A Survey Of
Radial Projection Techniques

Group 4

Lukas Neuhold, Ridvan Aydin, Georg Regitnig

706.057 Information Visualisation SS 2020
Graz University of Technology

18 May 2020

Abstract

Visualizing multivariate data is an important task in our modern world. These data sets have many dimensions, and a very basic approach of looking at the data in a spreadsheet will hardly reveal any interesting aspects of the data. There exist many different methods to visualize multivariate data. One way to achieve this visualization is with radial projection techniques. In this survey we will explain what those are and present common approaches and how they differ. We will further conduct a case study with commonly available tools that implement these techniques. The case study was conducted with the classic, publicly available, cereals dataset, as well as a publicly available dataset from the government of Styria.

Contents

List of Figures

Chapter 1

Introduction

Our modern world relies a lot on data, be it business or science, or government. A lot of this data is multivariate, meaning it has a high number of dimensions. Analyzing this data cannot simply be done by looking at a spreadsheet. To help find correlation, clusters, and other properties of the data powerful visualization tools are needed.

1.1 Coarse vs Exact

To visualize this data, different approaches can be taken. To distinguish these approaches, the authors of this survey thought of them as coarse or exact representations. This definition is expanded upon from Kandogan [2001], when talking about Permutation Matrices.

1.1.1 Exact

Exact visualization distinguishes itself by the fact that every data record is represented by one symbol per dimension of the data. From a visualization like this it is easier to recover the original data value. However due to the fact that every dimension is mapped by an individual symbol the view can get cluttered. Whilst exact visualizations offer a detailed few of the data, it might be difficult to find clusters or correlation in the data. A task like this might be easier with a simplified view provided by coarse visualizations.

One such exact approach is parallel coordinates. With this visualization one data record is represented as multiple line intersections. Data features are lines normal to an axis, and data records are lines along this axis. The crossing of this dimension axis and the data records line indicate the value of the record in this dimension. Another one would be star plots, also known as radar plots. Similar to parallel coordinates the data record is represented by multiple line intersections. The difference comes from the fact that whereas parallel coordinates are laid out along an axis, star plots are laid out radially around a center point.

1.1.2 Coarse

Coarse representation is defined by a data record only being represented by a single symbol. Most commonly this symbol is a point in a two or three dimensional space. It is not trivial to recover the original data values from this single projected point. Coarse mappings provide a simplified view on the data. However they introduce ambiguity by the fact that the dimensions of the data are reduced. They can offer a valuable view on the data especially early on in analysis. The fact that dimensions are reduced, will however always lead to ambiguity being introduced. This includes problems like cluster shapes being distorted, or different data points being mapped to the sampe position in the dimension reduced space. The radial projection techniques covered in this survey were categorized to be coarse visualizations.

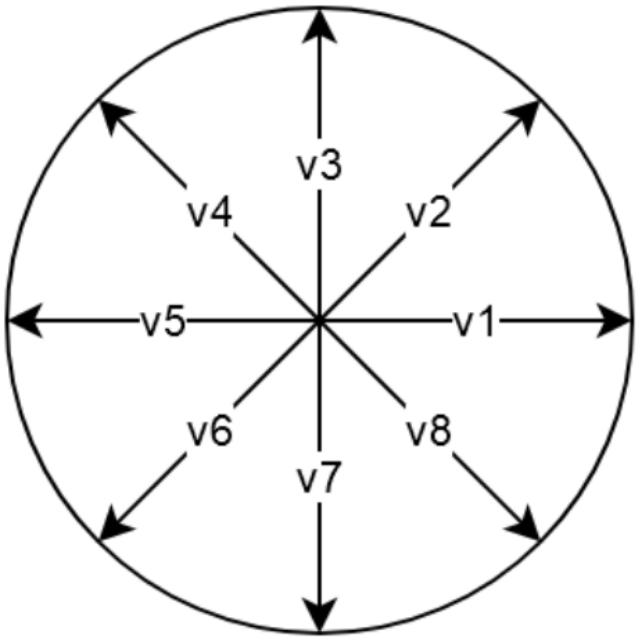


Figure 1.1: A basic radial layout of dimension base vectors. [Figure created by the authors of this paper.]

1.2 Radial Projection Techniques

Radial projection techniques are a coarse multidimensional data visualizations, where data records are mapped onto a 2D plane. These data records are represented by a single symbol in this 2D plane. The dimensions of the data are represented as two dimensional vectors laid out in some radial fashion, as seen in figure 1.1. These vectors build the base vectors with which all data records are mapped onto the 2D plane.

Different methods differ in how they approach this projection. Some techniques have a normalization step for the mapping, whereas others do not normalize the 2D coordinates of the data record. Other techniques improve upon the initial mapping with different optimization techniques. All these differences try to improve upon different goals, for example the clustering of data.

1.3 Non-Radial Projection Techniques

To further illustrate the difference between radial projection techniques, which wil be discussed in greated detail the following chapter, and non-radial projection techniques, some well known non-radial projection techniques are discussed briefly.

1.3.1 Star Plots

Star plots are called by many different names. They are also known as radar plots, kiviat plots, glyph plots and many other names. It is a visualization technique where each data record is represented as a star shaped icon. The shape of a star emerges because the dimensions are laid out radially and uniformly spaced originated from a single point in the center and a polyline crosses through all of the axes at a position proportional to its value for that dimension. The visual image (shape of the polygons) can vary for the same dataset when the dimensions are ordered differently. The number of polygons is equal to the number of data records in the dataset. Usually the plot is generated as a multi plot which consists of many star plots. Each star plot represents an observation of a different data record. With a set of star

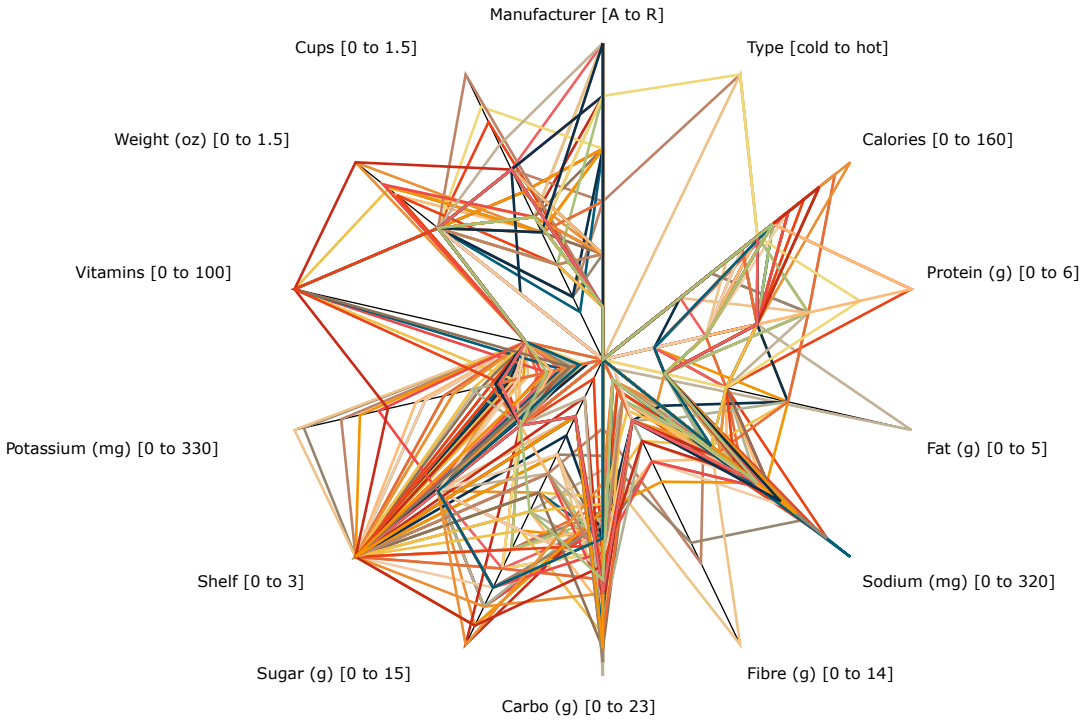


Figure 1.2: An overlapping star plot of the classic cereals dataset. [Image created by Keith Andrews and used with kind permission.]

plots for a given amount of observations, it is possible to identify similarities in the observations and also perform clustering operations. The star plots can also provide information about the relative dominance of a variable in comparison to other variables in the observation [Sangli 2014].

Limitations

The star plot technique is very useful to display multivariate data and to get information about the relative values of the variables with other data records in the dataset. However star plots can encounter a problem of a too cluttered visualization when the dataset provides a huge amount of data records. To overcome this problem several techniques can be used [Sangli 2014]. We will names these techniques in this survey but we will not cover them in detail.

- Overlapping Star Plot
- Shifted Origin Plot
- Multilevel Star Plot

1.3.2 Parallel Coordinates

Parallel coordinates can be also categorized as axis reconfiguration techniques. It is a visualization technique to visualize multivariate datasets. It does not use radial attributes like the techniques we already talked about. With parallel coordinates, each dimension is represented as a horizontal or vertical axis. It consists of n axes for n dimensions and the axes are organized as uniformly spaced lines. A data record in this n-dimensional space is mapped to a polyline. The line crosses through all of the axes at a position proportional to its value for that dimension. By laying out the vertical axis horizontally across the screen, parallel coordinates provide a huge advantage over conventional orthogonal coordinates since the number of dimensions that can be visualized are not restricted anymore. The only parameter restricting the visualization would be the horizontal resolution of the screen. With this encoding scheme, parallel coordinates does not have an issue with dimensionality anymore. It also helps to spot correlations between variables in the dataset very easily. The problem when using parallel coordinates as a visualization for

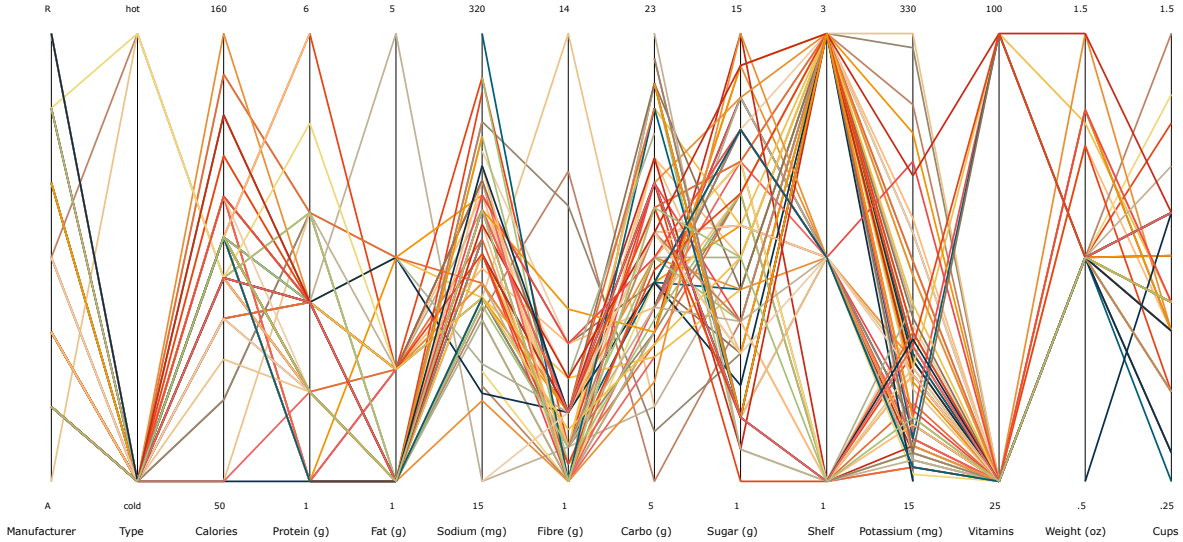


Figure 1.3: A simple parallel coordinates visualization using the cereals dataset. [Figure created and kindly provided by Keith Andrews.]

large datasets is that the amount of clutter gets higher the more the data records are. It can lead to a mass of overlapping lines which then precludes the reception of relative densities presented in the data [Ying-Huey Fua et al. 1999].

Hierarchical Parallel Coordinates

Hierarchical parallel coordinates is a variation of parallel coordinates to convey aggregation information for resulting clusters. It reduces the amount of clutter by imposing a hierarchical organization on the dataset. The aggregation of the data can then be displayed at different levels of abstraction [Ying-Huey Fua et al. 1999].

1.3.3 Biplots

Biplots are a form of multivariate data visualization based on the fact that any matrix $m \times n$ of rank 2 can be approximated by two matrices of size $m \times 2$ and $2 \times n$ [Gabriel 1971]. This can be generalized for any rank of matrix, however for visualization the rank 2 case is the most important one. As the decomposition into two 2-dimensional matrices is only possible for matrices of rank 2, higher ranking matrices need to be approximated. This can be achieved via singular value decomposition and least squares minimization [Gabriel 1971].

After having split the original matrix in such a way, one obtains two matrices that are 2 dimensional. These points can then be placed in a 2D plane. In general one set of points is drawn as vectors from the origin, called biplot vectors, and the other as points, called biplot points [Greenacre 2010]. Either one of the factorized matrices can be considered the biplot vectors whilst the other represents the biplot points.

The biplot vectors form a biplot axis where every biplot point can be projected onto via the dot product, letting one recover the original matrix entry [Greenacre 2010]. The length of the biplot vector informs one of the rate of change for the values on this axis [Greenacre 2010]. A short biplot vector means slow changing values, whereas a long biplot vector indicates fast changing values [Greenacre 2010]. The orientation of biplot vectors lets one argue about the correlation of variables. If they have the same orientation, it indicates high inter-variable correlation[Greenacre 2010].

Due to the fact that one can recover the original data value from the graph by a simple dot product, biplots would be considered an exact approach. This is further coroborated by the fact that we have two

symbols representing the data. However, it gets muddied by the fact that one might need to approximate the original data matrix to a matrix of rank 2. It is also not a radial projection technique as the dimensions of the data are not laid out radially and used to map the datapoints.

Chapter 2

Common Techniques

In this chapter a closer look is taken at the most well known radial projection techniques. The focus is on introducing these techniques, as well as some improvements that help further the usefulness of the technique in certain aspects. Towards the end of the chapter a couple of non radial projection techniques will be discussed.

2.1 Star Coordinates

Star coordinates were introduced by Kandogan [2001]. They extend the idea of two-dimensional scatter plots, where the data values of a certain point can easily be determined by checking the corresponding distances to the origin on the two coordinate axes. To allow a similar procedure using higher dimensions, the coordinate axes are not laid out orthogonal to each other, but in a circular shape. The origin of the axes defines the center of the circle. All axes are of equal length and the angles between the axes are of the same size. Before mapping data points, their dimensions have to be scaled to allow a meaningful visualization. The minimum of each dimension is mapped to the origin, the maximum to the point on the corresponding axis that has a distance equal to the circle's radius.

When mapping a data record, its 2D coordinates are calculated as the sum of all base vectors multiplied by the record's value for that dimension. An immediate consequence of this mapping is the ambiguity introduced by reducing the number of dimensions. A mapped point could belong to a possible infinite amount of data records since non-equal data records map to the same point if the sum of their dimensions multiplied with the unit vectors is the same. The Star Coordinates visualization provides several operations to reduce this ambiguity and allow better understanding of the underlying data:

Scaling The length of each axis can be modified, directly affecting its contribution during the mapping.

Rotation Individual axes can be rotated around the center, effectively changing the direction of the axis' unit vector. This allows to specify how much this axis should correlate with other axes.

Query When one data point is selected, all values of the corresponding data record are displayed.

Additionally the user can select an arbitrary amount of points which are drawn in a different color, providing helpful information about how scaling and rotating affects certain datapoints. It is also possible to define certain ranges on individual axes and mark datapoints according to their value for that coordinate.

2.1.1 Orthographic Projection

Orthographic Star Coordinates were introduced by Lehmann and Theisel [2013] and address the distortions that might occur during the projection of the datapoints. When using Star Coordinates, two points that are close to each other in the mapping might be far away from each other in the original high dimensional space and vice versa. This can be avoided by restricting the projections used for the data mapping

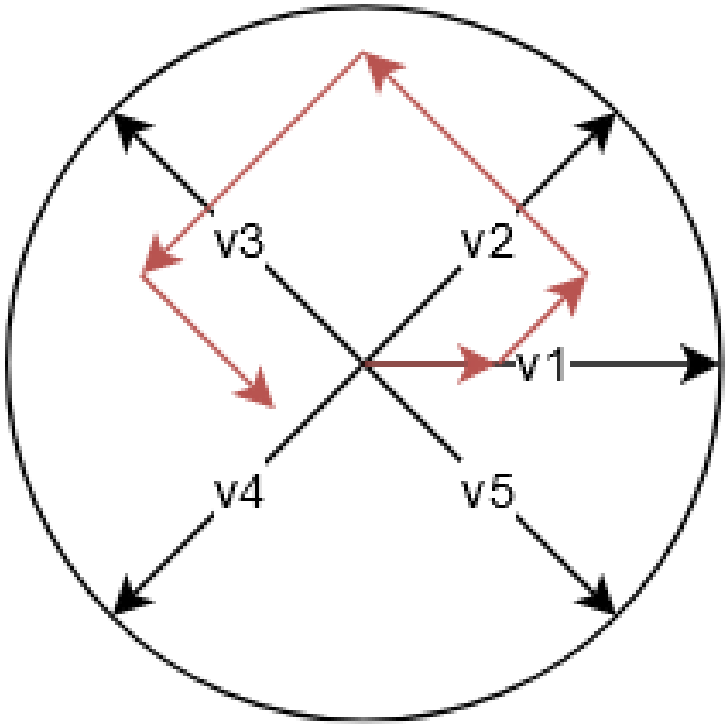


Figure 2.1: A data record is mapped by summing up its values in each dimension. [Figure created by the authors of this paper.]

to orthographic projections. They provide two important properties that improve the visualization result. Spheres with a certain radius in the high dimensional space are mapped to a 2D circle of the same radius. Additionally, the distance between two mapped points is not larger than their distance in the original space. Furthermore methods to interactively modify the axes layout while still satisfying the constraints of orthographic projections are provided.

2.1.2 Axis Calibration

One of the biggest challenges when working with Star Coordinates is estimating the original data values from the mapped datapoint. Since the mapped point is calculated using all available dimensions, changing the assumption about one dimension directly affects all other dimensions. Axis calibration, introduced by Rubio-Sánchez and Sanchez [2014], suggests two approaches to improve the data estimation. The first step moves the datapoints to reduce the expected estimation error for all points in the dataset. The second step restricts the dimension vectors to orthonormal vectors to minimize the estimated error during the projection. Combining these techniques greatly increases the accuracy of estimated data values.

2.2 RadViz

RadViz is a tool which can project a n-dimensinal dataset into a 2D space. The goal is to be able to interpret the influence of each dimension as a balance between the influence of all dimensions. This is done by a physical spring model [Hoffman et al. 1997]. RadViz uses a set of n-dimensional vectors v_i $(i = 1, ..., m)$ where the number of these vectors define the dimensional anchor points of n springs. The coordinates of a n-dimensional datapoint are represented by a point p which then is connected to these anchor points. The attributes of the datapoint determine the location of p on the graph and also its stiffness. The location of p lies at the position where the sum of the spring forces equals 0. There is also to mention that the attributes of this datapoint must be non-negative but they are usually normalized that the range of each variable is between 0 and 1 [Rubio-Sánchez et al. 2015].

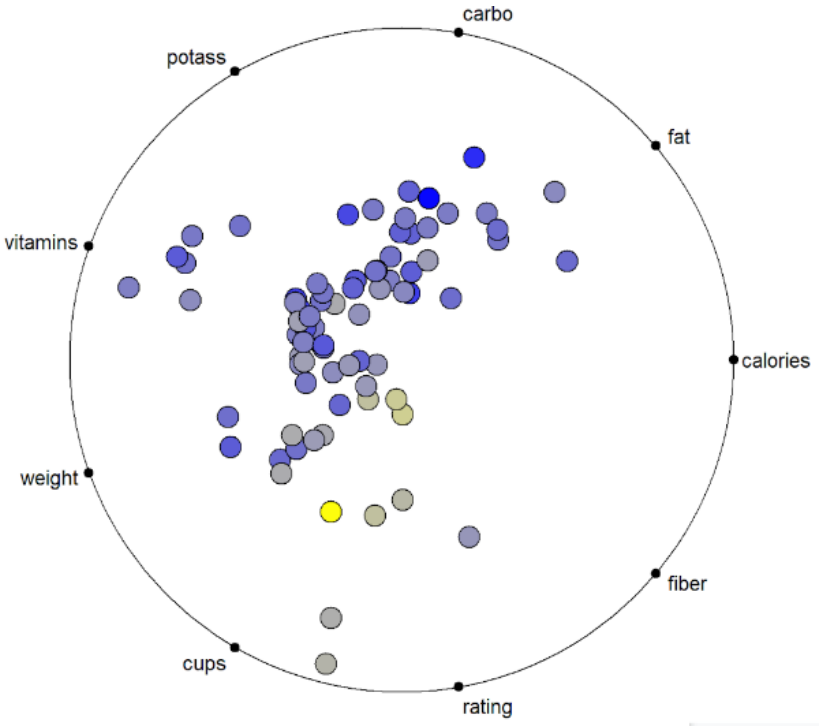


Figure 2.2: RadViz visualization of the cereals dataset. [Figure created by the authors of this paper using the tool RadViz-X.]

Mapping

The mapping of RadViz is very interesting and offers some features. If the graph has n dimensions and the values of all n coordinates have the same value, this datapoint will lie exactly at the center of the circle. When the values of the coordinates of the points are nearly equal, these points will lie close to the center of the graph. This means also that when the values of the coordinates are greater in a specific dimension, the point will be closer to this dimension. Datapoints which are unit vectors lie exactly at the position where the spring for that dimension is fixed. If there are points with similar values but dimensions which are opposite to each other on the radial circle, these points will lie near the center. Other features are that a n-dimensional line maps to a line and a n-dimensional plane maps to a bounded polygon. Also, a sphere maps to an ellipse. Overall this achieves an intuitive display which represents a nonlinear transformation of the data and also preserves certain symmetries [Hoffman et al. 1997].

2.3 FreeViz

"FreeViz optimizes a linear projection and displays the projected data in a scatterplot" [Demsar et al. 2007]. FreeViz is an extension of RadViz. To find the target projection, FreeViz uses a gradient optimization algorithm which aims to separate the distances of different classes in the class-labelled data. It helps to provide an explorative analysis to hint at classes, potential patterns and relations.

2.3.1 Methods

For the optimization, a gradient descent optimization can be used on the computed gradients. After each step, the gradients are recentered and renormalized to prevent imploding or exploding. Normalization is done so, that the sum of all base vector projections is zero and the longest base vector is of unit length. This is repeated until the potential energy is stable for the next steps. Other maybe more advanced algorithms than the gradient decent optimization can be used as well for the optimization [Demsar et al. 2007].

2.3.2 Visualization

The Visualization of FreeViz contains a lot of features and reveals a lot of information. The base vectors can be represented with lines or with a symbol at the non-origin endpoint of the vector projection. The lines represent the coordinate axes in the original space and are easier to spot and compare. Symbols instead of lines can be used for a grouping by proximity and can also be favoured when the dataset has many attributes to avoid having to many vector lines in the visualization. The points can be colored to show which class they belong. The regions in which a class lies can also be colored with the corresponding color [Demsar et al. 2007].

Datasets have often different features of different importances. Since the features are normalized, the features with a longer projection of the base vector have a larger impact on the placement of instances and thus are more important. FreeViz optimizes a projection based on classification so the features which are more important for the classification will have longer projections. If the dataset contains a lot of features, the most important eatures can be exposed by only visualizing the base vectors when there endpoints exceed a certain distance from the center. The optimization algorithm tries to group instances of the same class and places features with the same effects on the class together and features with opposite effects away from each other. This applies also to instances of different classes. More related classes will be places near each other and classes with different aspects will be places further away [Demsar et al. 2007]. Figure 2.3 shows an example visualisation of FreeViz with the classic zoology dataset.

2.4 Vectorized RadViz

With the ability to greatly deal with high dimensional datasets and the dimensional anchors that can flexibly placed, RadViz is a very important tool for Visualization. In this section we will talk about another extension for RadViz to analyse multiple clustered dataset. The extension Vectorized RadViz uses the advantages of this two major features of RadViz to visualize multiple clustered datasets or clusters. With VRV, each dimension can be partitioned into many more divisions. With more partitioned dimensions, each dimension can be positioned independently to each other. This helps to identify record patterns more clearly. VRV depends on ordering the data dimensions. To reorder the dimensions, first VRV uses a class discrimination layout algorithm and later a greedy algorithm to refine the placements of the dimensions [Sharko et al. 2008].

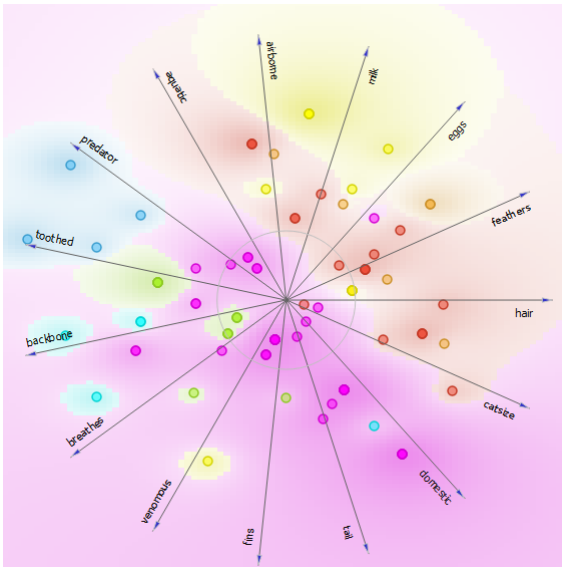
2.4.1 Methods

First you start with dataset which has been clustered several times. For each clustering, the data is extended by a new column. The values for that dimension are the cluster number each record is assigned for that cluster. From this extended cluster result dataset you can generate a VRV by following three steps [Sharko et al. 2008]:

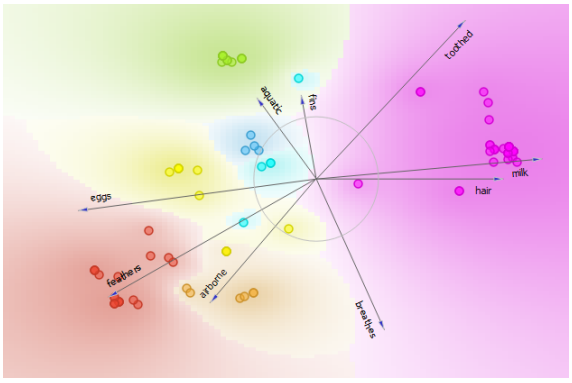
- expanding the number of dimensions by partitioning
- rearrange these dimensions into groups with the most similar characteristics
- relocate the most similar groups near each other

2.4.2 Analysis

VRV can be an effective tool to analyze multiple clustered datasets. It can provide the ability to visualize different important characteristics of multiple clustered datasets simultaneously. RadViz is very effective to display highly dimensional datasets and can place dimensions anywhere on the circumference of the circle because of its flexibility. Therefore increasing the number of dimensions with VRV and reordering the dimensions can be done without any complications. With VRV analysts can manipulate the individual values of a dimension and with that the placement of the dimensions in a highly flexible manner [Sharko et al. 2008].



(a) Initial Step



(b) Final step

Figure 2.3: FreeViz Example Visualization. [Figure created by the author of this paper using the tool Orange3.]

2.5 Gravi++

Gravi++ is a tool to explore time-oriented data and to identify predictors supporting the model building process [Hinum et al. 2005]. The idea is to provide functionalities to get new insights into the data to find clusters or similarities in attributes with the movement over time.

Concept

Gravi++ uses two different types of icons for the visualization. The first kind of icons represent the different dimensions (features) and the second the datapoints. The datapoints are attracted by the dimensional icons according to the values they have. This is based on a spring model like RadViz. Every dimensional anchor is connected with every datapoint by a spring. The strength of the values influences the strength of the individual springs. This leads to a formation of clusters of datapoints with similar features. The size of the icons can be mapped to the attraction force. The sphere is larger if it is attracted by higher values. This helps to discriminate different icons that are attracted by the same values with different coefficients. With this model, different datapoints can end up on the same positions. To distinguish different icons, the icons can be displayed by different transparent colors [Hinum et al. 2005].

Gravi++ uses animation to visualize changing values over time. The position of the icons of the datapoints change over time and allow to trace, compare and analyse the changing values. The change

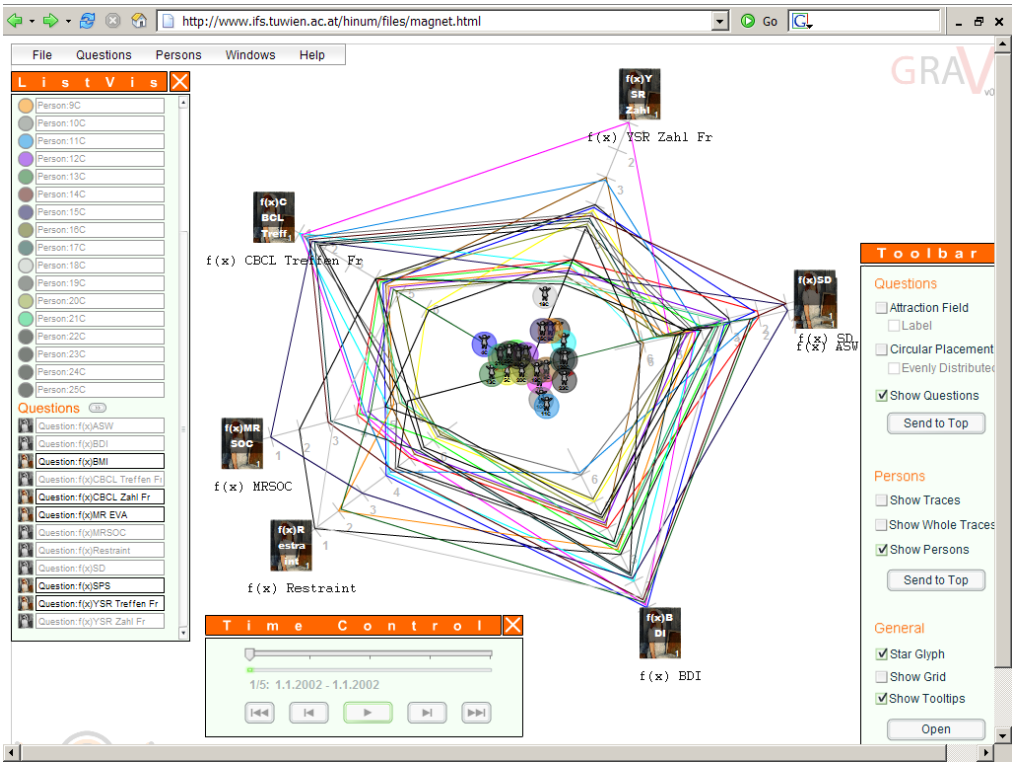


Figure 2.4: Gravi++ showing survey results from patients. [Screenshot taken and kindly provided by Keith Andrews.]

over time can also be represented by traces. The size and path can be show for all timestamps or a subset.

Rings can be drawn around the dimensional icons to visualize the exact values. The size of the rings comply with the attraction to the dimension. In addition, star glyphs can be shown to show the exact values. "The edges of the Star Glyphs are connected with the corresponding rings and both are drawn in the same color as the datapoint icon – [Hinum et al. 2005]". This is done to help identifying the matched datapoint. Gravi++ is more suited for a restricted parameter space, since the more dimensions are used, the smaller the influence of a single dimension on the position of the datapoint icon gets. It can also increase the probability of too many overlapping icons [Hinum et al. 2005].

2.6 GBC Plot - General Barycentric Coordinates Plot

This radial projection technique is based upon general barycentric coordinates. Besides mapping data records into a convex polygon built by data features, it offers multiple optimization approaches and other features to better interact with the data.

2.6.1 General Barycentric Coordinates

General barycentric coordinates are an extension of the well known barycentric coordinates to be well defined for irregular n-sided convex polygons [Meyer et al. 2002].

Meyer et al. [2002] define three basic properties that need to hold to be considered general barycentric coordinates. The first of these properties states that the mapping needs to be an affine combination. The second called smoothness, requires infinite differentiablity with regard to the point P and the vertices of the polygon. Lastly it must be a convex combination to guarantee no under- or over-shooting in the coordinates. Meyer et al. [2002] formulate the calculation of the weights for a point as

$$\omega_j = \frac{cot(\gamma_j) + cot(\delta_j)}{\|p - q_j\|^2} \tag{2.1}$$

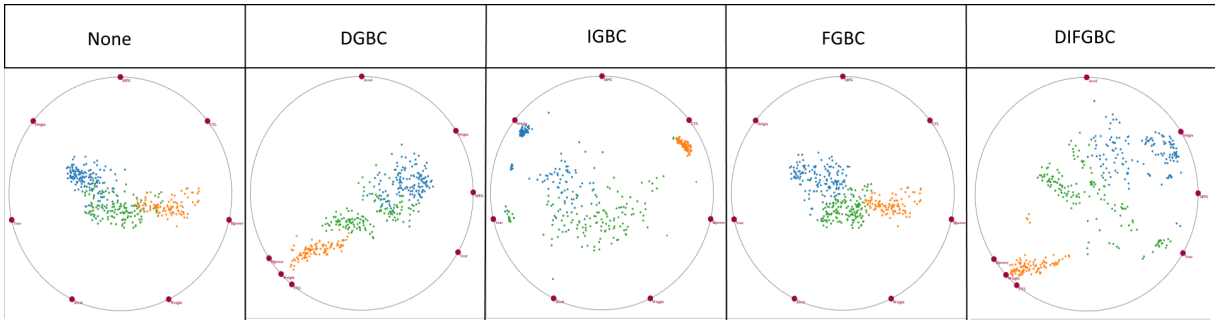


Figure 2.5: The cars dataset mapped by the GBC plot with different optimization steps enabled.
[Figure created by the authors of this paper using the tool developed by Cheng and Mueller [2015]]

,where p is the point being mapped, q_j is a vertex of the convex polygon, γ_j is the angle between the vectors $\overline{pq_j}$ and $\overline{q_j q_{j-1}}$, and δ_j is the angle between the vectors $\overline{pq_j}$ and $\overline{q_j q_{j+1}}$. Meyer et al. [2002] further proof that the three basic properties hold for this formulation. The technique developed by Cheng and Mueller [2015] assigns each vertex in the convex polygon a data feature. The GBC formulation is utilized in a reverse fashion to locate a data record inside the polygon [Cheng and Mueller 2015]. There have been three optimization steps implemented to further improve the data layout in the GBC plot, as seen in figure 2.5.

2.6.2 Optimizations

The first one, Distance Spaced GBC Plot Layout , tries to improve the ordering of the features to reduce the error of the mapping and have more well defined clusters. The approach rearranges the vertices based on an approximate Traveling Salesman Problem solver [Cheng and Mueller 2015].

The second optimization, Iterative GBC Plot Error Reduction, tries to optimize the location of the data record inside the convex polygon. This is done iteratively calculating an error polygon of a data record in regard to every data feature and decreasing its size [Cheng and Mueller 2015]. This approach further improves the clustering of the data, however it has less of an impact than the previous technique [Cheng and Mueller 2015].

The third approach, Force Directed GBC Plot Adjustment, also adjusts the location of data records. This is achieved via a multidimensional scaling (MDS) scheme called force directed layout [Cheng and Mueller 2015]. The results for this approach depended on the dataset. Whereas clustering was improved with two of the datasets used by Cheng and Mueller [2015], there was next to no improvement for the third.

All these optimization approaches are based on different error measures. These error measures are based upon three different distance measures one can take from the GBC plot. These distance measures are [Cheng and Mueller 2015]:

- data record to data record
- data record to data feature
- data feature to data feature

The first approach optimizes the data feature to data feature error, whereas the second improves upon the data record to data feature error and the last approach optimizes data record to data record error.

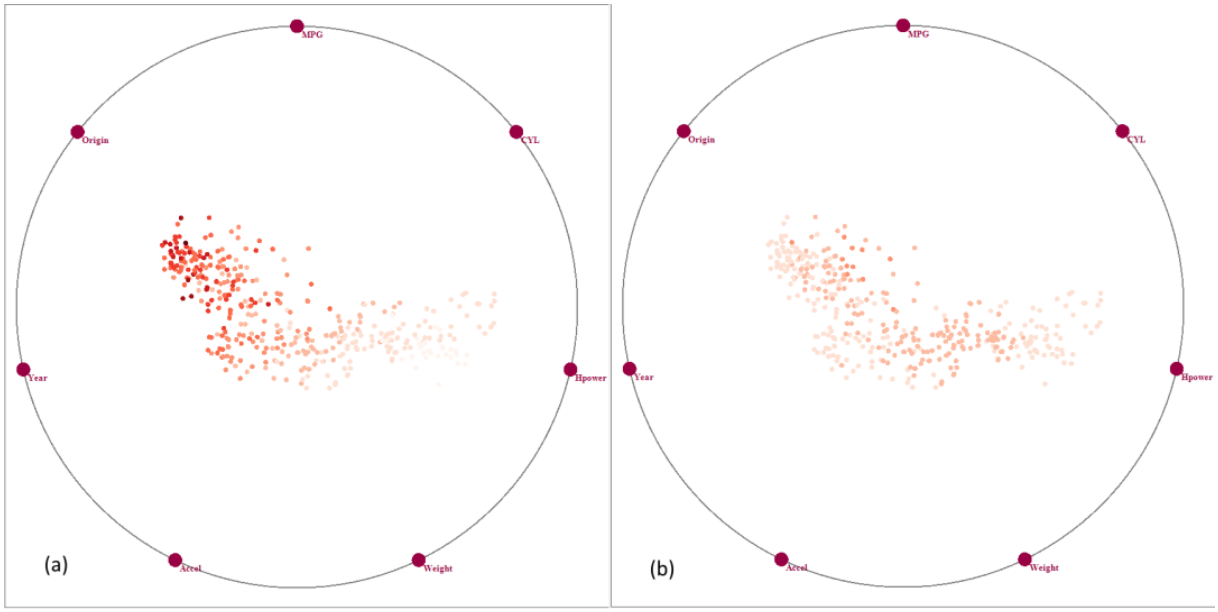


Figure 2.6: Example of the distance coloring in (a) and the error based coloring int (b) [Figure created by the authors of this paper using the tool developed by Cheng and Mueller [2015].]

2.6.3 Additional Features

There are additional features provided with these error and distance measures. Firstly in the plot itself the color of the mapped data records can be changed with regard to the distance measure or the error measure as shown in 2.6.

Furthermore there are a distance heat-map panel, and an error visualization panel, both seen in figure 2.7. These color-code and display all the distance measure matrices and error measure matrices respectively [Cheng and Mueller 2015]. The tool also includes a parallel coordinates display, that shows the data mapped with the parallel coordinate technique. Finally, one can select subsets of data records or data features to improve upon the layout locally [Cheng and Mueller 2015].

2.7 Dust and Magnet

Dust and Magnet is a multidimensional data visualization tool that operates with the principal idea of ferrous dust and magnets in mind. Data features are magnets, and data records are represented as dust. This dust is attracted to the magnets over time, and from their movement and final placement one can reason about the data, as showcased in figure 2.8.

This very basic metaphor grounded in the physical world makes the interaction between data and dimensions very easy to understand and significantly lowers the barrier of entry for laymen in the field [Soo Yi et al. 2005]. Dust and Magnet can be seen as an extension to Star Coordinates. Magnets are the base vectors laid out freely by the user. The mapping of the data records differs by having an additional factor that describes the attraction of magnet and dust particles [Cheng and Mueller 2015]. An important part of the tool is animation over time. Magnets can be placed freely in the scene and after initial placement can be dragged around, to see how they influence the dust particles. During this dragging process the dust particle positions are continuously updated. From this movement one can argue about the data [Soo Yi et al. 2005].

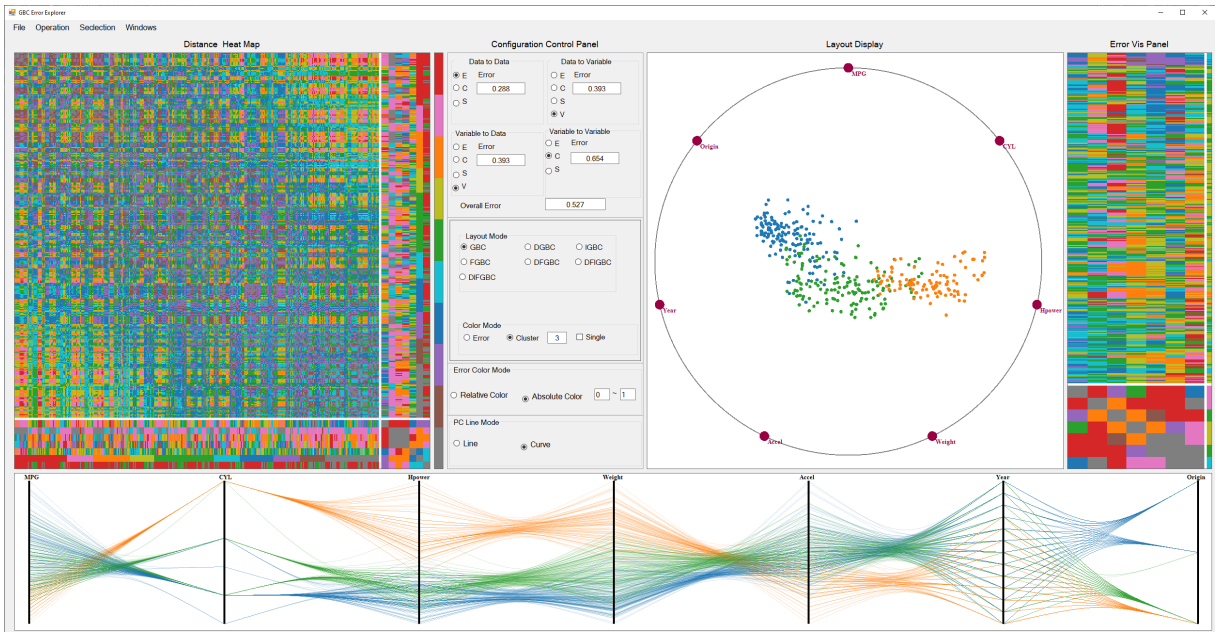


Figure 2.7: The GBC plot tool, with all of it's panels. [Figure created by the authors of this paper using the tool developed by Cheng and Mueller [2015].]

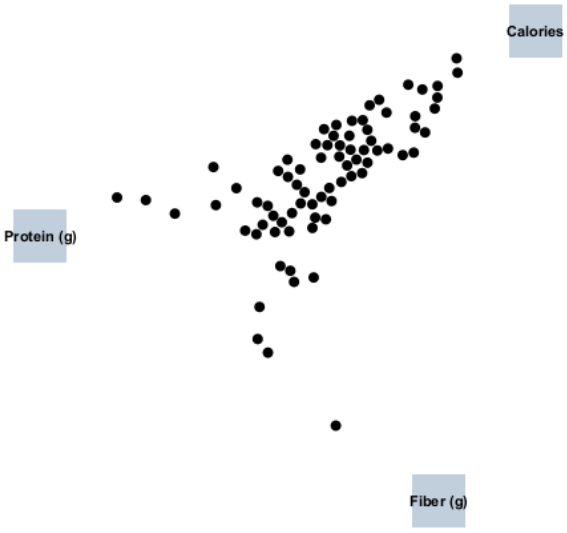


Figure 2.8: A basic scene in the dust and magnets tool. [Figure taken by the authors of this paper using the tool developed by Ji Soo Yi.]

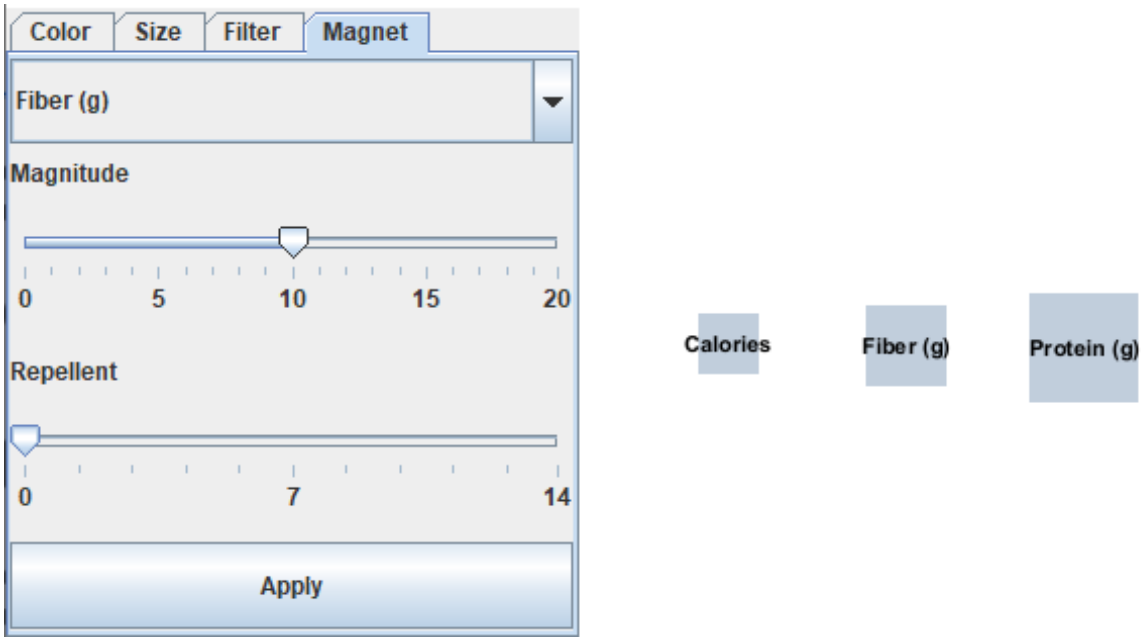


Figure 2.9: Changing the magnitude of attraction for a magnet and its impact in the simulation window. [Figure created by the authors of this paper using the tool developed by Ji Soo Yi.]

2.7.1 The Data

Data features are separated into three categories: nominal, ordinal and quantitative [Soo Yi et al. 2005]. Nominal features can not be used as magnets as their attraction values can not be calculated. These include things like names. Ordinal features are categories of values. As these categories can be mapped into, for example the natural numbers, one can quantify them and use them as magnets. Finally quantitative features are ranges of values that do not need any remapping to be quantified. These features are min-max normalized to remove the difference of scale such features might have. As the technique gives a coarse data representation at any point one can interact with a particle in the simulation. Selecting a particle in this manner shows the exact values in the detail view.

2.7.2 Interacting with Dust and Magnet

The tool provides additional features in regards to magnets and dust. These additional features mitigate problems a very simplistic implementation would have, or enhance the experience to allow for more detailed data analysis.

Magnets

For any magnet one can change the magnitude of attraction, this change is also shown by a size change in the simulation window as seen in figure 2.9. Changing the magnitude changed the overall impact the magnet has on the simulation. Counter to the physical dust and magnets metaphor one can also make magnets repellent. Making a magnet repellent would mean that any dust particle with a value lower than the repellent factor would now be repelled from the magnet instead of attracted, helping one further separate data.

Dust

Dust is simulated only in regards to magnets and there is no interaction between individual dust particles. This impliest that occlusion of dust particles may occur. Soo Yi et al. [2005] allow for this occlusion to happen as it is a natural result of the attraction that particles end up at the same point in space. However overlap is undesired as data points are ‘lost’. This includes the ability to interact with them. To mitigate

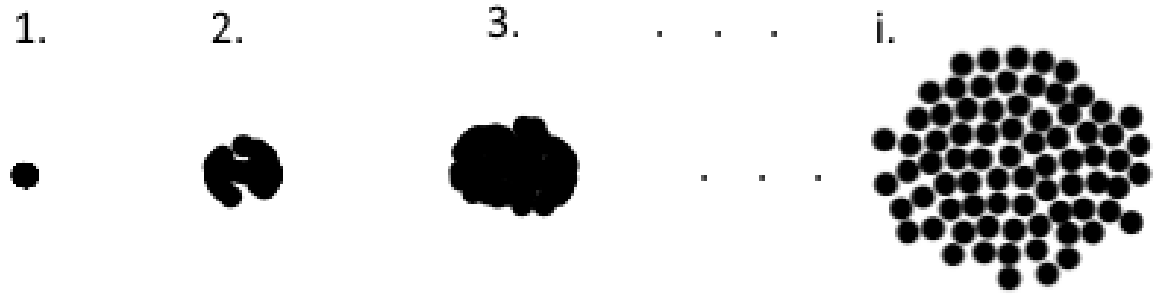


Figure 2.10: The iterative process of spreading out dust. [Figure created by the authors of this paper using the tool developed by Ji Soo Yi.]

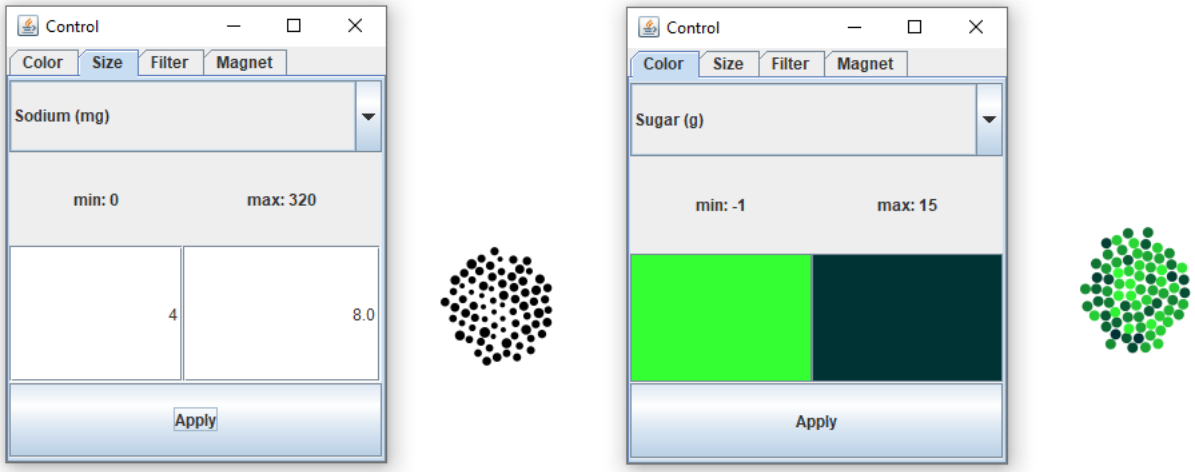


Figure 2.11: The effects of changing the size or the color of dust particles. [Figure created by the authors of this paper using the tool developed by Ji Soo Yi.]

this a 'Spread Dust' feature exists. As seen in figure 2.10 this iteratively spreads out the dust until there is no overlap anymore [Soo Yi et al. 2005].

Reproducibility is a further issue the tool has. As the final position of a dust particle is highly dependent on the magnet movement over time, one would need to almost perfectly retrace one's workflow to get the same result. There are two additional features to deal with the issue of reproducibility. Besides animating dust by dragging around magnets one can also manually step through the animation at small movements. The maximum distance a particle can move by manually updating its position is two units [Soo Yi et al. 2005]. With the 'Center Dust' feature one can reset dust to its initial position to easily restart the simulation.

Besides these features dealing with the simulation directly there exist further helpful additions that allow for more distinct dust particles and situations. One can filter dust, selecting subsections of the data. This removes clutter from the simulation and makes it easier to focus on specific data ranges. To more easily differentiate between data one can change the size and the color of dust particles as seen in figure 2.11. These changes, if for a quantitative value, are defined by a range which gets linearly interpolated and applied to the particle.

Chapter 3

RadViz as compared to Star Coordinates

RadViz and Star Coordinates follow a very similar basic approach when calculating the mappings, the only difference is the normalization step that is added by RadViz. This additional step affects the capabilites of visualizing certain properties of the data, like clusters, lines, spheres and outliers. Rubio-Sánchez et al. [2016] compared RadViz and Star Coordinates in detail and rated their suitability for such tasks. We will summarize their findings in the following sections.

3.1 Mapping lines

Both Star Coordinates and Radviz map lines in the high-dimensional space to lines in the 2D representation. In addition to that, Star Coordinates mapping maintains a uniform distribution of points on a line, which is not true for RadViz since the normalization step affects the distribution.

3.2 Clustering

The normalization used in RadViz leads to a very undesired property. The visualization of clusters highly depends on their size and also on their location in the data space. Clusters that are located close to the origin will appear much larger in the final visualization. Star Coordinates do not introduce this kind of problem since they always provide a linear mapping.

3.3 Sparse data

RadViz was designed with the primary focus on highlighting the case that one dimension of a data entry is significantly larger then all the others, which makes it the obvious choice when looking for sparse data points. Such points will always end up close to the respective anchor point, and it is clearly visible which dimension contains the high value. Star Coordinates will also map these points to locations with a large distance between the origin and the projection to the respective axis, but it is not obvious why the point ended up at this position since high values in different dimensions could have produced the same result.

3.4 Outliers

Star Coordinates perform better when trying to spot outliers since RadViz maps all data points inside the convex hull defined by the anchor points, which in many cases does not allow distinces high enough to recognize outliers. Radial projection methods usually have drawbacks when visualizing outliers since their dimensional reduction introduces the need for additional steps like reordering or resizing the dimensions to clearly show outliers, making finding them a non-trivial task.

3.5 Value estimation

Both techniques map data entries to points with lower dimensions, making exact recovery from the plot impossible. Star Coordinates provides an easy way to estimate the original values by normal projection from the data points to the coordinate axes, which yields better accuracy than trying to estimate the values from a RadViz visualization.

3.6 Summary

While the normalization step introduced by RadViz is very useful when working with sparse data, it reduces the quality of the plot regarding many other important aspects. If the data does not contain sparse records or analyzing this aspect is not in the focus of interest, Star Coordinates will very likely provide better results.

Chapter 4

Case Study

A case study was performed by the authors of this survey. This case study was done using existing tools that allow users to apply Dust and Magnet, RadViz, and Star Coordinates visualization.

4.1 The Data Sets

For this case study we decided on two publicly available datasets. The first is the cereals dataset, the second a dataset from the styrian government.

The cereals dataset is a classic dataset used by many to explore their newly developed tools. We optained the dataset from kaggle[1]. It includes about 78 data entries with 16 dimensions. These dimensions include things like, name of the cereal, sugar contained, calories and many more. Not all of these dimensions were used by us during our case study, as seen in figure 4.1, which shows the dimensions used most commonly. In most cases we ended up with around 12 dimensions after removing some data features.

The second dataset is a styrian employment dataset from the year 2017, directly optained from the government website[2]. There are 12 dimensions and 288 data records. The data are employment, unemployment and labor force numbers split according to biological sex, showcased in figure 4.2. The original dataset further included multiple identifiers for the data records. However, we removed those, as a single identifier was enough, which was chosen to be the name of the municipality.

4.2 Tasks

To effectively compare the tools with one another, at the start of the case study four tasks were agreed upon. These tasks are:

- *Task 1:* Cluster cereals according to healthy and unhealthy options.
- *Task 2:* Is there a correlation between sugar and calories?
- *Task 3:* Cluster the data into high and low unemployment.
- *Task 4:* Is there a correlation between male employment and female unemployment?

With these tasks we want to gauge the effectiveness of each tool in this respective area on two seperate datasets. However we do not presume that the conclusions we derive from our exploration are correct and overlap with what actual research into the topic at hand might have found. These conclusions should be seen as isolated results and are not meant to verify or deny any claim in any way.

[1] kaggle.com/semakulapaul/cereals-dataset#cereal.csv

[2] data.steiermark.at/cms/beitrag/11822084/97108894/?AppInt_OGD_ID=976

	name	calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	weight	cups	rating
1	100% Bran	70	4	1	130	10	5	6	280	25	1	0.33	68.402973
2	100% Natural Br	120	3	5	15	2	8	8	135	0	1	1	33.983679
3	All-Bran	70	4	1	260	9	7	5	320	25	1	0.33	59.425505
4	All-Bran with Ext	50	4	0	140	14	8	0	330	25	1	0.5	93.704912
5	Almond Delight	110	2	2	200	1	14	8	-1	25	1	0.75	34.384843
6	Apple Cinnamon	110	2	2	180	1.5	10.5	10	70	25	1	0.75	29.509541
7	Apple Jacks	110	2	0	125	1	11	14	30	25	1	1	33.174094
8	Basic 4	130	3	2	210	2	18	8	100	25	1.33	0.75	37.038562
9	Bran Chex	90	2	1	200	4	15	6	125	25	1	0.67	49.120253
10	Bran Flakes	90	3	0	210	5	13	5	190	25	1	0.67	53.313813
11	Cap'n'Crunch	120	1	2	220	0	12	12	35	25	1	0.75	18.042851
12	Cheerios	110	6	2	290	2	17	1	105	25	1	1.25	50.764999
13	Cinnamon Toast	120	1	3	210	0	13	9	45	25	1	0.75	19.823573
14	Clusters	110	3	2	140	2	13	7	105	25	1	0.5	40.400208
15	Cocoa Puffs	110	1	1	180	0	12	13	55	25	1	1	22.736446
16	Corn Chex	110	2	0	280	0	22	3	25	25	1	1	41.445019
17	Corn Flakes	100	2	0	290	1	21	2	35	25	1	1	45.863324
18	Corn Pops	110	1	0	90	1	13	12	20	25	1	1	35.782791

Figure 4.1: A sample of the classic cereals dataset seen in a spreadsheet. [Figure created by the authors of the survey]

	DISTRICT_COD	DISTRICT_NAM	LAU_NAME	EMPL_M	UNEMPL_M	LABOUR_FORCE_M	EMPL_W	UNEMPL_W	LABOUR_FORCE_W	EMPL_TOTAL	UNEMPL_TOTAL	LABOUR_FORCE_TOTAL
1	601	Graz-Stadt	Graz	72239	6978	79217	64722	5460	70182	136961	12438	149399
2	603	Deutschlandsber	Frauental an der Laßnitz	788	35	823	682	35	717	1470	70	1540
3	603	Deutschlandsber	Lannach	1020	28	1048	897	37	934	1917	65	1982
4	603	Deutschlandsber	Pölfing-Brunn	390	25	415	332	25	357	722	50	772
5	603	Deutschlandsber	Preding	495	26	521	421	17	438	916	43	959
6	603	Deutschlandsber	Sankt Josef (Weststeiermark)	449	14	463	404	16	420	853	30	883
7	603	Deutschlandsber	Sankt Peter im Sulmtal	339	21	360	268	11	279	607	32	639
8	603	Deutschlandsber	Wettmannstätten	440	21	461	380	15	395	820	36	856
9	603	Deutschlandsber	Deutschlandsberg	3004	155	3159	2691	141	2832	5695	296	5991
10	603	Deutschlandsber	Eibiswald	1711	73	1784	1430	60	1490	3141	133	3274
11	603	Deutschlandsber	Groß Sankt Florian	1189	41	1230	1003	41	1044	2192	82	2274
12	603	Deutschlandsber	Sankt Martin im Sulmtal	857	47	904	682	44	726	1539	91	1630
13	603	Deutschlandsber	Sankt Stefan ob Stainz	993	34	1027	901	23	924	1894	57	1951
14	603	Deutschlandsber	Schwanberg	1248	33	1281	1079	56	1135	2327	89	2416
15	603	Deutschlandsber	Stainz	2381	86	2467	2089	87	2176	4470	173	4643
16	603	Deutschlandsber	Wies	1200	62	1262	997	54	1051	2197	116	2313
17	606	Graz-Umgebung	Feldkirchen bei Graz	1833	104	1937	1554	76	1630	3387	180	3567
18	606	Graz-Umgebung	Gössendorf	1133	44	1177	993	50	1043	2126	94	2220
19	606	Graz-Umgebung	Gratkorn	2110	115	2225	1908	128	2036	4018	243	4261

Figure 4.2: A sample of the styrian employment dataset seen in a spreadsheet. [Figure created by the authors of the survey]

4.3 Dust and Magnet

To conduct the Dust and Magnet case study we decided on using the tool developed by Ji Soo Yi, author of the original paper. The tool can be found on his personal github page[3]. This version of the tool boasts an additional feature compared to the tool used in the original paper, namely snapshots. This feature allows one to take a snapshot of the settings and layout, and return to it at any point in time. If a promising data layout has been found, it is possible to take a snapshot recording all the positions of dust and magnets, as well as any further settings one might have changed. If after further investigating one believes themselves to be in a less helpful state as previously in, it is now possible to revert to the previous state with the snapshot.

4.3.1 Task 1

Task one was the most straight forward out of all of them using this tool. One attempt to solve this task is by placing healthy magnets on one side and unhealthy on the opposite. To better differentiate coloring was added based on potassium content of the cereals, as seen in figure 4.3.

With a setup like this a quite prominent cluster of unhealthy cereal option forms, and a few healthy options emerge as well. Placing the requirements for the clustering on opposite ends of the scene and simulating them seems to be a natural first guess. This is corroborated by the findings of Soo Yi et al. [2005] with their findings for a similar task.

[3] github.com/yijisoo/DnM

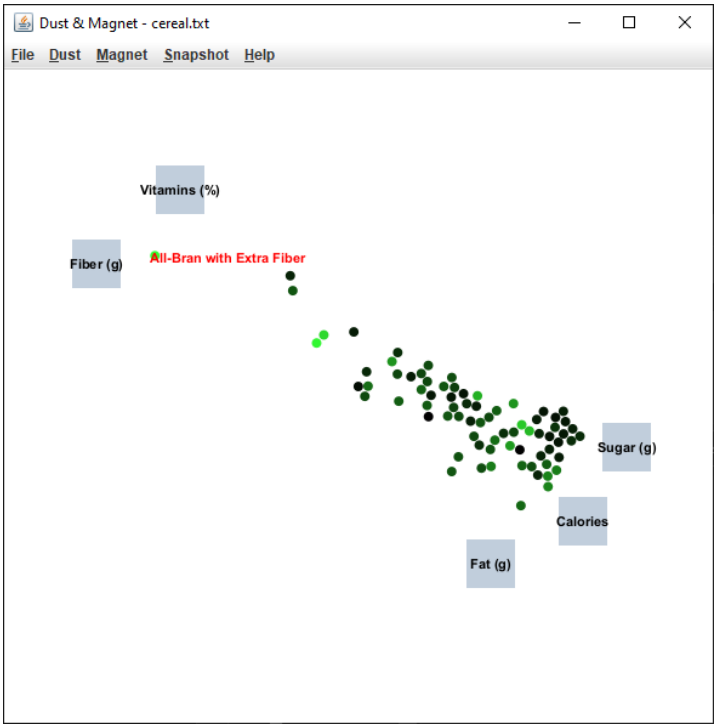


Figure 4.3: A solution for task 1 using Dust and Magent. [Figure taken by the authors of this paper using the tool developed by Ji Soo Yi.]

4.3.2 Task 2

The second task with the cereals dataset was perceived to be a lot harder to complete. A solution used coloring and a single magnet, the calories one, to solve the task. The dust was colored from green to red, as seen in figure 4.4 where green indicates low sugar and red high sugar content.

After moving the magnet around it was concluded that there was a slight correlation between high sugar content and calories, shown in figure 4.5. A failed attempt included placing the magnets for sugar and calories in the scene and trying to see how movement is affected by moving the magnets individually. This was quickly abandoned, the final layout can be seen in figure 4.6. As one can see no correlation can be read from this setup.

4.3.3 Task 3

Using the knowledge gained from task 1, task 3 was quickly dealt with. The magnets for total employment and total unemployment were placed on opposite sides and the attraction was simulated. In general one can see in figure 4.7, that there are a few outliers with very high employment and low unemployment.

Similarly there were only a few outliers closer to the total unemployment magnet. However the majority of the dust remained in the center suggesting a balanced unemployment-employment number between all data records, trending towards higher employment than unemployment. Notable was the fact that for significant movement of the data one needed to zoom out quite far. This resulted in the fact that the magnet labels disappeared during the simulation making the movement a little less obvious as one needs to remember what magent is currently dragged. In a complex scene with many magnets this might get confusing. It seems that larger datasets need more zoomed out scenes to see a lot of movement and have dust particles that are not all on top of one another.

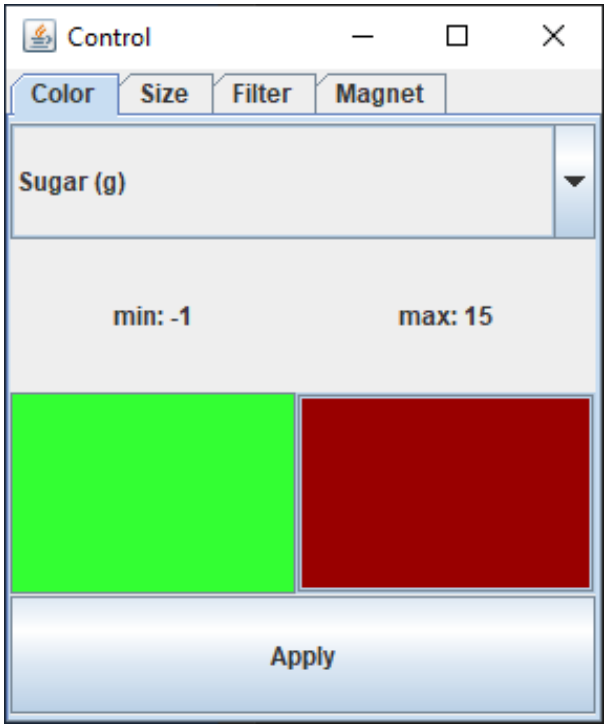


Figure 4.4: The coloring of the dust particles for task 2. [Figure taken by the authors of this paper using the tool developed by Ji Soo Yi.]

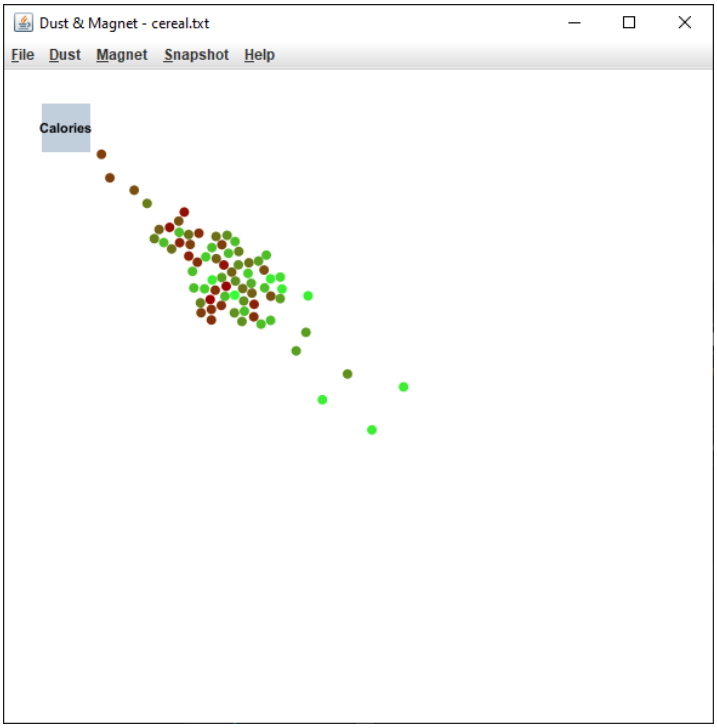


Figure 4.5: The result of using only one magnet and coloring the dust according to sugar content. [Figure taken by the authors of this paper using the tool developed by Ji Soo Yi.]

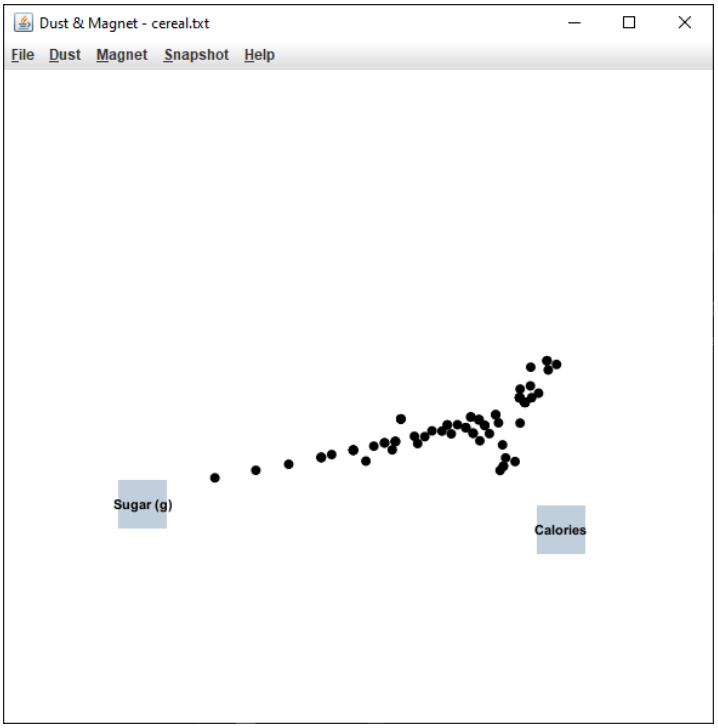


Figure 4.6: The final layout of a failed attempt to solve task 2. [Figure taken by the authors of this paper using the tool developed by Ji Soo Yi.]

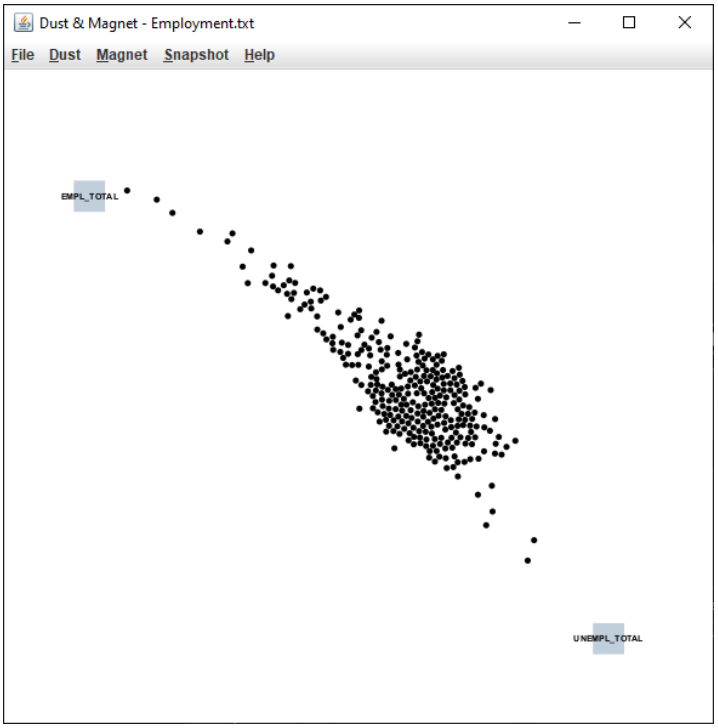


Figure 4.7: Layout after seperating employment- and unemployment-totals. [Figure taken by the authors of this paper using the tool developed by Ji Soo Yi.]

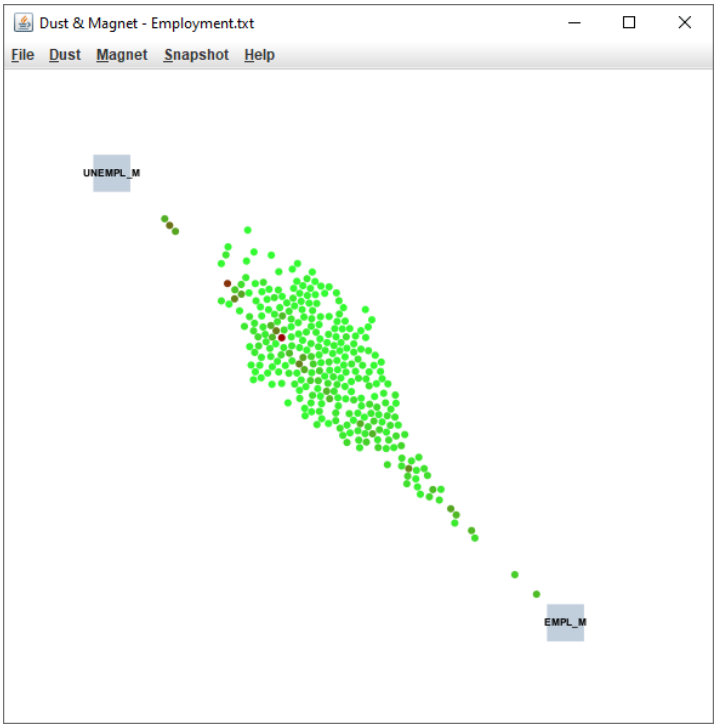


Figure 4.8: Attempt to find correlation between male emplyoment and female unemployment. [Figure taken by the authors of this paper using the tool developed by Ji Soo Yi.]

4.3.4 Task 4

Even though familiarity with the tool was gained from task 2, task 4 was by far the most difficult one. In the end an arguably successful attempt was made by placing male employment and unemployment on opposite sides, and coloring based on female unemployment, as seen in figure 4.8.

Placing magnets like this lets one conclude that there was no significant correlation between male employment and female unemployment. Rather, high male unemployment, also correlated with higher female unemployment. One failed attempt included placing the magnet of unemployed men close to the magnet of employed women. On the opposite side of the scene the magnet of employed men and unemployed women was placed, as seen in figure 4.9. Additionally to this dust was colored according to female unemployment, and resized based on male employment. Doing all this had the result of making it hard to read what a single dust particle represented. The magnet placement resulted in very minimal movement of the dust overall. It was concluded that no statement about the correlation could be made and a new attempt was started.

4.3.5 Summary

From all the tasks the clustering exercises were the easiest to complete. One just needed to place the magnets in a very obvious manner and simulate the attraction. The correlation tasks were a lot harder to complete. A similar task about correlation was completed in the paper by Soo Yi et al. [2005], however even with the knowledge of how this task was solved most correctly, one had to think a lot of how to color and place the magnets before trying it out. Without this knowledge it could be quite taunting to complete this task. Especially as with just magnet attraction it seems quite impossible to find out correlation and one needs to think about every feature of the tool available to mold a setup that can work.

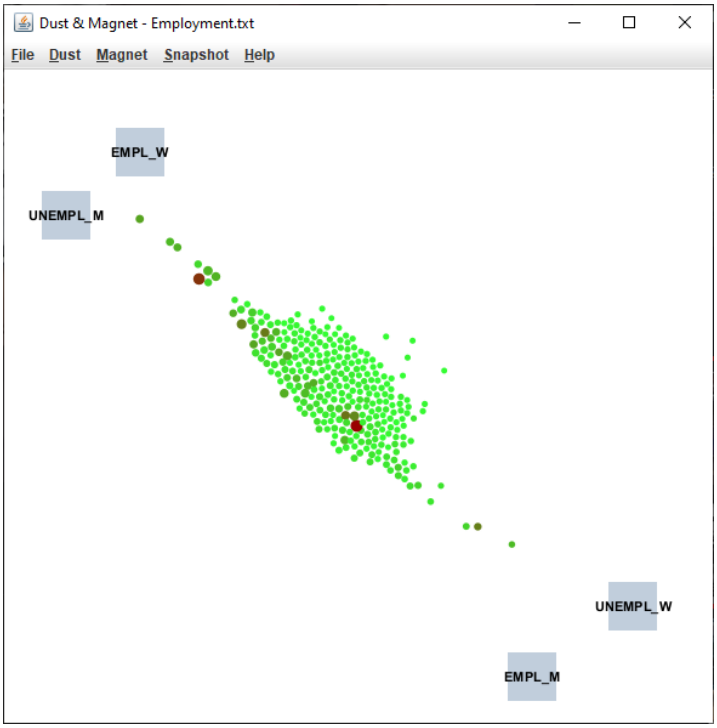


Figure 4.9: An inconclusive try for task four using Dust and Magnet. [Figure taken by the authors of this paper using the tool developed by Ji Soo Yi.]

4.4 RadViz

We used the RadViz-X tool provided by Hoffman et al. [1997]. The tool is available as a .exe file for Windows and works without additional dependencies. It provides all the necessary features like column manipulation, size mapping and color mapping we are going to need to complete the four tasks.

4.4.1 Task 1

Since we want to find healthy cereals, we want to separate the ones high in vitamins from the ones high in sugar and fat. We remove the columns we are not interested in since we no not want them to affect the point positions and order the columns in a way that place the vitamins on one side of the circle, fat and sugar on the opposite side. Additionally, we make use of two additional features the RadViz-X tools provides. We scale the size of the mapped points according to the amount of sugar the corresponding cereal contains, and color the points in different colors depending on the amount of vitamins. As shown in figure 4.10, we can easily spot the healthy cereals that are attracted to the bottom left part of the circle.

4.4.2 Task 2

When trying to find out if several dimensions of the data records contain similar values, RadViz's normalization step comes in handy. By definition, records that contain similar values accross all dimensions are mapped close to the center. If one dimension is much higher than all the others the point will be mapped close to the respective anchor point. All we need to do to visualize the relation between sugar and calories in the cerals is reducing the visible columns to just these two and see where the points end up. We make use of the color and size mapping as well to enhance the visualization. In figure 4.11 we see the final result. While the major part of the points is located close to the center, there are some cereals that contain a high amout of calores while they are quite high in sugar, resulting in their points being pushed to the left side. We can conclude that cereals that are high in sugar are likely to be high in calories as well since no points are pushed rather close to the right side, but the opposite is not true. There are cereals that contain just a small amount of sugar but are still high in calories.

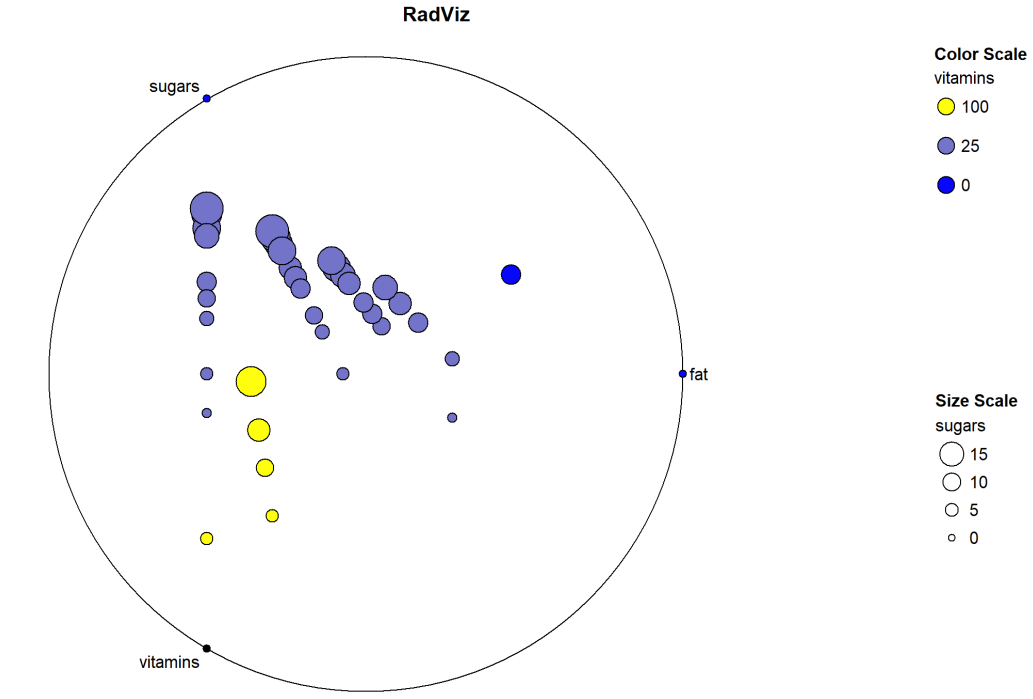


Figure 4.10: Healthy cereals are located in the bottom left part of the circle. [Figure taken by the authors of this paper using RadViz-X.]

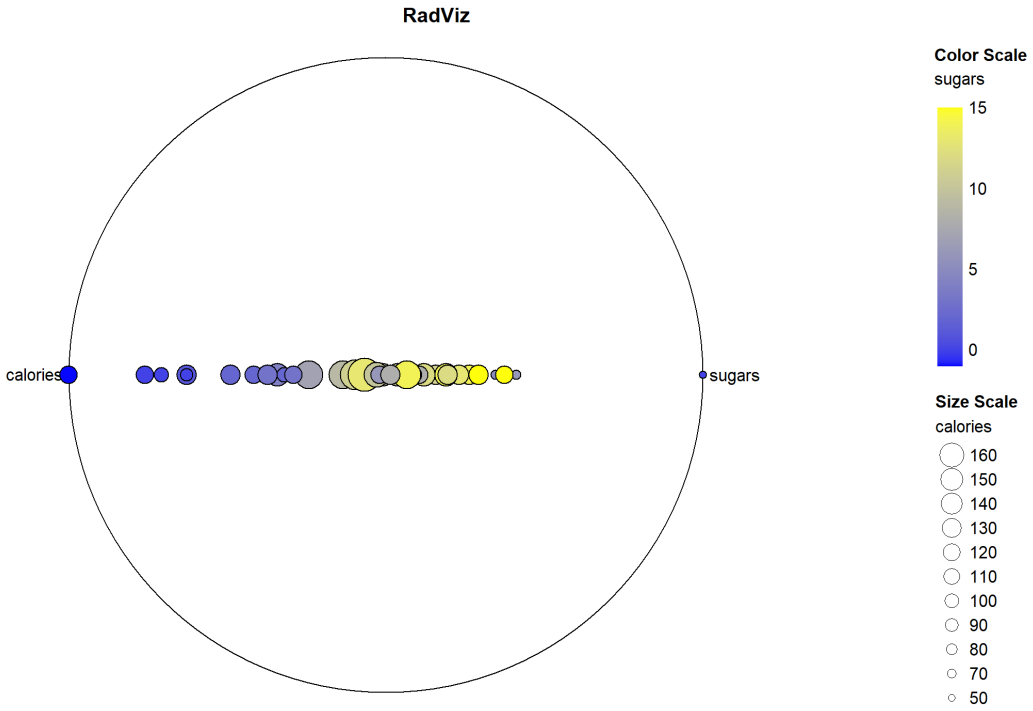


Figure 4.11: Similar values in both dimensions result in mappings close to the center. [Figure taken by the authors of this paper using RadViz-X.]

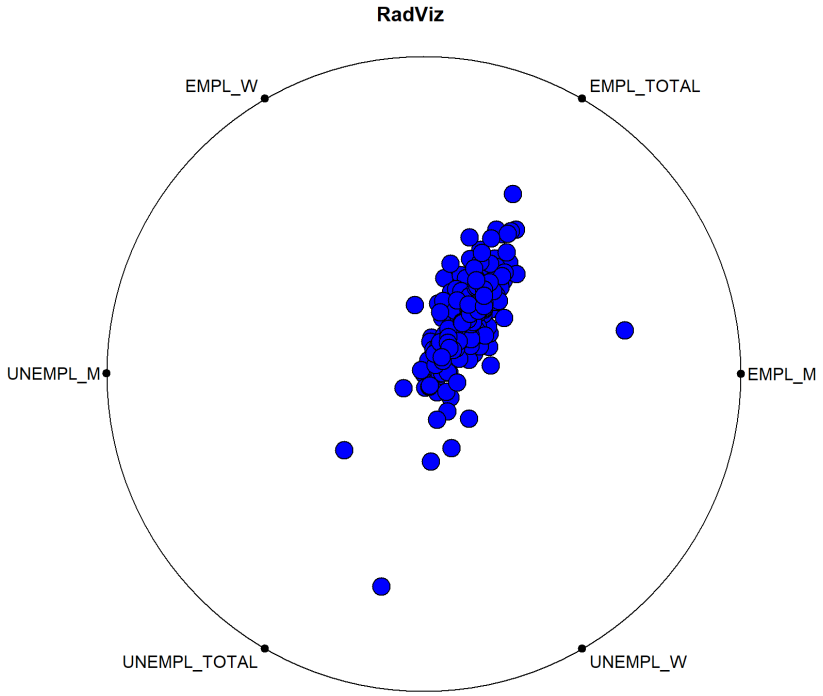


Figure 4.12: No clustering with RadViz-X, but two interesting outliers. [Figure taken by the authors of this paper using RadViz-X.]

4.4.3 Task 3

The easiest approach to show the amount of employed and unemployed workers would be to simply select the columns containing the total amount of employed and unemplyoed workers. Using the available columns containing the respective numbers divided into male and female workers, we might spot some additional details while still achieving the desired separation. We place the three columns containing male, female and total employed workers opposite to the male, female and total unemployed workers and end up with the image showin in figure 4.12. While we cannot spot a clear separation, it is easy to see that more than half of the people are employed in most of the places. An interesting additonal observation are the two significant outliers. One town has a very high number of unemployed females, while another town has a very high number of employed males compared to the avarage.

4.4.4 Task 4

For the final task, we try to find a correlation between male employment and female unemployment. We again make use of the normalization and remove all the columns but the two we are intersted in. The image in 4.13 shows the result. The majority of the points end up in the right part of the circle, showing us that a high number of employed males does not imply a high number of unemployed female workers. In addition to that we can spot two points that are mapped at the maximum distance from the center, showing us towns with almost no unemployed women and a very high number of employed men on the right side. On the opposite side one can observe a town with a very high number of unemployed women, but a very low number of employed men.

4.4.5 Summary

The normalized mapping used in RadViz makes it very useful for analyzing similarites across dimensions and spotting outliers. Since it is not possible to increase the contribution of individual axes, trying to find clusters or specific relations often requries removing a lot of columns. This takes away the chance to spot

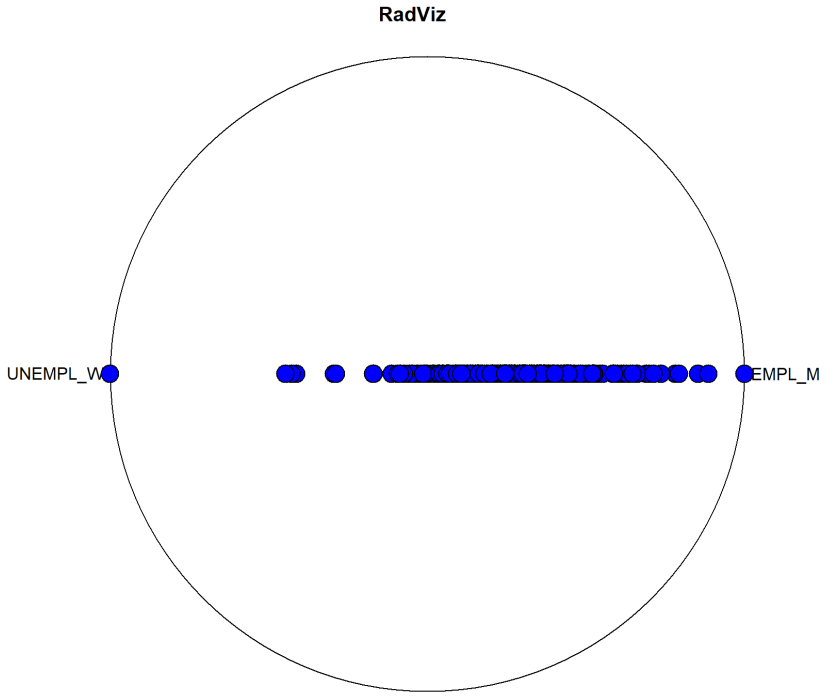


Figure 4.13: Visualization of Task 4. One can see two extrema mapping onto the anchor points. [Figure taken by the authors of this paper using RadViz-X.]

interesting, but unexpected patterns in the data. When attempting to spot clusters in the data with respect to specific dimensions while keeping an overview of all dimensions, RadViz is not a good choice.

4.5 Star Coordinates

For the Star Coordinates case study we used the tool InterStar. InterStar was developed by Kandogan [2001]. The tool was kindly provided by Eser Kandogan himself after we took contact with him per email. InterStar is written in Javascript and can easily be run after setting up a local http server. With InterStar the user is able to move the axis for different dimensions per drag and drop in the given canvas. The further the features are moved away from the center, the higher their impact gets. The features can also be moved near the center to reduce their impact on this dimension. It is also possible to entirely deactivate different dimensions. Deactivating nearly all dimensions and only letting a few dimensions active can lead to a problem of overlapping datapoints. Therefore, most of the time it is better to just let all dimensions active and move the ones which are not needed to the center. The tool provides also a feature for detecting clusters. While the user interactively explores the data and the datapoints are moving in the canvas, the tool tries to find clusters and if there is any, it will suggest them to the user with an annotation. If the user hovers with the mouse over this annotations, the respective datapoints in this cluster will be marked with a different color.

4.5.1 Task 1

Like with the previous tools, task 1 was really easy to solve. As unhealthy cereals, we defined cereals with high sugar and fat values, and low values on vitamins and fiber. To solve this task, we moved sugar and fat to one side and since fiber and vitamins should be low we moved those two dimensional axis to the other side. The rest of the features were moved to the center to reduce their impact. Figure 4.14 shows the final layout for this task and can we see that most of the cereals would be categorized as unhealty. Most of the datapoins lie on the right side near sugar and fat. These cereals have a high value in sugar

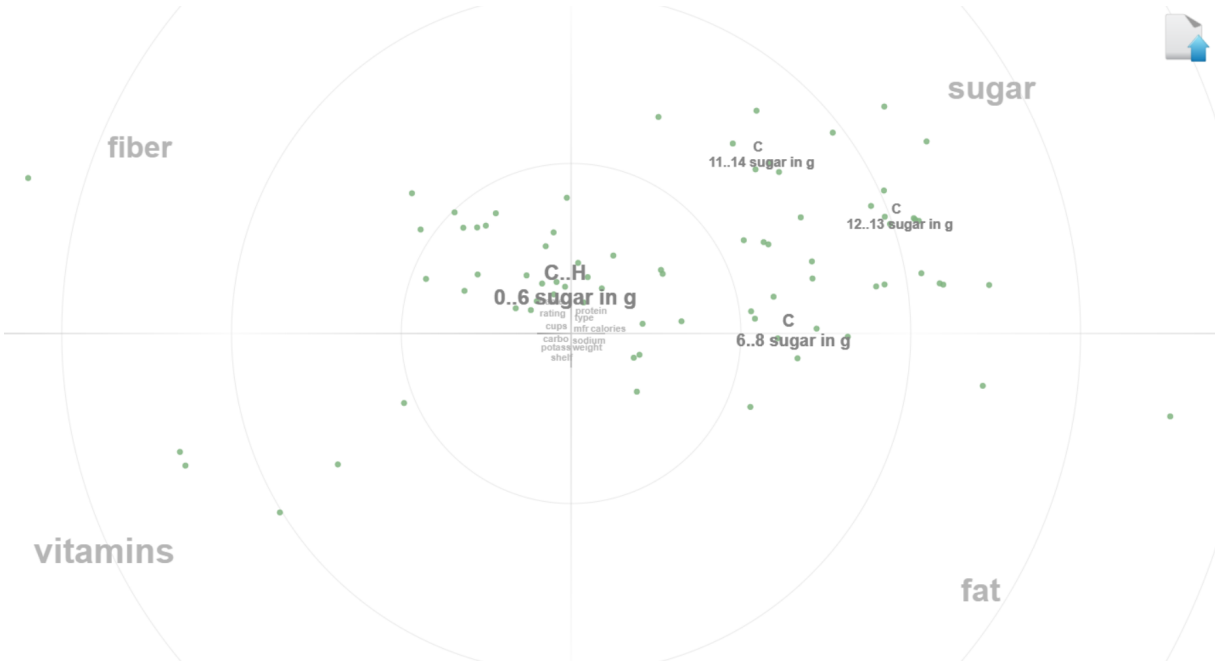


Figure 4.14: Finding healthy and unhealty cereals with Star Coordinates. [Figure taken by the authors of this paper using InterStar developed by Kandogan [2001].]

and fat and much lower values in vitamins and fiber. Only a few datapoints are on the left near vitamin and fiber. These would be the more healthier cereals. We can also see that the tool found a few clusters of cereals based on the sugar level.

4.5.2 Task 2

In this task we tried to find out if there is a correlation between sugar and calories. Since we only need these two dimensional axis to find a correlation, we moved the other dimensional axis to the center. Because we only need to observe two dimensions in this task, we arranged them like a basic 2D coordinate system. Looking on the final result of task 2 seen in figure 4.15, we can observe that there are many cereals with high sugar values and we can conclude that cereals with higher sugar values have also a higher value on calories. However this is not quite true the other way around. There exist a few cereals with high calories but still have low values in sugar.

4.5.3 Task 3

For task 3 to find clusters in high and low employment, we used the columns of employed and unemployed workers and of course the total employment and unemployment. The other features were moved to the center since they were not needed. In this approach with InterStar we encountered an unexpected problem. The dataset contains employment data of cities in Styria and most of these cities are very small. The biggest city of course is the city Graz. Graz can be immediately detected as an outlier in the final layout seen in figure 4.16. All other cities are stacked together in the center of the graph. This is because the huge difference in the populations and the Star Coordinates system scales the axis linearly from the minimum value to the maximum at the end of the axis. The city Graz has in comparison to the other cities in Styria a really huge difference in population and therefore a huge difference in employed and also unemployed people. The values of the remaining cities seem to be too insignificantly small compared to Graz and therefore the datapoints do not really change their positions in the Star Coordinates layout.

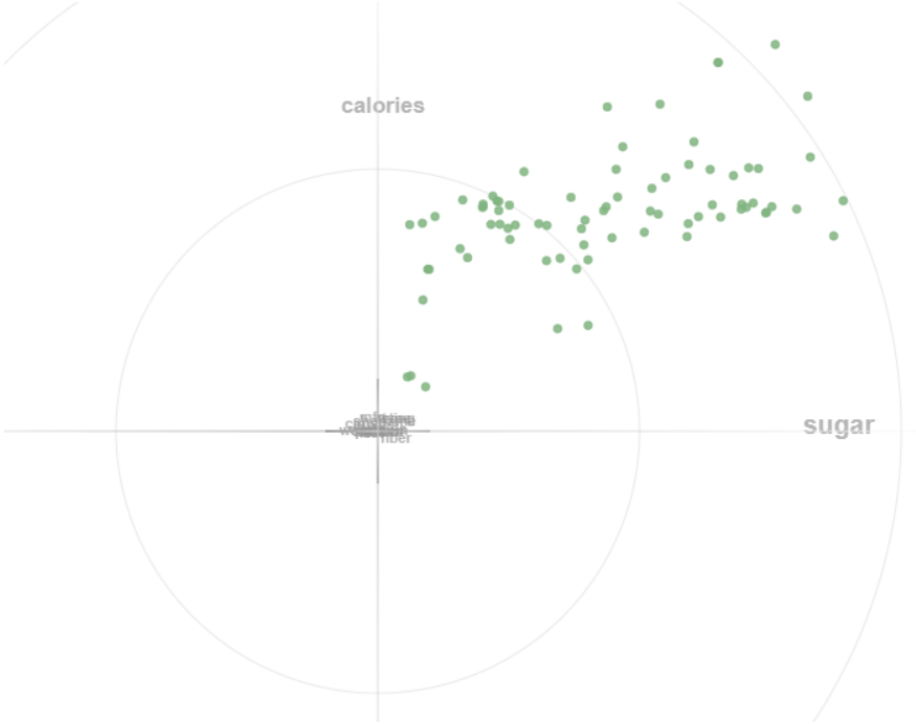


Figure 4.15: Correlation of sugar and calories with a Star Coordinates layout. [Figure taken by the authors of this paper using InterStar developed by Kandogan [2001].]

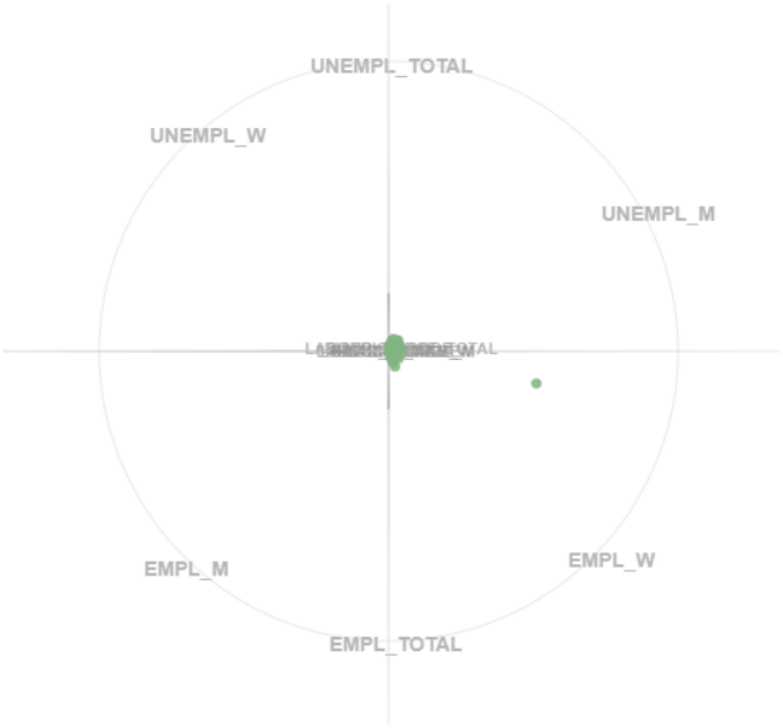


Figure 4.16: Employment and unemployment in Styria with the city Graz as an outlier. [Figure taken by the authors of this paper using InterStar developed by Kandogan [2001].]

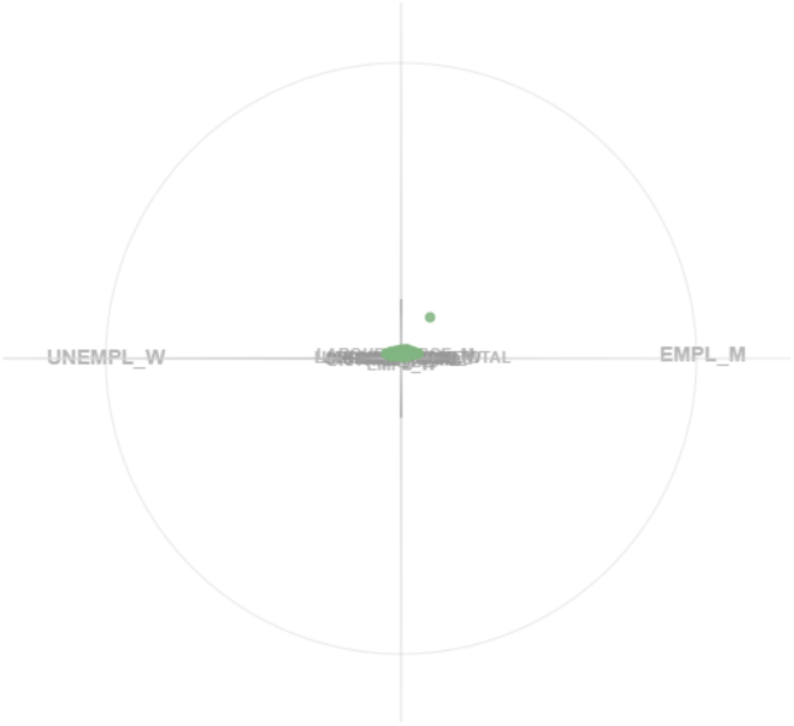


Figure 4.17: Correlation between employed man and unemployed women in Graz. [Figure taken by the authors of this paper using InterStar developed by Kandogan [2001].]

4.5.4 Task 4

In this task we used the dimensions for employed men and unemployed women to find some sort of correlation. We encountered the same problem as in task 3. The datapoints in the center do not really change their position when moving around the two selected dimensions. Therefore we cannot say much about the other cities. The only city which was affected by the movement of the dimensional axis was the city Graz. We tried to find out a correlation of employed men and unemployed women in Graz and moved the two respected dimensions into the opposite sides. The datapoint representing Graz moved really close to the center but still could be detected as an outlier. We can conclude from the final layout seen in 4.17 that compared to other cities, Graz has a slightly higher rate in employed man to unemployed women but overall a really good balance since the datapoint is really close to the center of the graph.

4.5.5 Summary

Task 1 and 2 were easy to solve with Star Coordinates. Being able to move the dimensional axis freely gives the user enough freedom in exploring the data and since the tool suggests the found clusters it makes is really easy and convenient for the user to use and to find these clusters. However, we had problems with the tasks 3 and 4. We used another dataset for these tasks and this dataset had huge differences in values in different datapoints. The population in the city Graz is way higher than every other city in Styria. This results in way more people being employed and also unemployed. The city Graz has the highest values in every dimension and since the axis of the Star Coordinates scales linearly from the minimum value to the maximum, the values of every other city seem to be too insignificant in relation to the high values of Graz. To conclude, Star Coordinates is not suited for every dataset and especially when the dataset contains outliers like data records with huge differences in values to the rest of the datapoints.

4.6 Summary and Comparison

All tools had advantages and disadvantages. In general RadViz and Star Coordinates had an easier time finding correlation in data than Dust and Magnet did. Overall RadViz seemed to handle the different data sets the best and was not influenced by the size of the dataset. Dust and Magnet had an issue with the bigger dataset where the amound of data points meant one needed to zoom out the simulation. Not doing that meant that the huge amount of data moved little to nothing and dust particle overlap was unmanagable. However with the zoomed variant one lost some labels. InterStar was the tool that was most negatively effected by the second dataset. The population distribution of Styria meant that Graz was a siginificant outlier in regard to the size of all values. This lead to the fact that all other cities were always mapped to the center of the coordinate system, as seen in figure 4.16. Due to this one could only ever argue about Graz and not the whole dataset when exploring the data.

Chapter 5

Conclusion

In conclusion there exist many different approaches to radially project data. These provide different advantages and disadvantages when trying to analyze data. One should know about these differences and what one wants to achieve with their data analysis before deciding on which tool to use. In general, concluded from our case study, RadViz seemed the easiest to detect correlations but had some issues with clustering. Star Coordinates was an acceptable all-rounder, whereas Dust and Magnet had an easy time finding clusters but difficulty in finding correlations. From the tools we used for our case study, RadViz-X seemed the most usable one. It had minimal issues with differing data sets and its usage was easily understood. Both InterStar and Dust and Magnet were still usable but had quirks one had to work around. InterStar could not handle the second dataset which had an interesting datarecod outlier with huge values in every dimension. This dwarfed every other record. Dust and Magnet provided many additional features, however it had a big flaw by dust being able to hide behind magnets. This resulted in the only viable analysis being able to be done with a static magnet setup and using manual attraction. An exception was the case with only one magnet in the scene.

Compared to non-radial techniques, radial projections provide a simplified visualization of the data, which is easily read and understood. This makes them an especially useful method early on to analyze data.

Bibliography

Cheng, Shenghui and Klaus Mueller [2015]. *Improving the fidelity of contextual data layouts using a generalized barycentric coordinates framework*. 2015 IEEE Pacific Visualization Symposium (PacificVis) (Hangzhou, China). IEEE, Apr 2015, pages 295–302. doi:10.1109/PACIFICVIS.2015.7156390 (cited on pages 13–15).

Demsar, Janez, Gregor Leban, and Blaz Zupan [2007]. *FreeViz—An intelligent multivariate visualization approach to explorative analysis of biomedical data*. Journal of biomedical informatics 40.6 (Dec 2007), pages 661–71. doi:10.1109/TVCG.2015.2467324 (cited on pages 9–10).

Gabriel, Karl Ruben [1971]. *The biplot graphic display of matrices with application to principal component analysis*. Biometrika 58.3 (Dec 1971), pages 453–467. doi:doi.org/10.1093/biomet/58.3.453 (cited on page 4).

Greenacre, Michael J [2010]. *Biplots in practice*. Fundacion BBVA, 2010. ISBN 9788492384686 (cited on page 4).

Hinum, Klaus, Silvia Miksch, Wolfgang Aigner, Susanne Ohmann, Christian Popow, Margit Pohl, and Markus Rester [2005]. *Gravi++: Interactive Information Visualization to Explore Highly Structured Temporal Data.* J. UCS 11.11 (Nov 2005), pages 1792–1805 (cited on pages 11–12).

Hoffman, Patrick, Georges Grinstein, Kenneth Marx, Ivo Grosse, and Eugene Stanley [1997]. *DNA Visual and Analytic Data Mining*. Proceedings. Visualization'97 (Cat. No. 97CB36155) (Phoenix, AZ, USA). IEEE, Oct 1997, pages 295–302. doi:10.1109/VISUAL.1997.663916. `https://cs.uml.edu/~phoffman/dna1/` (cited on pages 8–9, 27).

Kandogan, Eser [2001]. *Visualizing Multi-Dimensional Clusters, Trends, and Outliers Using Star Coordinates*. Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '01. ACM. San Francisco, California: Association for Computing Machinery, Aug 2001, pages 107–116. ISBN 158113391X. doi:10.1145/502512.502530 (cited on pages 1, 7, 30–33).

Lehmann, D. J. and H. Theisel [2013]. *Orthographic Star Coordinates*. IEEE Transactions on Visualization and Computer Graphics 19.12 (2013), pages 2615–2624 (cited on page 7).

Meyer, Mark, Alan Barr, Haeyoung Lee, and Mathieu Desbrun [2002]. *Generalized barycentric coordinates on irregular polygons*. Journal of graphics tools 7.1 (2002), pages 13–22. doi:10.1080/10867651.2002.10487551 (cited on pages 12–13).

Rubio-Sánchez, M., L. Raya, F. Díaz, and A. Sanchez [2016]. *A comparative study between RadViz and Star Coordinates*. IEEE Transactions on Visualization and Computer Graphics 22.1 (2016), pages 619–628. doi:10.1109/TVCG.2015.2467324 (cited on page 19).

Rubio-Sánchez, M. and A. Sanchez [2014]. *Axis Calibration for Improving Data Attribute Estimation in Star Coordinates Plots*. IEEE Transactions on Visualization and Computer Graphics 20.12 (2014), pages 2013–2022 (cited on page 8).

Rubio-Sánchez, Manuel, Laura Raya, Francisco Diaz, and Alberto Sanchez [2015]. *A comparative study between RadViz and Star Coordinates*. IEEE transactions on visualization and computer graphics 22.1 (Aug 2015), pages 619–628. doi:10.1109/TVCG.2015.2467324 (cited on page 8).

Sangli, Shabana [2014]. *Adopting Star Plot for Visualization of High Dimensional Multivariate Data*. Master's Thesis. Louisiana State University, 2014 (cited on page 3).

Sharko, J., G. Grinstein, and K. A. Marx [2008]. *Vectorized Radviz and Its Application to Multiple Cluster Datasets*. IEEE Transactions on Visualization and Computer Graphics 14.6 (Oct 2008), pages 1444–1427. doi:10.1109/TVCG.2008.173 (cited on page 10).

Soo Yi, Ji, Rachel Melton, John Stasko, and Julie A Jacko [2005]. *Dust & magnet: multivariate information visualization using a magnet metaphor*. Information visualization 4.4 (Dec 2005), pages 239–256. doi:10.1057/palgrave.ivs.9500099 (cited on pages 14, 16–17, 22, 26).

Ying-Huey Fua, M. O. Ward, and E. A. Rundensteiner [1999]. *Hierarchical parallel coordinates for exploration of large datasets*. Proceedings Visualization '99 (Cat. No.99CB37067) (San Francisco, CA, USA). IEEE, Oct 1999, pages 43–508. doi:10.1109/VISUAL.1999.809866 (cited on page 4).