data

MDPI

Data Descriptor

The Fundamental Clustering and Projection Suite (FCPS): A Dataset Collection to Test the Performance of Clustering and Data Projection Algorithms

Alfred Ultsch [1] and Jörn Lötsch [2,3,*]

[1] DataBionics Research Institute, University of Marburg, 35032 Marburg, Germany; ultsch@informatik.uni-marburg.de
[2] Institute of Clinical Pharmacology, Goethe - University, 60590 Frankfurt am Main, Germany
[3] Fraunhofer Institute for Molecular Biology and Applied Ecology IME, Project Group Translational Medicine and Pharmacology TMP, 60590 Frankfurt am Main, Germany
* Correspondence: j.loetsch@em.uni-frankfurt.de

Received: 27 December 2019; Accepted: 28 January 2020; Published: 30 January 2020

Abstract: In the context of data science, data projection and clustering are common procedures. The chosen analysis method is crucial to avoid faulty pattern recognition. It is therefore necessary to know the properties and especially the limitations of projection and clustering algorithms. This report describes a collection of datasets that are grouped together in the Fundamental Clustering and Projection Suite (FCPS). The FCPS contains 10 datasets with the names "Atom", "Chainlink", "EngyTime", "Golfball", "Hepta", "Lsun", "Target", "Tetra", "TwoDiamonds", and "WingNut". Common clustering methods occasionally identified non-existent clusters or assigned data points to the wrong clusters in the FCPS suite. Likewise, common data projection methods could only partially reproduce the data structure correctly on a two-dimensional plane. In conclusion, the FCPS dataset collection addresses general challenges for clustering and projection algorithms such as lack of linear separability, different or small inner class spacing, classes defined by data density rather than data spacing, no cluster structure at all, outliers, or classes that are in contact. This report describes a collection of datasets that are grouped together in the Fundamental Clustering and Projection Suite (FCPS). It is designed to address specific problems of structure discovery in high-dimensional spaces.

Dataset: Available as a supplementary file in this submission. link http://www.mdpi.com/2306-5729/5/1/13/s1.

Keywords: clustering; data projection; performance tests; benchmark standards; high dimensional complex data

1. Summary

The exploration of high-dimensional data spaces is a challenge. Starting from four dimensions, high dimensions become increasingly incomprehensible and the everyday experience of spatial relationships between data points is replaced by strange phenomena for which the term "curse of dimensionality" was coined [1]. It can be shown that as dimensionality increases, "space" is concentrated on an ever smaller hypersurface [2]. The difference of the largest and smallest distances disappears with increasing dimensionality. Furthermore, it is clear that high-dimensional spaces generally do not fit into spaces of lower dimensionality, that is, projections from high-dimensional spaces onto $\mathbb{R}^2$, or $\mathbb{R}^3$ must make errors.

Projections into the $\mathbb{R}^2$ provide visualizations that improve the search for groups in the data that have common properties, that is, improve clustering. This is essential for the exploration of complex and large data. Due to the properties of high-dimensional spaces, these algorithms must overcome unavoidable difficulties. In the context of data science and especially data mining and knowledge discovery, this being the attempt to find new and previously unknown structures in high-dimensional data, it is necessary to know the properties and especially the limits of projection and clustering algorithms.

One approach to demonstrate the properties of such algorithms is the following: hand design some canonical problems in low dimensions. If an algorithm is not able to solve such obvious problems, the results for high-dimensional spaces may not be trustworthy. To facilitate this approach, the present report describes a collection of datasets that are grouped together in the Fundamental Clustering and Projection Suite (FCPS), which from the outset is focused on specific problems of structure-finding in high-dimensional spaces. The FCPS has already been successfully for comparisons of the performance of clustering or data projection algorithms, as reported previously [3–5].

2. Data Description

2.1. General Properties of the FCPS Datasets

The 10 datasets in this suite (Figure 1) pose different challenges for each clustering and/or projection algorithm. FCPS is called fundamental because any suitable clustering and/or projection algorithm should satisfactorily address these problems. The different subsets have been created to address specific challenges to the clustering algorithms, such as lack of linear separability, different or small inner class spacing, classes defined by data density rather than data spacing, no cluster structure at all, outliers, or classes in contact. A summary of the basic characteristics is given in Table 1, specific numerical properties are listed in Table 2, and the particular benchmarking problems associated with each dataset are listed in Table 3.

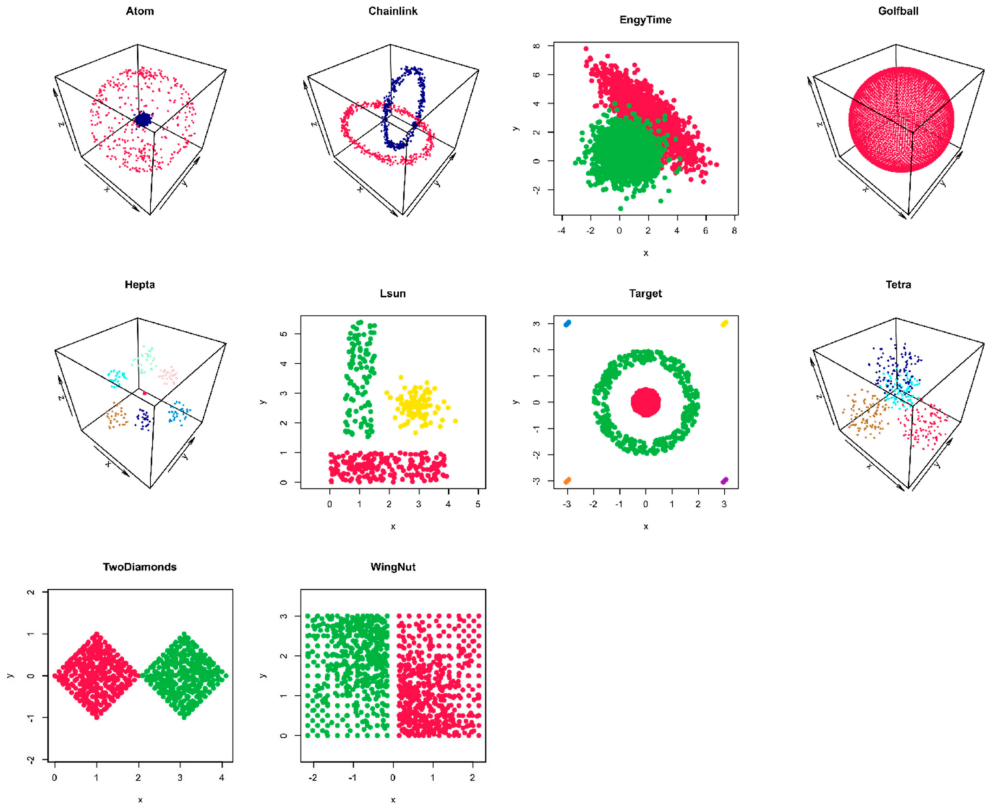

Figure 1. Visualization of the datasets of the Fundamental Clustering and Projection Suite (FCPS) suite.

Table 1. Basic properties of the components of the FCPS dataset.

Name	Cases	Dimensions	Classes
Atom	800	3	2
Chainlink	1000	3	2
EngyTime	4096	2	2
GolfBall	4002	3	1
Hepta	212	3	7
Lsun	400	2	3
Target	770	2	6
Tetra	400	3	4
TwoDiamonds	800	2	2
WingNut	1016	2	2

Table 2. Properties of the cluster components of the FCPS dataset.

Name	Hyperplane-Separable	Changing Variances	Typical Inner Class Distance[1]	Typical Interclass Distance[2]	Inter/Inner Ratio
Atom		1	2.82	41.49	14.71
Chainlink			0.08	0.98	11.59
EngyTime		1	2.70	2.70	1.00
GolfBall			1.41	1.41	1.00
Hepta	1	1	0.58	3.07	5.32
Lsun	1	1	0.17	0.87	5.21
Target		1	0.07	2.19	30.23
Tetra	1		0.45	1.01	2.26
TwoDiamonds	1		0.08	1.09	13.34
WingNut	1		0.11	0.34	3.06

[1.] Typical inner class distance: median distance in inner classes Delaunay graph; [2] typical interclass distance: median distance in inter classes Delaunay graph.

Table 3. Problems that the components of the FCPS suite of datasets pose on clustering or data projection algorithms.

Name	Key Problems
Atom	Linearly not separable, different inner class distances
Chainlink	Linear not separable
EngyTime	Density defined classes
GolfBall	No cluster structure
Hepta	Different inner class variances
Lsun	None
Target	Outlier
Tetra	Small inter class distances
TwoDiamonds	Touching classes
WingNut	Density variation within class

For the benchmarking of projection and/or clustering algorithms, a number of challenges have to be considered. The first is whether clearly defined structures are maintained in principle. The simplest structural feature is homogeneous data, that is, data that do not have a particular structure. The GolfBall data set is the canonical example (Figure 1). All points have exactly the same distances to all other points.

A second basic property is that clearly defined, well-separated clusters, each enclosed in spheres, should be visualized and clustered in exactly this way. The Hepta dataset works like this "does the algorithm work at all" test. The Tetra dataset adds a small complication to this—the distances between the different clusters are relatively small compared to the inner cluster distances. This can be seen in the last column of Table 2, where the ratio of the inner and inter cluster distances falls from more than 30 for Hepta to less than 3 for Tetra.

The Lsun dataset consists of three well-separated data classes, but with different convex hulls: a sphere and two "bricks" of different size. This structural property raises the problem of different variances or densities in the cluster.

With TwoDiamonds and WingNut, the problem of defining the boundary between two neighboring clusters arises. In TwoDiamonds, two sets almost touch each other. In WingNut, there is a clear gap between the two clusters, but the density at the gap varies.

The separability of hyperplanes, that is, that two n-dimensional datasets can be completely separated by an n-1-dimensional hyperplane, is a property that is not always given for data, especially in machine learning. Atom and chain link are canonical examples of datasets that are firstly clearly divisible into two well-separated clusters and secondly cannot be separated by hyperplanes. These datasets raise the problem that the cluster has to be "unbundled".

Outliers are a problem with all real word data. Target provides a simple canonical problem with four groups of outliers.

In some datasets, the boundaries between clusters may not be defined as "empty space" but rather consist of local minima of data density. The EngyTime dataset is of this type.

2.2. Description of the Single Datasets

2.2.1. Chainlink

Chainlink is the canonical dataset for not hyperplane separable. That is, two clusters, these being subsets of data A and B, are hyperplane (i.e., linear) separable in the $\mathbb{R}^n$ if a hyperplane of dimensionality n-1 exists, such that A and B are on different sides of the hyperplane [6].

The Chainlink dataset consists of two classes. Each class is sampled uniformly from within a torus with minor radius r = 0.1, and major radius R = 1. The two tori are orthogonally intertwined with maximal distances in between. The main challenge is here the two quite distinct classes, which are, however, not separable for any type of plane.

2.2.2. GolfBall

The GolfBall dataset has no class structure at all. It consists of 4002 points located on the surface of a sphere, such that the distances from each point to its immediate neighbors (Delaunay distances) are the same.

2.2.3. Lsun

Lsun contains 2D data with three distinct and linear separable classes of different shape. The Lsun dataset consists of $n = 400$ points in three distinct groups on a plane. Two classes are drawn uniformly distributed from within a 1×4 rectangle. These classes are arranged in the form of an "L" with a gap in between. The third group is drawn from a two-dimensional independent and identically distributed standard Gaussian centered at (3,2.5). Classes are well separated by a minimum distance of 0.6, and the classes are linearly separable.

2.2.4. Atom

The Atom dataset contains 3D data similar to an atom kernel and hull. It consists of $n = 400$ points in two distinct groups, "kernel" and "hull". Both classes consist of $n = 200$ points. The "kernel" class is drawn uniform from within a sphere placed at the origin with a radius of r = 10, whereas the "hull" class data is drawn uniform within the surface of a sphere placed at the origin with a radius of r = 50. The minimum inter class distance is 38. Classes are distinct, but are linearly not separable.

2.2.5. EngyTime

The EngyTime dataset is a density-defined 2D dataset obtained from a Gaussian mixture model (GMM). It has been originally proposed by Baggenstoss [7]. It consists of two sets of points on a plane

generated by a GMM. The Gaussian of the first class of 2048 points is centered at (0.5, 0.5), independent and identically distributed with unit variance. The Gaussian of the second class of 2048 points is centered at (2, 3), and with a variance of 2 and 1.6 oriented at the (x,$-y$) diagonal. Class membership is determined by a Bayesian decision using the GMM. The classes are defined by the probability densities. There is no distance gap between the two classes. The class separation surface is a parabola.

2.2.6. Hepta

The Hepta dataset represents seven very separate 3D clouds, one with a considerable larger density than the others. The Hepta dataset consists of six classes with $n = 30$ points each, drawn uniform from the unit cube centered at the ±3 points on the axes. The seventh class consists of $n = 32$ points drawn uniform-independent and identically distributed from a sphere of radius 0.15 centered at the origin.

2.2.7. Target

The Target dataset poses the problem of outliers. It consists of two large classes with $n = 395$ and $n = 363$ points on a plane located in a circle within a ring (linearly non separable). Furthermore, there are four sets of only three points (outliers) located away from the ring.

2.2.8. Tetra

In the Tetra dataset, clusters are located close together. The Tetra dataset consists of four classes with $n = 100$ three dimensional points in each. The points are drawn independent and identically distributed from within a unit sphere. The centers of the spheres are located at a tetrahedron with an edge length of 2.2.

2.2.9. TwoDiamonds

The TwoDiamonds dataset consists of two classes with $n = 400$ points in each. The points are drawn independent and identically distributed from within a 45° rotated square of edge length $\sqrt{2} = 1.41$, that is, one "diamond". The center of the second diamond is moved along the x-axis, such that the minimal distance between the closest points in the two classes is 0.09.

2.2.10. WingNut

The WingNut dataset has two classes and a density gradient within each class. It consists of two equal-sized datasets with $n = 508$ points on a plane. Each set is located within a rectangle of size 3×2, such that along one of the diagonals of the rectangle the density of the points is linearly increasing. The densest edges of the rectangle are placed at the opposite ends of a gap with a minimum width of 0.3 between the rectangles.

3. Clustering and Data Projection

3.1. Performance of Different Clusteriung Algorithms

Traditional clustering algorithms tend to impose a structure on the data rather than identifying the true structure in the data. As discussed previously [3], this is because the majority of clustering algorithms use an implicit or explicit shape model for the structure of a cluster, such as a sphere in k-means or a hyperellipsoid in Ward clustering. This means that for a given number of clusters k, a clustering algorithm calculates the coverage of the data with k of these geometric shapes, whether or not it matches the structure of the data. This can lead to incorrect cluster associations of samples or to the imposition of cluster structures that are not present in the data. The effects are evident in the Ward clustering results of for the five sample datasets selected (Figure 2). Only the TwoDiamonds dataset was clustered correctly, whereas the clustering of the Chainlink, LsunWingNut, and Target datasets was completely wrong. Similar problems arose for the k-means clustering algorithm, whereas the other

example methods with single and complete linkage and the median methods were also performed heterogeneously on different datasets. This shows that the use of these algorithms can lead to a wrong cluster identification and thus can lead to distortions in research.

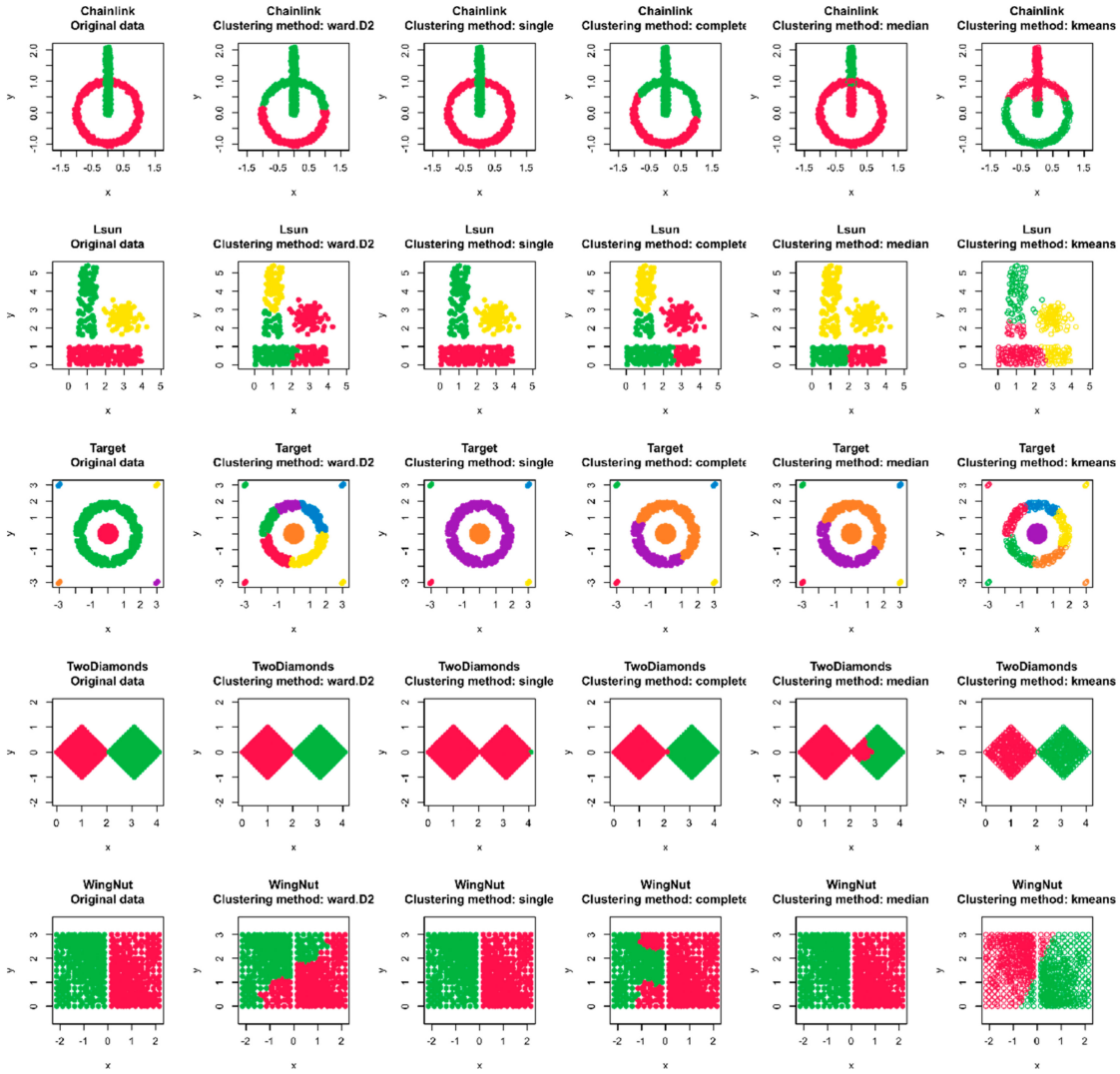

Figure 2. Examples of the resulting clusterings when applying different clustering methods to the FCPS datasets. The figure has been created on the basis of results obtained with the R library "cluster" [8] and with the "kmeans" command implemented in the base "stats" module of the R software package [9].

3.2. *Performance of Different Data Projection Methods*

The results for the various sample datasets, which were obtained using different methods of data projection, were again heterogeneous (Figure 3). Although principal component analysis (PCA) seemed to project the data correctly, with the exception of the three-dimensional tetra dataset, where the groups partially overlapped (compare the 3D visualization in Figure 1), a subgroup or cluster structure was imposed on the structureless GolfBall dataset by t-distributed stochastic neighborhood embedding (t-SNE), which was analyzed in detail elsewhere [5]. This dataset was correctly projected as cluster-free by the autoencoding neuronal network (ANN), which, however, tended to overlay the grouped data points of the Tetra and dataset, at least with the actual parameters of the network.

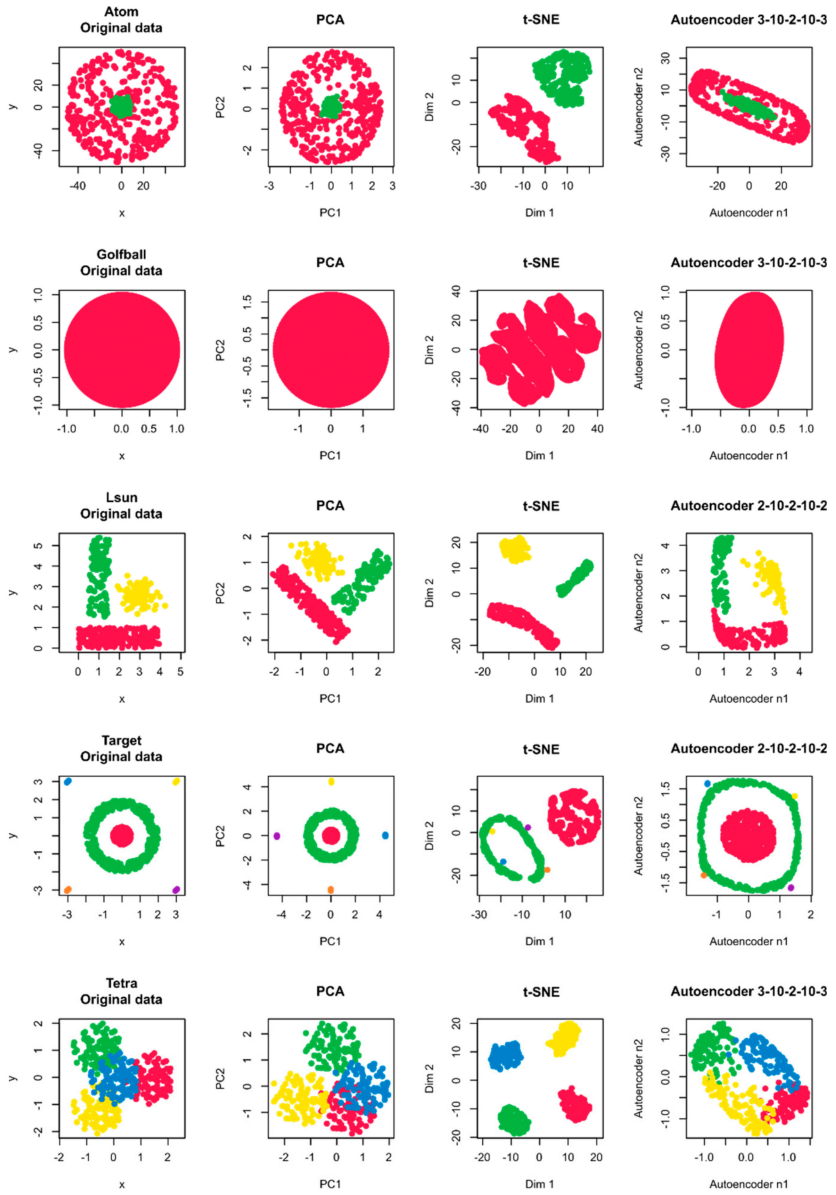

Figure 3. Examples of the resulting data projections when applying different projection methods to data from the FCPS dataset. The figure has been created based on the principal component analysis (PCA) impended in the R library "FactoMineR" [10], t-distributed stochastic neighborhood embedding (t-SNE) analysis implemented in the R library "Rtsne" [11], and the autoencoding provided by the R library "ANN2" [12].

4. Methods

Demonstrations of clusterings and data projections were created with the R software package (version 3.6.2 for Linux; http://CRAN.R-project.org/ [9]) on an Intel Core i9 computer running Ubuntu Linux 18.04.3 LTS 64-bit). A large number of frequently used methods were selected without attempting to provide a complete set of clustering or data projection methods. For the present demonstration purpose, the default hyperparameter settings implemented in the receptive R libraries were used.

Clustering problems were analyzed with the R-library "cluster" for hierarchical methods (https://cran.r-project.org/package = cluster [8]). In particular, Ward clustering [13], single and complete linkage, and the median method [14] were used. The core R implementation was used for k-means-based clustering. Clustering was limited to FCPS datasets with more than one single class, and in particular

to the datasets Chainlink, Lsun, TwoDiamonds, WingNut, and Target, which were proven to be suitable for demonstrating the challenges they pose to standard clustering algorithms.

For the data projections, the selection of FCPS datasets was different. There, datasets without any class structure were preferred, as it has been shown time and again that one of the weaknesses of some data projection methods is the dizziness of apparent cluster structures in data without such a structure [3,5]. Therefore, the GolfBall dataset was chosen. To demonstrate the power of projection algorithms on different scenarios, the datasets Atom, Lsun, Target, and Tetra were also used. As a data projection method, principal component analysis (PCA) [15] was chosen as a very commonly used method. In addition, t-distributed stochastic neighborhood embedding (t-SNE [2]) was applied as a projection method, one that is currently widely used in biomedical research [5]. Furthermore, autoencoders were implemented, which use supervised learning multilayer feedforward artificial neuronal networks (ANN) to extract the essential features of the structure of a dataset [16]. The analyses were performed using the R libraries "FactoMineR" (https://cran.r-project.org/package=FactoMineR [10]) for PCA; "Rtsne" (https://cran.r-project.org/package=Rtsne [11]) for t-SNE, which employs a Barnes–Hut implementation of the t-SNE algorithm speeding-up the computation [17]; and "ANN2" (https://cran.r-project.org/package=ANN2 [12]) for the autoencoders.

Supplementary Materials: The FCPS is available as a zip package "FCPS.zip", which contains the 10 datasets as comma separated file named as the respective dataset. The first column of each dataset contains the class membership information, the subsequent columns contain the data point coordinates. http://www.mdpi.com/2306-5729/5/1/13/s1

Author Contributions: Conceptualization, A.U. and J.L.; methodology, A.U. and J.L.; software, A.U. and J.L.; validation, A.U. and J.L.; formal analysis, J.L. and A.U.; writing—original draft preparation, J.L. and A.U.; writing—review and editing, J.L. and A.U.; visualization, J.L.; supervision, A.U.; project administration, A.U.; funding acquisition, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been funded by the Landesoffensive zur Entwicklung wissenschaftlich - ökonomischer Exzellenz (LOEWE), LOEWE-Zentrum für Translationale Medizin und Pharmakologie (JL).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Wilcox, R.H. Adaptive control processes—a guided tour, by Richard Bellman, Princeton University Press, Princeton, New Jersey, 1961, 255 pp., $6.50. *Naval Res. Logist. Q.* **1961**, *8*, 315–316. [CrossRef]
2. Peters, M.H. On the shrinking volume of the hypersphere. *College Math. J.* **2015**, *46*, 178–180. [CrossRef]
3. Ultsch, A.; Lötsch, J. Machine-learned cluster identification in high-dimensional data. *J. Biomed. Inform.* **2017**, *66*, 95–104. [CrossRef] [PubMed]
4. Ultsch, A. U*c: Self-Organized Clustering with Emergent Feature Maps. In Proceedings of the Lernen, Wissensentdeckung und Adaptivität (LWA) 2005, GI Workshops, Saarbrücken, Germany, 10–12 October 2005; pp. 240–244.
5. Lötsch, J.; Ultsch, A. Current projection methods-induced biases at subgroup detection for machine-learning based data-analysis of biomedical data. *Int. J. Mol. Sci.* **2019**, *21*, 79. [CrossRef] [PubMed]
6. Freund, Y.; Schapire, R.E. Large margin classification using the perceptron algorithm. *Machine Learn.* **1999**, *37*, 277–296. [CrossRef]
7. Baggenstoss, P.M. *Statistical Modeling Using Gaussian Mixtures and HMMs with MATLAB*; Technical Report; Naval Undersea Warfare Center: Newport, RI, USA, 2002.
8. Maechler, M.; Rousseeuw, P.; Struyf, A.; Hubert, M.; Hornik, K. Cluster: Cluster analysis basics and extensions R package version 2.0. 1. 2015. 2017.
9. R Development Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2018; Available online: https://www.R-project.org/ (accessed on 26 December 2019).
10. Le, S.; Josse, J.; Husson, F.c. Factominer: A package for multivariate analysis. *J. Stat. Softw.* **2008**, *25*, 1–18. [CrossRef]

11. Krijthe, J.H. Rtsne: T-distributed stochastic neighbor embedding using barnes-hut implementation. 2015. Available online: https://github.com/jkrijthe/Rtsne (accessed on 26 December 2019).
12. Lammers, B. Ann2: Artificial neural networks for anomaly detection. 2019. Available online: https://rdrr.io/cran/ANN2/ (accessed on 26 December 2019).
13. Ward Jr, J.H. Hierarchical grouping to optimize an objective function. *J. Am. Stat. Assoc.* **1963**, *58*, 236–244. [CrossRef]
14. Gower, J.C. A comparison of some methods of cluster analysis. *Biometrics* **1967**, *23*, 623–637. [CrossRef] [PubMed]
15. Pearson, K. LIII. On lines and planes of closest fit to systems of points in space. *London, Edinburgh&Dublin Philosoph. Mag. J. Sci.* **1901**, *2*, 559–572.
16. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
17. Van der Maaten, L. Accelerating t-SNE using tree-based algorithms. *J. Machine Learn. Res.* **2014**, *15*, 3221–3245.