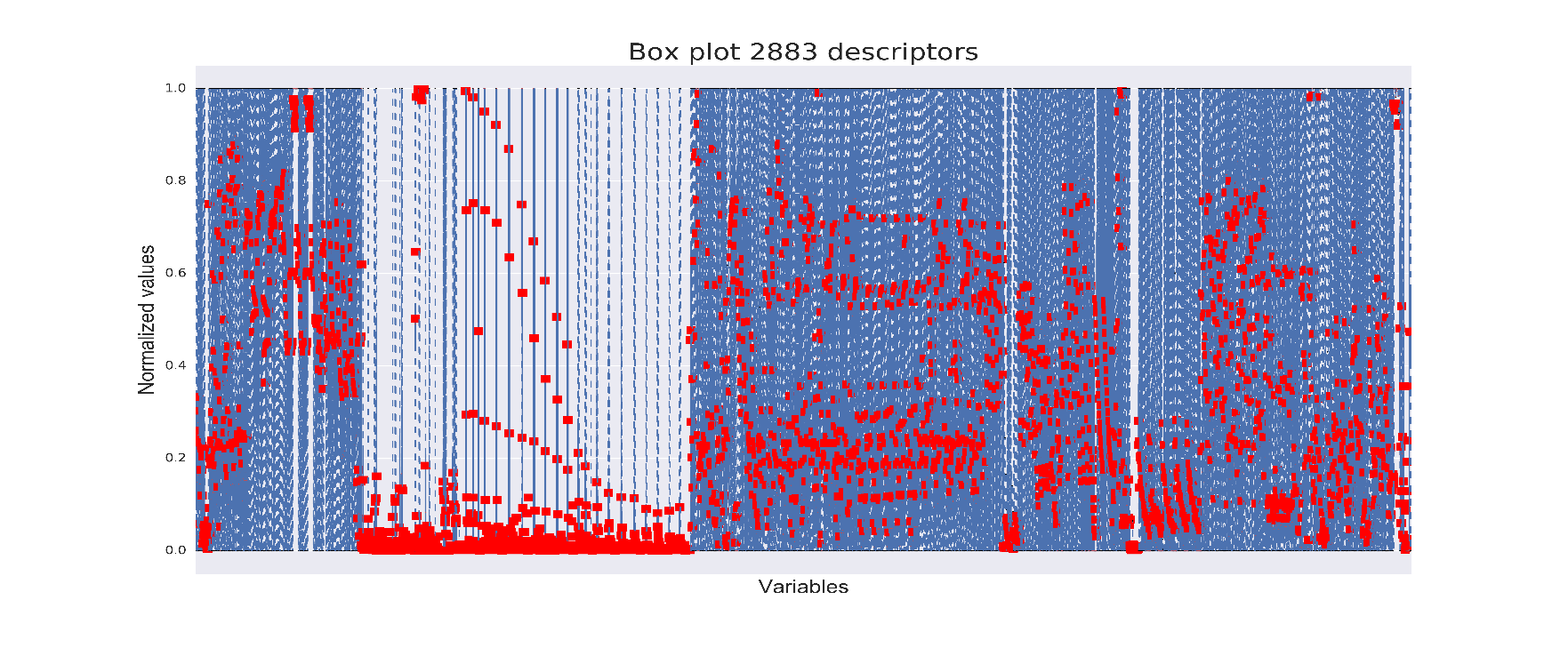
**Supplementary Information**

**Data Analysis and Machine Learning Techniques for predicting Glass Transition Temperatures with Assembled Homopolymers Data**

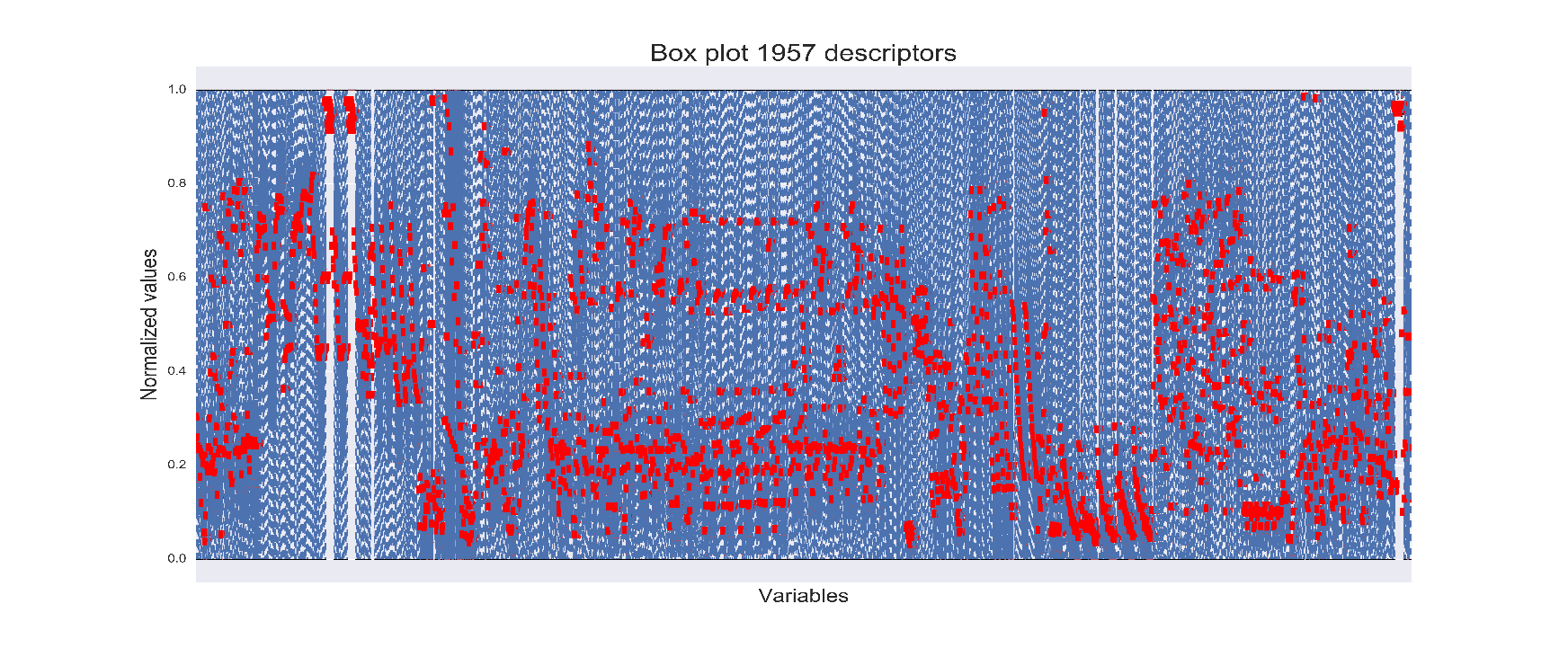
Gerardo M. Casanola-Martin1†, Anas Karuth1†, Hai Pham-The2, Humbert González-Díaz3,4,5 Dean C. Webster1, Bakhtiyor Rasulev1\*.

**Table of Contents**

|  |  |
| --- | --- |
| **Figure S1** Box plot of the 2883 features. ……………………………….................. | **2** |
| **Figure S2**. Box plot of the 1957 features non- near-constant and non-zero values... | **2** |
| **Section S3.** Fundamental Remarks on Statistical and Machine Learning Techniques | **3** |
| **Figure S4.** Density distribution of the descriptors values for each variable in a violin plot | **7** |



**Figure S1**. Box plot of the 2883 features (Plot generated using matplotlib and seaborn library of Python).



**Figure S2**. Box plot of the 1957 features non near-constant values and zero values (Plot generated using matplotlib and seaborn library of Python).

**S3.** Fundamental Remarks on Statistical and Machine Learning Techniques

**SVM:** Is asupervised machine learning technique for learning classification that has become of common use in many fields in the last times, and their area of influence continues to expand today 1. SVMs are based on the concept of decision hyperplanes that define decision boundaries. A decision hyperplane is one with a maximum *margin* that separates between a set of objects having different class memberships. In the case of training SVM2 In the case of c-SVM non-linear, give a training set of input pairs (xi,yi), i=1,2,…,l where xi Rn and y 1l, training SVM involves the minimization of the error function:

  (1)

subject to  (2)



where C is the capacity constant that can be adjusted by the user and can either increase or decrease the penalty for classification errors, w is the vector of coefficients, b is a constant and are parameters for handling non-separable data (inputs). The function is used to transform (map) data from the input (independent) to the feature space. is called the kernel function. It should be noted that the larger the C, the more the error is penalized, so decrease the misclassified patterns while the smaller the C, the more the margin maximizes, so the optimal separating hyperplane is less sensitive to the errors from the learning set. It is not known beforehand which C is the best for one problem. Thus, an optimum value of C is very important and can be selected for each case using cross validation. Another aspect very important in training SVM is the selection of appropriate kernel function and optimizing its parameters. The functionis used to transform (map) data from the input (independent) to the feature space and is called the kernel function. It should be noted that the larger the C, the more the error is penalized, so decrease the misclassified patterns while the smaller the C, the more the margin maximizes, so the optimal separating hyperplane is less sensitive to the errors from the learning set. Thus, an optimum value of C is very important. another aspect very important in training SVM is the selection of appropriate kernel function and optimizing its *parameters.* The most common kernels are: linear, polynomial, radial basis function (RBF), Gaussian Radial basis function and sigmoid.3, 4 More details can be revised and consulted in references 5, 6.

In our case the radial basis function (RBF) kernel was used, and the experiments were set up with gamma = 0.1 to 5 in 0.1 intervals. The different C values were also proved from 1 – 100 in 10 units range (10, 20, 30, etc.). The best model showed a value of gamma = 5 and a C =1. All parameters from this model depicted good performances.

**KNN:** This algorithm7 is the greatest direct classifier of machine learning techniques and is used in several studies.8-10 This method was developed from the need to perform discriminant analysis when reliable parametric estimates of probability densities are unknown or difficult to determine 11. In this technique the distance metric between an unclassified vector q (query case) and each individual vector xi in the training set is measured.12 K-NN accomplish the outcome of this unclassified vector by finding k vectors that are closest in distance to it. The class of the majority of the k nearest neighbors is chosen as the predicted class of the unclassified vector q. There are two stages on k-NN approach: first is the determination of k (training), then using determined value of k together with the training example to make membership predictions of new case. In the training state, we need to define a metric for measuring the distance between the query point and cases from the examples sample. In WEKA this approach is called instance-based learning (IBL) algorithm. For the current study several combinations of hyperparameters were proved as follow: the number of nearest neighbors (k) from 1-20 (1 unit step), the nearest neighbor search algorithm, in our case, we use two of the better implemented in WEKA (LinearNNSearch, KDTree). For the LinearNNSearch, the four distances were used (ChebyshevDistance, EditDistance, EuclideanDistance and ManhattanDistance. For the KDTree the different splitting method to the nodes were explored. In this search algorithm KDTree the max number of instances in a leaf was also a parameter used with values from 5 to 40 (5, 10, 15, 20, 25, 30, 35, 40). All these combinations proved the data set and resulted in 360 combinations. The best model encountered was based in LinearNNSearch with Manhattan as distance measure and with k=3.

**RF:** Random Forest is an extension of classification and regression trees (CART). They perform well even in the presence of a large number of features and a small number of observations 13, 14. In Random Forest, each tree is assembled using a different bootstrap sample, each node is split using the best features among a subset of predictors randomly chosen at that node. This process is fast even for big data sets with successful results where the variable importance and reduction of the number of features can be measured. Is important to stand out that although random forests perform well in many applications, their theoretical properties are not fully understood 15.

In the case of the random forest, three main parameters were varied. The number of trees (I) from 10 to 300 in 1 step intervals. Another value taken into consideration was the number of features (K) proved from 1-20 ranging in 1-unit steps. Besides the depth of the forest (D) was assessed from 1-10. The QSPR-RF model with the best performance results showed the following hyperparameters values: depth =6, I= 220 and K=13.

**MLP**: The best ANN based-model was obtained using WEKA with the sigmoidal function as activation function. The number of hidden nodes was selected by “trial and error” strategy ranking the results of performances, with a total 864 combinations evaluated in the training set. The ANN with the best configuration was a 15-3-1 ANN.

**GPR**: A Gaussian Process Regressor (GPR) is a type of machine learning algorithm used for regression problems. GPR is a non-parametric method, meaning that it doesn't assume a specific functional form for the relationship between the input variables and the output variables. Instead, it models the relationship as a distribution over functions. One of the parameters taking into consideration was the appropriate level of Gaussian noise, as a factor that an affect the performance of models trained on the data, because this take into account the noise level in the real-world data and the desired level of model complexity. In our case, the best GPR model in WEKA was obtained with the RBFKernel and a gamma value equal to 0.3, a noise level value as 1, and after an iteration over 735 different hyperparameter combinations.

(1) Vapnik, V. *The Nature of Statistical Learning Theory*; Springer, 1995.

(2) Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273-297.

(3) Ivanciuc, O. Applications of Support Vector Machines in Chemistry. In *Reviews in Computational Chemistry*, Lipkowitz, K. B., Cundari, T. R. Eds.; Vol. 23; Wiley-VCH, 2007; pp 291-400.

(4) Hsu, C.-W.; Chang, C.-C.; Lin, C.-J. A Practical Guide to Support Vector Classification. In [*http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf*](http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf).

(5) Christianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines*; Cambridge University Press, 2000.

(6) Burges, C. A tutorial on support vector machines for pattern recognition. In *Data Miningand Knowledge Discovery*, Vol. 2; Kluwer Academic Publishers, 1998; pp 1 - 43.

(7) Fukunaga, K. *Introduction to statistical pattern recognition*; Academic Press Professional, Inc., 1990.

(8) Abidin, T.; Perrizo, W. SMART-TV: A Fast and Scalable Nearest Neighbor Based Classifier for Data Mining. In *Proceedings of ACM SAC-06*, Dijon, France, April 23-27, 2006; ACM Press, New York: pp 536-540.

(9) Khan, M.; Ding, Q.; Perrizo, W. K-Nearest Neighbors Classification of Spatial Data Streams using P-trees. In *Proceedings of the PAKDD*, 2002; pp 517-528.

(10) Yu, K.; Ji, L. Karyotyping of Comparative Genomic Hybridization Human Metaphases Using Kernel Nearest-Neighbor Algorithm. *Cytometry* **2002**, *48*, 202-208.

(11) Peterson, L. E. K-nearest neighbor. *Scholarpedia* **2009**, *4* (2), 1883. DOI: 10.4249/scholarpedia.1883.

(12) Fix, E.; Hodges, J. L. *Discriminatory analysis: Non-parametric discrimination*; USAF School of Aviation Medicine, 1951.

(13) Ziegler, A.; König, I. R. Mining data with random forests: Current options for real-world applications. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **2014**, *4* (1), 55-63. DOI: 10.1002/widm.1114.

(14) Genuer, R.; Poggi, J.-M.; Tuleau-Malot, C. Variable selection using random forests. *Pattern Recognition Letters* **2010**, *31* (14), 2225-2236. DOI: <http://dx.doi.org/10.1016/j.patrec.2010.03.014>.

(15) Tripoliti, E. E.; Fotiadis, D. I.; Manis, G. Modifications of the construction and voting mechanisms of the Random Forests Algorithm. *Data and Knowledge Engineering* **2013**, *87*, 41-65. DOI: 10.1016/j.datak.2013.07.002.

A group of blue shapes

Description automatically generated

**Figure S4.** Density distribution of the descriptors values for each variable (Plot generated using matplotlib and seaborn library of Python).