.2	22.72	96.0	79.72
IOF	79.84	76.89	91.03	63.63	96.0	81.75
OF	81.92	74.0	90.85	17.42	95.33	79.05
Lin	81.92	74.72	92.7	71.96	95.33	84.45
Lin1	81.92	75.09	90.85	18.93	94.0	82.43
Goodall1	81.92	74.36	90.85	72.72	95.33	86.48
Goodall2	81.92	73.28	90.85	59.09	96.66	81.08
Goodall3	81.92	73.64	90.85	39.39	95.33	85.13
Goodall4	81.92	74.72	91.03	53.78	96.0	81.08
Smirnov	81.92	71.48	90.85	59.84	94.0	85.81
Gambaryan	81.92	76.53	91.03	53.03	96.0	82.43
Burnaby	81.92	70.39	90.85	19.69	95.33	75.0
Anderberg	81.92	72.2	90.85	25.0	94.0	80.4

**Table 3.** Accuracy for  $k$ -NN with  $k = 10$

## 5.3 Discussion of Results

As can be seen from the experimental results, DISC is the best similarity measure for classification for all datasets except Lymphography, Primary Tumor and Hayes Roth Test where it is the third best for the first two and the second best for the last one. On the basis of overall mean accuracy, DISC outperforms the nearest competitor by about 2.87% where we define overall mean accuracy as the mean of accuracies over all classification datasets considered for our experiments. For regression, DISC is the best performing similarity measure on the basis of Root Mean Square Error (RMSE) for all datasets.

Dataset Sim. Measure	Primary Tumor	Hayes Roth Test	Tic Tac Toe	Zoo	Teaching Assist.	Nursery	Mean Accuracy
DISC	41.66(3)	89.28(2)	100.0(1)	91.08(1)	58.94(1)	98.41(1)	83.51(1)
Overlap	41.66	82.14	92.48	91.08	50.33	94.75	78.81
Eskin	41.36	75.0	94.46	90.09	50.33	94.16	74.19
IOF	38.98	71.42	100.0	90.09	47.01	94.16	77.57
OF	40.17	60.71	84.96	89.1	43.7	95.74	71.08
Lin	41.66	67.85	95.82	90.09	56.95	96.04	79.13
Lin1	42.26	42.85	82.56	91.08	54.96	93.54	70.87
Goodall1	43.15	89.28	97.07	89.1	51.65	95.74	80.64
Goodall2	38.09	92.85	91.54	88.11	52.98	95.74	78.52
Goodall3	41.66	71.42	95.51	89.1	50.99	95.74	75.89
Goodall4	32.73	82.14	96.24	89.1	55.62	94.16	77.38
Smirnov	42.55	78.57	98.74	89.1	54.3	95.67	78.57
Gambaryan	39.58	89.28	98.74	90.09	50.33	94.16	78.59
Burnaby	3.86	60.71	83.29	71.28	40.39	90.85	65.30
Anderberg	37.79	53.57	89.14	90.09	50.33	95.74	71.75

**Table 4.** Accuracy for  $k$ -NN with  $k = 10$ 

Dataset	Comp. Strength	Flow	Abalone	Bodyfat	Housing	Whitewine
DISC	4.82(1)	13.2(1)	2.4(1)	0.6(1)	4.68(1)	0.74(1)
Overlap	6.3	15.16	2.44	0.65	5.4	0.74
Eskin	6.58	16.0	2.45	0.66	6.0	0.77
IOF	6.18	15.53	2.42	0.76	5.48	0.75
OF	6.62	14.93	2.41	0.66	5.27	0.75
Lin	6.03	16.12	2.4	0.63	5.3	0.74
Lin1	7.3	16.52	2.41	0.87	5.41	0.74
Goodall1	6.66	14.97	2.41	0.64	5.27	0.74
Goodall2	6.37	15.09	2.43	0.66	5.33	0.75
Goodall3	6.71	14.96	2.41	0.65	5.27	0.74
Goodall4	5.98	15.67	2.47	0.71	6.4	0.78
Smirnov	6.89	15.5	2.4	0.67	5.17	0.74
Gambaryan	6.01	15.46	2.46	0.67	5.73	0.76
Burnaby	6.63	15.23	2.41	0.65	5.32	0.74
Anderberg	7.15	15.16	2.42	0.67	5.84	0.75

**Table 5.** RMSE for  $k$ -NN with  $k = 10$ 

For classification datasets like Iris, Primary Tumor and Zoo the algorithm halted after the 1<sup>st</sup> iteration while for datasets like Balance, Lymphography, Tic-Tac-Toe, Breast Cancer the algorithm halted after the 2<sup>nd</sup> iteration. Also, for Car-Evaluation, Hayes Roth, Teaching Assistant and Nursery the algorithm halted after the 3<sup>rd</sup> iteration while it halted after the 4<sup>th</sup> iteration for Hayes Roth Test. For regression, the number of iterations was less than 5 for all datasets except Compressive Strength for which it was 9. Thus, it can be seen that the number of iterations for all datasets is small. Also, the authors observed that the major bulk of the accuracy improvement is achieved with the first iteration

Dataset	Slump	Servo	Redwine	Forest Fires	Concrete	Auto Mpg
DISC	6.79(1)	0.54(1)	0.65(1)	65.96(1)	10.29(1)	2.96(1)
Overlap	7.9	0.78	0.67	67.13	11.61	3.58
Eskin	8.12	0.77	0.68	67.49	11.15	3.98
IOF	8.11	0.77	0.68	67.95	11.36	3.71
OF	7.72	0.8	0.68	67.76	12.55	3.3
Lin	8.33	0.76	0.67	67.16	10.99	3.74
Lin1	8.42	1.1	0.68	67.96	12.16	3.89
Goodall1	7.76	0.77	0.66	67.97	11.45	3.5
Goodall2	7.82	0.81	0.67	68.64	12.21	3.39
Goodall3	7.75	0.78	0.67	68.48	11.52	3.39
Goodall4	7.87	0.95	0.71	70.28	12.96	3.92
Smirnov	8.22	0.78	0.69	67.07	11.59	3.39
Gambaryan	7.8	0.83	0.69	69.54	12.38	3.75
Burnaby	7.89	0.8	0.68	67.73	12.62	3.28
Anderberg	7.94	0.9	0.7	66.63	12.66	3.53

**Table 6.** RMSE for  $k$ -NN with  $k = 10$ 

and hence for domains with time constraints in training the algorithm can be halted after the first iteration. The reason for the consistently good performance can be attributed to the fact that a similarity computation is a major component in nearest neighbour classification and regression techniques, and DISC captures similarity accurately and efficiently in a data driven manner.

The computational complexity for determining the similarity measure is equivalent to the computational complexity of computing cosine similarity for each pair of values belonging to the same categorical attribute. Let the number of pairs of values, the number of tuples, number of attributes and the average number of values per attribute be  $V$ ,  $n$ ,  $d$  and  $v$  respectively. It can be seen that, construction of categorical collection is  $O(nd)$ . Also, for all pairs of values  $V$ , we compute the similarity as the mean of cosine similarity of their representative points for each dimension. This is essentially  $(v^2d)$  for each pair and hence the computationally complexity is  $O(Vv^2d)$  and hence the overall complexity is  $O(nd + Vv^2d)$ . Once, the similarity values are computed, using them in any classification, regression or a clustering task is a simple table look up and is hence  $O(1)$ .

## 6 Conclusion

In this paper we have presented and evaluated DISC, a similarity measure for categorical data. DISC is data intensive, generic and simple to implement. In addition to these features, it doesn't require any domain expert's knowledge. Finally our algorithm was evaluated against 14 competing algorithms on 24 standard real-life datasets, out of which 12 were used for classification and 12 for regression. It outperformed all competing algorithms on almost all datasets. The experimental results are especially significant since it demonstrates a reasonably large improvement in accuracy by changing only the similarity measure while keeping the algorithm and its parameters constant.

Apart from classification and regression, similarity computation is a pivotal step in a number of application such as clustering, distance-based outliers detection and search. Future work includes applying our algorithm for these techniques also. We also intend to develop a weighing measure for different dimensions for calculating similarity which will make the algorithm more robust.

## References

1. Shyam Boriah, Varun Chandola and Vipin Kumar. Similarity Measures for Categorical Data: A Comparative Evaluation. *Proceedings of SDM '08*, Atlanta, Georgia, USA. SIAM, 2008.
2. P. H. A. Sneath and R. R. Sokal. *Numerical Taxonomy: The Principles and Practice of Numerical Classification*. W. H. Freeman and Company, San Francisco, 1973.
3. M. R. Anderberg. *Cluster Analysis for Applications*. Academic Press, 1973.
4. A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, Englewood Cliffs NJ, 1988.
5. J. A. Hartigan. *Clustering Algorithms*. John Wiley & Sons, New York, NY, 1975.
6. D. R. Wilson and T. R. Martinez. Improved heterogeneous distance functions. *J. Artif. Intell. Res. (JAIR)*, 6:1–34, 1997.
7. Y. Biberman. A context similarity measure. In *ECML '94*, pages 49–63. Springer, 1994.
8. G. Das and H. Mannila. Context-based similarity measures for categorical databases. In *PKDD '00*, London, UK, 2000. Springer-Verlag.
9. C. R. Palmer and C. Faloutsos. Electricity based external similarity of categorical attributes. In *PAKDD '03* pages 486–500. Springer, 2003.
10. Z. Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304, 1998.
11. V. Ganti, J. Gehrke, and R. Ramakrishnan. CACTUS—clustering categorical data using summaries. In *KDD '99*, New York, NY, USA, 1999. ACM Press
12. W. P. Jones and G. W. Furnas. Pictures of relevance: a geometric analysis of similarity measures. *J. Am. Soc. Inf. Sci.*, 38(6):420–442, 1987.
13. T. Noreault, M. McGill, and M. B. Koll. A performance evaluation of similarity measures, document term weighting schemes and representations in a boolean environment. In *SIGIR '80: Proceedings of the 3rd annual ACM conference on Research and development in information retrieval*, pages 57–76, Kent, UK, 1981. Butterworth & Co.
14. R. Zwick, E. Carlstein, and D. V. Budesu. Measures of similarity among fuzzy concepts: A comparative analysis. *International Journal of Approximate Reasoning*, 1(2):221–242, 1987.
15. C. P. Pappis and N. I. Karacapilidis. A comparative assessment of measures of similarity of fuzzy values. *Fuzzy Sets and Systems*, 56(2):171–174, 1993.
16. X. Wang, B. De Baets, and E. Kerre. A comparative study of similarity measures. *Fuzzy Sets and Systems*, 73(2):259–268, 1995.
17. D. Gibson, J. M. Kleinberg, and P. Raghavan. Clustering categorical data: An approach based on dynamical systems. *VLDB Journal*, 8(34):222–236, 2000.
18. S. Guha, R. Rastogi, and K. Shim. ROCK—a robust clustering algorithm for categorical attributes. In *Proceedings of IEEE International Conference on Data Engineering*, 1999.
19. I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2 edition, 2005.
20. U. M. Fayyad and K. B. Irani. On the handling of continuous-valued attributes in decision tree generation. *Machine Learning*, 8:87–102, 1992.