A Galaxy of Correlations

Detecting Linear Correlated Clusters through k-Tuples Sampling using Parameter Space Transform

Daniyal Kazempour
Ludwig-Maximilians-University Munich
kazempour@dbs.ifi.lmu.de

Lisa Krombholz
Ludwig-Maximilians-University Munich
Lisa.Krombholz@campus.lmu.de

Peer Kröger
Ludwig-Maximilians-University Munich
kroeger@dbs.ifi.lmu.de

Thomas Seidl
Ludwig-Maximilians-University Munich
seidl@dbs.ifi.lmu.de

ABSTRACT

In different research domains conducted experiments aim for the detection of (hyper)linear correlations among multiple features within a given data set. For this purpose methods exist where one among them is highly robust against noise and detects linear correlated clusters regardless of any locality assumption. This method is based on parameter space transformation. The currently available parameter transform based algorithms detect the clusters scanning explicitly for intersections of functions in parameter space. This approach comes with drawbacks. It is difficult to analyze aspects going beyond the sole intersection of functions, such as e.g. the area around the intersections and further it is computationally expensive. The work in progress method we provide here overcomes the mentioned drawbacks by sampling d-dimensional tuples in data space, generating a (hyper)plane and representing this plane as a single point in parameter space. By this approach we no longer scan for intersection points of functions in parameter space but for dense regions of such parameter vectors. By this approach in future work well established clustering algorithms can be applied in parameter space to detect e.g. dense regions, modes or hierarchies of linear correlations in parameter space.

1 INTRODUCTION

Typing into Google Scholar [1] the query **"linear correlation between"** yields around 343.000 scientific works from various domains such as medical science, chemistry, biology, pharmacology, electric engineering, economics, physics. Further limiting the search by adding "multivariate" to the previous query reduces the results down to around 20 scientific works. The insights are here twofold: first, there is a demand for detecting linear correlations in various domains and second, as for now only few scientific works have investigated linear correlations between multiple variables. One real-world example for linear correlations among multiple features is in the wages data set[2]. It contains the statistics of potentially influencing factors of wages from 1985 Current Population Survey. Visualizing the data reveals that there are linearly correlated clusters among the features "years of education", "years of work" and age. As a second example in the scientific domain of water research in a work by [4] the authors revealed a linear correlation between the hydroxyl-radical concentration and the inactivation time of E.coli in a photocatalytic disinfection substance.

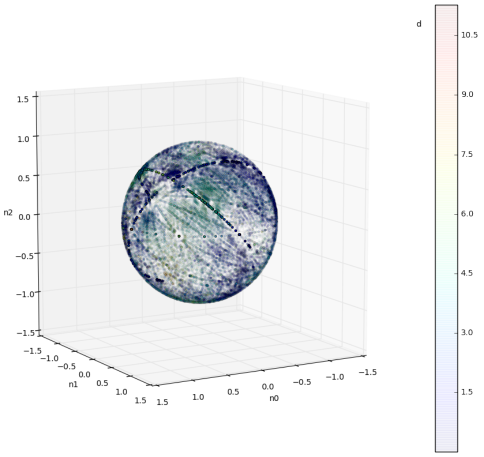

Figure 1: Triple of 3D data points sampled in data space yield this spherical galaxy of correlations in a Hessian Normal Form parameter space. Highly dense regions represent areas with high correlations between data points.

Among the various methods for detecting linear correlated clusters in high dimensional data there is one particular method which relies on parameter space transform: CASH [1]. This method comes, compared to its competitors, with advantages that it is highly robust against noise and detects global linear correlated clusters, being independent of any kind of locality assumption. As we shall elaborate more in detail in section 3 this method works by projecting data points from data space to parameter space becoming a data point functions. The parameter space is then scanned for intersections of such data point functions which represent data points being linearly correlated. However the scan for intersections in parameter space is computationally expensive and the capabilities to further analyze the area around the intersection areas are not given by this approach. We provide the following two major contributions in this work:

- Providing a novel approach for detecting regions of intersection by generating d-dimensional samples from which a linear function is derived. This linear function is then represented in parameter space as a vector eliminating the need to scan for intersections resulting in galaxy-like shapes as seen in Figure 1 and
- An opportunity to analyze further aspects on the detected clusters such as e.g. different densities and thus variances

[1] scholar.google.com
[2] http://lib.stat.cmu.edu/datasets/CPS_85_Wages

10.5441/002/edbt.2019.91

Table 1: Selected Parameter Transform Methods

Method	Strategy	Strengths	Weaknesses
Hough Transform	grid-based (accumulators)	Simple strategy	No pruning
CASH	iterative parameter axis splitting	efficiency from DFS	Slow on high-noise data
D-MASC	De-noising with Mean-Shift, Rasterization of functions	Effective against high levels of noise	slow on low-jitter and low-noise data

of linear correlated clusters as well as density connected clusters and their semantics.

2 RELATED WORK

The parameter transform, which is also known as Hough transform, has been first introduced in a patent by Paul V.C. Hough in [6] in context of edge detection on images. The first application of parameter transform in context of detecting (hyper)linear correlated clusters was in the work by [1]. This approach despite its high level of sophistication suffered regarding its runtime if the data has a high amount of noise and jitter. To approach this issue in a recent work [7] the authors provide a method in which the data is pre-aggregated using Mean Shift [3] in data space which yields modes. These modes are transformed into mode-functions in parameter space, which then are rasterized into cells. Those cells from mode functions which are overlapping most with other mode-function cells are considered as candidates for linear correlated clusters. All the mentioned related works aim primarily at finding intersections in parameter space at a specific resolution. The intuition behind this resolution is that the smaller the detected cells in parameter space, the more are the points located on a specific line, plane or hyperplane. An overview on the mentioned linear parameter transform methods are provided in Table 1. It would be of interest to detect e.g. chains of dense regions by applying DBSCAN [5], or centroids by applying centroid or mode based methods such as e.g. MeanShift, or determining hierarchies of linear correlated clusters by applying e.g. single-link to the parameter space. Since we are dealing with functions and not with points in parameter space, we can not apply the mentioned methods.

3 PROJECTING SAMPLED K-TUPLES TO PARAMETER SPACE

Having given an overview on the related work, we elaborate in this section on our work in progress in more detail and compare it to the currently used approach. In the methods described in the related work section, each data point p_i in data space $\mathbb{D}$ is projected to parameter space $\mathcal{P}$ as a data point function $p_i \mapsto f_{p_i}$ where $f_{p_i} := b = y - m \cdot x$, where m represents the slope of a line and b the intercept. An intersection of several of such data point functions at a specific point (m_s, b_s) means that their corresponding data points are located on a line with a common slope and common intercept as it can be seen in Figure 2. Since data points are rarely located perfectly on a line, in the related work it is looked for at least *minpoint* data point functions intersecting within a maximum (m, b)-range. Here *minpoint* is a hyperparameter set by the domain experts. Further regarding the range, the intuition is the following: the smaller the range for the slope and intercept, the more precise are the data points located on an explicit line, and the higher the linear correlation.

In contrast to the related work, in our method all k-Tuples (here all pairs) of data points are taken. From these tuples (p_i, p_j), for each of them a line with a specific slope and intercept is calculated. From the lines we obtain the slope and intercept for each tuple which is a point in parameter space. A point or region in parameter space where these slope-intercept coordinates are densely located represent a correlation in data space as it can be seen in Figure 2. In a formalized manner:

$$\forall (p_i, p_j), where p \in \mathcal{D} : (p_i, p_j) \mapsto f_{p_i, p_j} = (m_{p_i, p_j} . b_{p_i, p_j}) \in \mathcal{P} \tag{1}$$

,

For detecting the dense regions in parameter space we apply density based clustering algorithms such as DBSCAN [5] and OPTICS [2]. In DBSCAN we have two hyperparamters, namely *minpoints* which defines the minimum number of data points which are expected to be located within an ϵ*-neighborhood*. In context of the parameter space, the effects of controlling minpoints and ϵ are the following:

$$\begin{aligned} minpoints &\mapsto |S| \in Corr_{(m,b)}, \\ where \quad S &:= \{(p_0, p_1), \ldots, (p_i, p_j)\} \subseteq DB \\ and \quad \epsilon &\mapsto \sigma(Corr_{(m,b)}) \end{aligned} \tag{2}$$

Here the intuition is as follows: the minpoints in DBSCAN represent the minimum number of data point tuples which are expected to have the same parameter values (or value ranges) and thus belonging to the same linear correlation. The ϵ hyperparameter represents the variance σ or resolution we allow for the data points around a linear correlation.

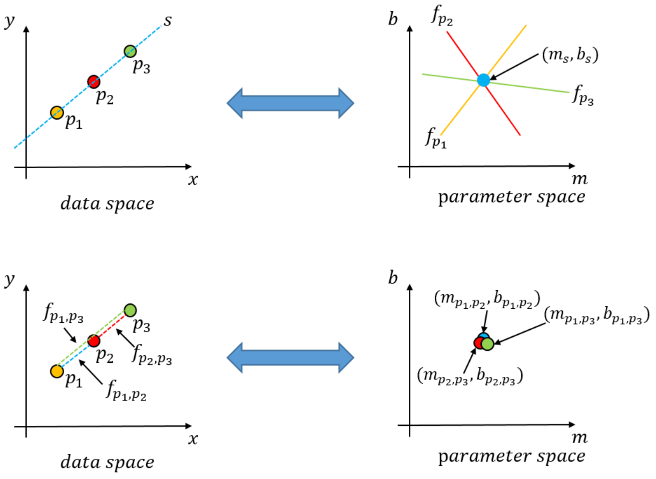

Figure 2: Comparison of current parameter space transform approaches (top) to our method (bottom).

At this point we have to highlight that a question which may rise immediately is: how do we choose the two parameters? This can be partially addressed by using OPTICS which eliminates mostly the issue of determining a proper ϵ value and enables the detection of density hierarchies. Having determined the dense regions or sampled linear correlations, our method computes the median of such regions. The median comes with a pleasant effect that it weeds out the influence of outlier k-tuple correlations in parameter space. As for now our method can be summarized into the following steps as it can be seen in Figure 3

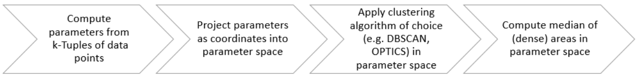
Compute parameters from k-Tuples of data points
Project parameters as coordinates into parameter space
Apply clustering algorithm of choice (e.g. DBSCAN, OPTICS) in parameter space
Compute median of (dense) areas in parameter space

Figure 3: Pipeline with its single stages of our method.

The parameter space provided in our example is the slope-intercept form. There is a variety of linear parameter space representations. Since the slope-intercept form comes with various drawbacks (unbounded dimensions, unable to project y-axis parallel linear correlations, non-ambiguous representations in higher dimensions etc.) we use in this work the Hessian Normal Form (HNF) since it comes with the advantages of the slope-intercept and other representations but with none of their disadvantages as stated in [8]. In the HNF representation a line or (hyper)plane is represented through:

$$\begin{aligned} \delta =< p, n >, where p := (p_0, p_1, ..., p_d) \in \mathcal{D}^d, \\ and \quad n := (n_0, n_1, ..., n_d) \in \mathcal{P}^d \end{aligned} \tag{3}$$

,

Here n represents the normal vector and δ represents the distance from origin orthogonal to the hyperplane.

Further we shall see in the complexity section why a full k-Tuple construction is computationally infeasable, especially in higher-dimensional settings.

4 COMPLEXITY

The runtime complexity of the related work CASH is in worst case $O(s \cdot c \cdot n \cdot 2^d)$ where s reflects the number of intersections and thus the resolution of the grid, c denotes the number of found clusters, n represents the number of data points in the data set and d stands for the dimensionality of the data space. In comparison D-MASC has a runtime complexity of $O(Tn\log(n)+(\frac{len(bounds_d)}{w})^d m^2)$ with T denoting the number of iterations of the MeanShift algorithm for initially reducing noise and jitter in data space, $len(bounds_d)$ representing the range of the parameter space range in which we are looking for intersections, w standing for the width of the cells being generated in a rasterization process and m for the number of resulting modes after applying MeanShift. Our method requires for computing the parameter coordinates for all data point all k-tuples, where k corresponds to the dimensionality d of the original data set. With regards to the dimensionality, we require in a 2D data set all two-tuples, in a 3D data set all three-tuples etc. This yields a runtime of $O(\binom{n}{d})$. DBSCAN requires with an indexing structure that executes the neighborhood query an overall runtime of $O(n\log(n))$. Thus we get for our method in total a runtime of $O(\binom{n}{d} + n\log(n))$. From a runtime point of view, our method has an exponential runtime with regards to the dimensionality like CASH and D-MASC. However instead of generating all $\binom{n}{d}$ tuples, one strategy is to sample over the data points. In some preliminary experiments we could observe that sampling yielded in most cases as accurate results as performing a full enumeration of all k-tuples. As for now, our assumption is, that if data points are correlated in data space, so do their samples reflect the correlation up to a certain extent. However, since this is a work in progress, an exhaustive analysis on theoretical as well as on experimental level is required to prove the assumptions.

5 EXPERIMENTS AND DISCUSSION

Now that we have elaborated on our method and its runtime complexity we provide here experiments which focus on the quality of the detected clustering results. As a first experiment we take one of the data sets which is used in D-MASC. The two-dimensional data set consists of 100 data points contributing to three linear correlated clusters with irregular densities. To these 100 data points 90% noise is contributed. According to the pipeline mentioned in section 3, first all 2-tuples of data points are created, then projected into the parameter space. In the parameter space we can already observe dense regions. Applying OPTICS we get the following plot as seen in Figure 4.

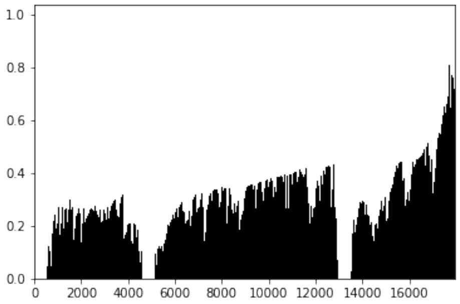

Figure 4: OPTICS plot of the parameter vectors in parameter space.

The three valleys indicate three almost equally high-dense regions in parameter space. As a result of DBSCAN with $minpoints = 33$ and $\epsilon = 0.015$ we obtain the following regions in parameter space which are marked with an 'x' in Figure 5

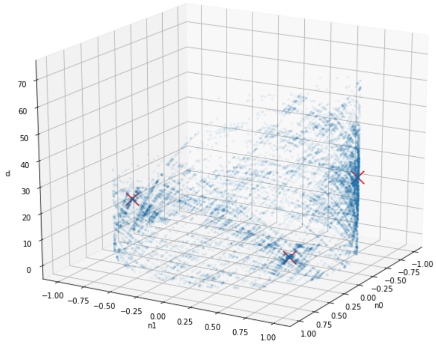

Figure 5: Detected three high-density regions in parameter space.

After having computed the median from each of the dense regions, our method was capable of detecting all three linear correlated clusters with all the data points being assigned to their respective cluster as it can be seen in Figure 6.

As a teaser for its performance on data sets with a dimensionality higher than two, we have a three dimensional data set

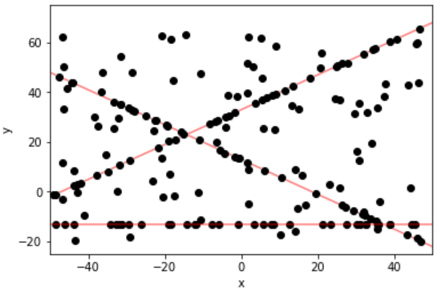

Figure 6: Detected three linear correlated clusters in data space.

consisting of 66 data points. From which 36 data points belong to planar correlations and 30 data points are randomly generated noise. The 36 data points belong to two planar correlated clusters, with 18 data points per correlation. In parameter space we generate thus $\binom{66}{3} = 45760$ parameter vectors in parameter space. Our method detects in parameter space two clusters as it can be seen in Figure 7 where two very deep valleys can be seen. The first figure in the introduction of the paper is the actual parameter space of this data set. The three axes represent each a dimensional of the normal vector. The color (black) represents the distance δ. The parameters for the density based clustering were $minpoints = 250$ and $\epsilon = 0.01$.

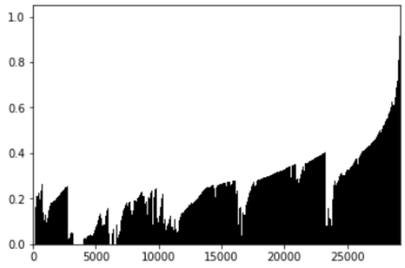

Figure 7: OPTICS plot of the 3D data set with two highly-dense clusters.

As an result we get both planes detected correctly and all points assigned to their corresponding planar correlated clusters as seen in Figure 8.

6 CONCLUDING REMARKS AND FUTURE PROSPECTS

In this work in progress, we have provided a first concept for a different approach in detecting (hyper)linear correlated clusters in parameter space by sampling k-Tuples in data space, generating k-dimensional (hyper)planes. The parameters of these (hyper)planes are projected to parameter space. In the parameter space we used as an example density based methods for detecting highly dense regions. This approach of dealing with points in parameter space instead with (hyper)linear functions opens new possibilities of analysis. Primary targets for future work are evaluating sampling strategies, applying the experiments to high-dimensional and also real world data and evaluating different clustering models in parameter space (hierarchical, centroid-based, subspace etc.). We hope to encourage with this paper to not only develop different approaches for clustering using parameter transform itself, but also fostering the research of parameter space transformation based methods making discoveries in the galaxies of correlations.

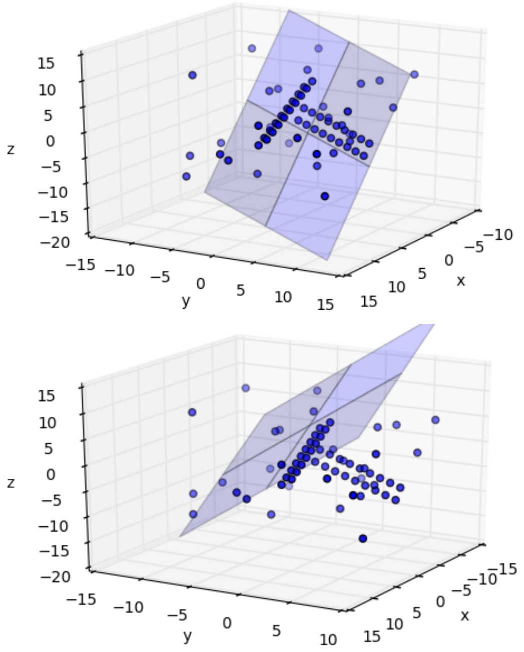

Figure 8: Two planar correlated clusters

ACKNOWLEDGMENTS

This work has been funded by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The authors of this work take full responsibilities for its content.

REFERENCES

[1] Elke Achtert, Christian Böhm, Jörn David, Peer Kröger, and Arthur Zimek. [n. d.]. Global Correlation Clustering Based on the Hough Transform. *Statistical Analysis and Data Mining: The ASA Data Science Journal* 1, 3 ([n. d.]), 111–127.

[2] Mihael Ankerst, Markus M. Breunig, Hans peter Kriegel, and Jörg Sander. 1999. OPTICS: Ordering Points To Identify the Clustering Structure. ACM Press, 49–60.

[3] Yizong Cheng. 1995. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17, 8 (1995), 790–799.

[4] Min Cho, Hyenmi Chung, Wonyong Choi, and Jeyong Yoon. 2004. Linear correlation between inactivation of E. coli and OH radical concentration in TiO2 photocatalytic disinfection. *Water Research* 38, 4 (2004), 1069 – 1077.

[5] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *KDD*.

[6] Paul VC Hough. 1962. Method and means for recognizing complex patterns. (Dec. 18 1962). US Patent 3,069,654.

[7] Daniyal Kazempour, Kevin Bein, Peer Kröger, and Thomas Seidl. 2018. D-MASC: A Novel Search Strategy for Detecting Regions of Interest in Linear Parameter Space. In *Similarity Search and Applications - 11th International Conference, SISAP 2018, Lima, Peru.* 163–176.

[8] Daniyal Kazempour, Andrian Mörtlbauer, Peer Kröger, and Thomas Seidl. 2018. Mirror Mirror on the Wall, What is the Fairest Linear Parameter Space Representation of All? On Representations of Linear Parameter Space in Context of Clustering. In *Proceedings of the Conference "Lernen, Wissen, Daten, Analysen", LWDA 2018, Mannheim, Germany.* 169–173.