6. Exploratory data analysis I

In the following chapters, we will focus on exploratory analysis of compositional data. In general, exploratory data analysis involves searching for errors and outliers in the data set, looking for patterns, and reporting descriptive statistics. In this chapter, we will focus on the principal component analysis method and the results that can be derived from it. In the following lectures, we will look at some more advanced approaches.

We will consider a data set represented by a matrix $\mathbf{X}$, with n rows (samples) and D columns (parts). The data consists of food consumption in 25 European countries in the early 1980s, broken down into several categories. The values in each category is the percentage of protein provided by the category. The categories do not add up to 100%, since some fraction of the protein consumption is provided by food items that do not fall in any of the reported categories. The data is presented in table 6.1.

Along with the protein sources, there are also two descriptive variables listed in the table, EW and NS. These are not part of the compositions, but rather two categorical variables which, somewhat arbitrarily, describe the location of the country within Europa: eastern or western Europe (E:1, W:2) and nothern or southern Europe (N:1, S:2). This data set is from a time when the economic situation was vastly different between east and west and the goal of this example is to see if this is reflected in the nutritional data.

6.1 Descriptive statistics

For standard real value data, it would be normal procedure to calculate the (arithmetic) mean and the standard deviation (or variance) in order to describe the central trend and the sample dispersion in the data set. For compositional data, these properties however, have no meaning, since they rely on Euclidean geometry, which we have seen in chapter 3, is not appropriate for compositional data. We need to make use of the geometric mean to describe the sample center and the compositional variation matrix to describe the dispersion.

▪ **Example 6.1 — Arithmetic versus geometric mean.** Suppose we buy shares in a company for the price of 100 DKK and keep it for two years before we sell it again. In the first year, the value increase by 90%, while in the second year, the value drops by 90%. What is our mean return for the two years? If we simply use the arithmetic mean, we find that the net return is 0%, suggesting that we can sell the shares for the price that we bought them. This is incorrect. After the first year, our shares have the value $100\text{ DKK} \times 1.9 = 190\text{ DKK}$ and after the second $190\text{ DKK} \times (1-0.9) = 19\text{ DKK}$, that is, we have lost 81 DKK in total when we sell. If we calculate the geometric mean, we get $\sqrt{1.9 \times 0.1} = 0.436 = 43.6\%$. We can check that this is correct: after the first year the average value is $100\text{ DKK} \times 0.436 = 43.6\text{ DKK}$ and after the second year $43.6\text{ DKK} \times 0.436 = 19\text{ DKK}$. ▪

The geometric mean is closely related to the arithmetic mean through logarithms, because the logarithm of the geometric mean of x_i equals the arithmetic mean of the log of x_i.

Definition 6.1.1 — Sample center. The sample center for a set of compositional samples is the closed composition of geometric means of parts. Given n samples with D parts each

$$\text{cen}[\mathbf{X}] = \mathscr{C}\left[\hat{g}_1, \hat{g}_2, ..., \hat{g}_D\right], \tag{6.1}$$

where

$$\hat{g}_j = \left(\prod_{i=1}^{n} x_{i,j}\right)^{1/n}, \qquad j = 1, 2, ..., D \tag{6.2}$$

is the geometric mean of the j'th part.

It is important to note that for the sample center, the geometric mean is considered by parts (column), as opposed to the geometric mean used in CLR-transformation, where it is considered by sample (row).

Definition 6.1.2 — Variation Matrix. The dispersion in the log-ratio of parts is given by the variation matrix

$$\mathbf{T} = [t_{ij}], \qquad t_{ij} = \text{var}\left(\ln \frac{x_i}{x_j}\right), \qquad \text{var}(\mathbf{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2 \tag{6.3}$$

and the total variance

$$\text{totvar}[\mathbf{T}] = \frac{1}{2D}\sum_{i,j=1}^{D} t_{ij} \tag{6.4}$$

The variation matrix is always symmetric around the diagonal and the diagonal elements are always zero. The sample center and variation matrix of the protein consumption data is shown in table 6.2. From the sample center we can see which food items are the biggest contributor of protein in Europe. Cereals are the main contributor with almost $\sim$40% followed by dairy products at $\sim$20%.

The log-ratios with smallest variability are red meat to milk and white meat to eggs. Having small variability means that the parts are almost proportional, which in this case can be explained by producing read meat (cows) also results in milk, while producing white meat (chicken) also results in eggs. The largest contributors to variation are the log-ratios involving fish and nuts, which means that there is little correlation across Europe in food consumption involving these food sources.

Finally, we can use the sample center and the total variance to center and rescale our data set for further analysis. This is done in order to bring all the parts into the same scale. As we saw in Chapter 5, centering and scaling the data preserved the metric properties.

6.2 Principal component analysis

The next step in exploring our data is to do principal component analysis (PCA). Principal components are displayed in a biplot which is also sometimes refered to as a PCA plot. Principal components are the singular value decomposition (SVD) of the data after centering and scaling the variance to 1. SVD algorithms, particularly the ones that are implemented in modern programming languages like R and Python, rely on real Euclidean vector space algebra and are therefore not appropriate for compositional data. However, CLR coefficients preserve the metric properties of the data and obey Euclidean geometry, and they are therefore ideal for PCA on compositional data.

SVD results in two sets of eigenvectors, and a set of eigenvalues. The row eigenvectors form an orthonormal basis for the dataset. These were denoted $\mathbf{\Psi}$ in Chapter 3 and can be used to obtain the ILR-coordinates when multiplied with the CLR-coefficients. In the PCA framework, they are known as *loadings*. The column eigenvectors are known as *scores*. When the PCA is performed on the (centered) CLR coefficient matrix, the diagonal of the score matrix is exactly ILR coordinates in the basis spanned by the loadings. The eigenvalues are proportional to the sample variance of these coordinates.

The biplot is drawn by plotting the two first components of the eigenvectors. Typically, the D loadings are plotted as vectors, often called *rays*, while the n scores are plotted as points called *markers*. The length of the rays is are proportional to the standard deviation of the CLR-coefficients. Longer rays imply greater variance in that part. The interpretation of the PCA is that it reveals the internal structure of the data in a way that best explains the variance of the data. It is a projection of the data matrix onto a two-dimensional space, viewed from the most informative viewpoint.

As mentioned above, SVD is a generalization of the eigendecomposition of a square matrix to a general $m \times n$-matrix. We need this generalization because we do not *a priori* expect to have the same number of samples and parts. The limitation is, that we only get as many eigenvalues as we have columns (samples) and we can therefore not display more parts in the biplot than the number of samples that we have. If we have more parts than samples, we need to get rid of some of the parts before we can do PCA, typically by extracting a sub-composition or by amalgamation. If we have no other way to prioritize the parts, we want to get rid of the parts that contribute the least to the variation matrix.

6.2.1 Scree plots

The amount of variation that is explained in a biplot is a measure of how well the data is represented by the biplot. The fraction of retained variability is the sum of the first two

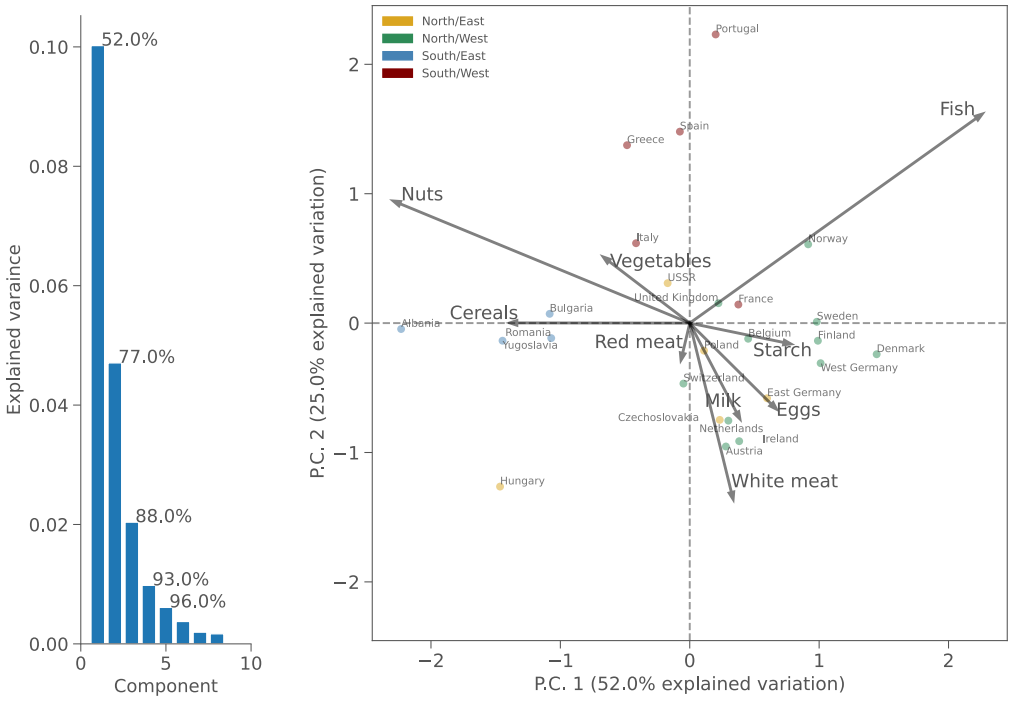

Figure 6.1: Scree plot and biplot of the protein data set.

eigenvalues, divided by the sum of all the eigenvalues. If this ratio is low, then the biplot does not provided a particularly good view of the data set. In order to explore the explained variance per dimension, we can draw a so-called Scree plot. Ideally, we would like to see that the majority of the variance is explained by the first two components. Figure 6.1 shows the Scree plot for the protein data set. We can see that the first two components explain 77% of the total variation in the data set. This means that almost 25% of the variation in the data set is not shown in the biplot.

6.2.2 Interpretation of biplots

The actual biplot is shown next to the Scree plot in Fig. 6.1. PCA biplots are incredible rich and dense in information. The origin of the biplot (0,0) represents the geometric mean of the parts. All loadings and scores are shown relative to the mean (because we centered the data prior to CLR transformation). The length of each ray is proportional to the standard deviation of that part, so long rays means large variance and vice versa for short rays. Moreover, the line segment that connects two rays, called links, are proportional to the standard deviation of the log-ratio of the two parts. The longer the links (i.e, rays are further apart) the larger the variance in the log-ratio of those parts. Consequently, short links corresponds to low variance in the log-ratio, which means that clusters of rays imply that the parts are correlated. In the protein biplot, the fish and the nuts rays are the longest, which also means that all links involving fish and nuts and in particular the link between fish and nuts are long. This matches the result we found when we looked at the variation matrix, that log-ratios involving fish and nuts have the greatest variance. It should be noted, that the length of a ray is actually the link between that part and the origin, which is the

geometric center, so the interpretation of the length of a ray is actually the variance in the log-ratio between the part and the geometric mean. Since the geometric mean will change, in general, when a subcomposition is considered, that log-ratio, and hence the variance in it, will also change. Therefore, drawing conclusions on the length of a single ray is only meaningful when considering the full composition.

Angles between links provide information about correlation of subcompositions. The cosine to the angle between two links is proportional to the correlation coefficient between the log-ratios. A small angle means a correlation coefficient approaching 1, whereas perpendicular links results in a correlation coefficient of zero, that is, completely uncorrelated.

Markers are also plotted in the biplot and they represents the samples in the data set. Markers are typically drawn as points or symbols and they can be colored according to a pre-determined grouping of the samples. That could for instance be the country of sample origin, male/female for human samples, sample year, etc. In cases where there are many samples, it may be useful to plot the group centroid instead (which is possible because the metric properties are preserved under CLR transformation) as well as covariance error ellipses or other indicators of sample dispersion in the group. Cluster analysis can be performed, e.g., k-means, in order to detect samples that are more alike and have more similar compositions. Again, due to the conservation of the metric properties, markers that are closer together are more similar.

Markers can be projected onto the rays. The closer the point along the ray where the projection falls, is to the origin, the closer the part represented by the ray for that particular sample will be to the geometric mean of the set. Marker projections that fall near the end of a ray will have a CLR value for that part which is about one standard deviation larger than the set average. The further beyond the tip of a ray a projection falls, the more extreme is the CLR value for that part in that sample. The opposite is true if the projection falls on the reverse side of the origin.

Likewise, markers can be projected onto links. If the projection of the marker onto a link coincides with the projection of the origin onto that link, then the sample has a log-ratio between those two parts that is equal to the sample set average. If the projection of the marker falls one link length away from the point of the projection of the origin, then that sample has a log-ratio of those two parts that is one standard deviation larger that the set average (or smaller, depending on which part is denominator in the log-ratio).

6.2.3 Subcompositional analysis

We can explore the data further by looking at subcompositions and look for three-part subcompositions with a one-dimensional pattern, that is, constant log-ratios. We are looking for two aligned vertices and one which is perpendicular within the biplot. Once a promising three-part subcomposition has been identified, we re-close it, re-center it, CLR transform and calculate eigenvectors and values.

If, for instance, we pick the three-part subcomposition white meat, cereal, and nuts, we can close it to 100, center it by perturbing by the inverse mean, do a CLR transform and calculate the loadings and eigenvalues using SVD. By doing so, we get,

	Cereals	Nuts	White meat	λ_i
PC.1	-0.554	-3.076	3.630	4.79
PC.2	1.158	-0.723	-0.436	1.43

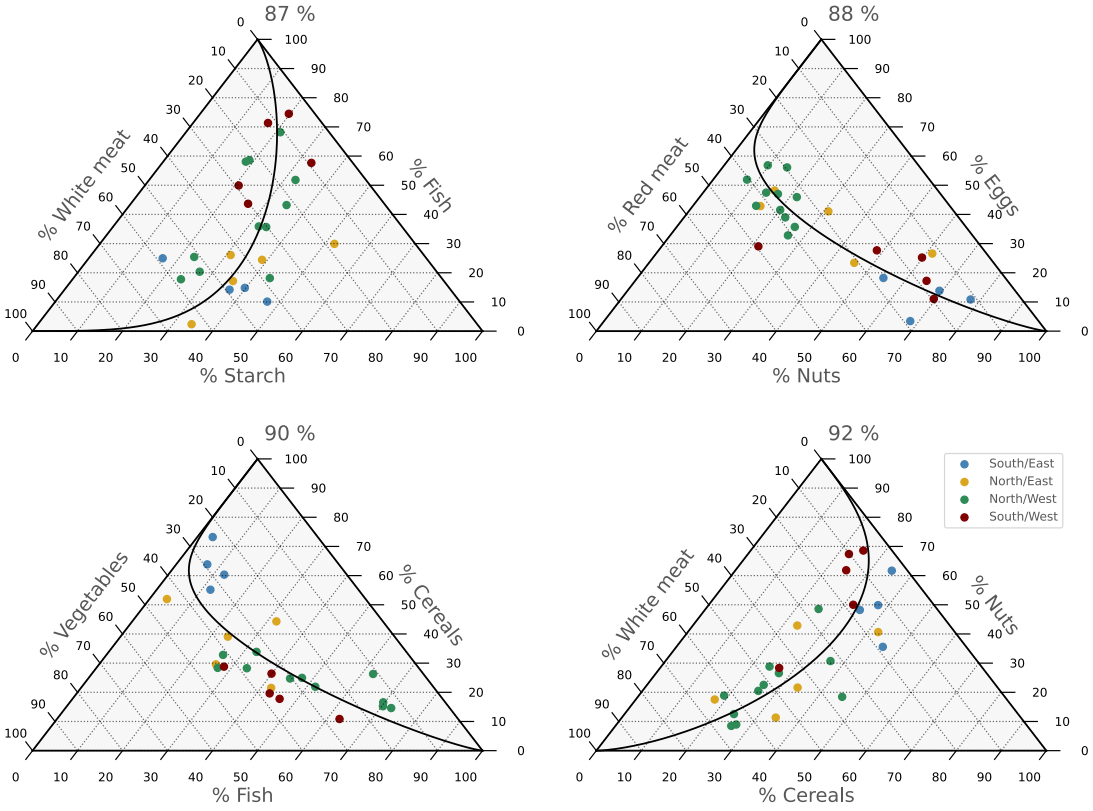

Figure 6.2: Ternary diagrams of various 3-part sub-compositions. The percentage at the top of each diagram is the explained variance along the first principal component.

The explained variance is $\lambda_i^2 / \sum \lambda_i^2 = (0.92, 0.08)$, that is, 92% explained variance along the first principal component. We can proceed to plot the subcomposition in a ternary diagram, along with the compositional line $\mathbf{y} = (\alpha \odot e^{\text{PC.1}}) \oplus \mathbf{g}_m$, where α is a scaling parameter, PC.1 is the first principal component, and $\mathbf{g}_m$ is the geometrical mean of the samples. The result of four selected subcompositions is plotted in Fig. 6.2.

In each of the four cases, around 90% of the variation is explained by the first principal components. This is obvious when looking at the ternary plots, where it is seen that the points are distributed rather tightly along the black lines. The deviation from the black lines is the remaining 10% of the variance and because we have relatively little variation along the second principal component, the *balance* which makes up PC.2, given by,

$$\begin{aligned} PC.2 &= 1.158 \log(\text{Cereals}) - 0.723 \log(\text{Nuts}) - 0.436 \log(\text{White meat}) \\ &= \log \frac{\text{Cereals}^{1.158}}{\text{Nuts}^{0.723} \text{White meat}^{0.436}}, \end{aligned}$$

is close to constant across the samples. This means that cereals to a certain power is highly correlated to the product of nuts to a certain power and white meat to a certain power. The interpretation is, that cereal consumption is balanced by nut and white meat consumption; some countries will consume more nuts, others more white meat, and always more white meat than nuts, but the product will be more or less constant relative to cereals.

Likewise, vegetable consumption is balanced by varying proportions of fish to cereal. In the vegetable-cereals-fish ternary plot, we see that the blue points (south/east) and some of the green points (north/west) falls above the PC.1 line, which means that they all consume a little less vegetables relative to fish and cereals, but the blue countries balance the ratio with more cereals (or less fish) and the green countries with more fish (or less cereals).

This type of analysis can be used to break down high dimensional and complex data sets to look for trends within subcompositions, which may then be pieced together to obtain an understanding of the full data set. Thus we can conclude that the western Mediterranean countries have a high relative consumption of fish, nuts, vegetables, and cereals, but mainly fish. Scandinavian countries are characterized by high relative consumption of starch, eggs, and milk. Eastern block countries are high in nuts and vegetables and low in meat and fish. USSR, France and UK are very close to the European average, probably mainly because of their large population sizes.

6.3 Exercises

For the exercises in this chapter, we will once again make use of the set of three-part compositions which we also used for the Exercises 2.3 in Chapter 2. This time we have added more samples. For convenience, the data can also be found in the file `03_exercises_data.csv`.

	1	2	3	4	5	6	7	8	9	10
x_1	79.07	31.74	18.61	49.51	29.22	21.99	11.74	24.47	5.14	15.54
x_2	12.83	56.69	72.05	15.11	52.36	59.91	65.04	52.53	38.39	57.34
x_3	8.10	11.57	9.34	35.38	18.42	18.10	23.22	23.00	56.47	27.11
	11	12	13	14	15	16	17	18	19	20
x_1	57.17	52.25	77.40	10.54	46.14	16.29	32.27	40.73	49.29	61.49
x_2	3.81	23.73	9.13	20.34	15.97	69.18	36.20	47.41	42.74	7.63
x_3	39.02	24.02	13.47	69.12	37.89	14.53	31.53	11.86	7.97	30.88

Exercise 6.1 Compute the geometric center, the variation matrix, and the total variance of the data set. ■

Exercise 6.2 Perturb the data with the inverse of the geometric center. Compute the center, variation matrix, and total variation of the perturbed data. ■

Exercise 6.3 Make a biplot of the perturbed (centered) data by following these steps:

- Calculate the CLR transform.
- Calculate eigenvectors and eigenvalues of the CLR transformed data[a].
- Plot the first 2 principal components of the 3 loadings in a Cartesian coordinate system as arrows.
- Plot the first 2 principal components of the 20 scores in the same coordinate system as points.

■

[a] Eigenvectors and eigenvalues can be calculated using Singular Value Decomposition (SVD), which is a standard package in most programing languages. In python, SVD can be done using numpy.linalg.svd(). This function returns 3 matrices: the scores (row eigenvectors), eigenvalues, and loadings (column eigenvectors). To bring scores and loadings onto the same scale, scale the loadings by `np.inner(eigvalues*np.identity,loadings.T)`.

Country	Red meat	White meat	Eggs	Milk	Fish	Cereals	Starch	Nuts	Vegetables	EW	NS
Albania	10.1	1.4	0.5	8.9	0.2	42.3	0.6	5.5	1.7	1	2
Austria	8.9	14.0	4.3	19.9	2.1	28.0	3.6	1.3	4.3	2	1
Belgium	13.5	9.3	4.1	17.5	4.5	26.6	5.7	2.1	4.0	2	1
Bulgaria	7.8	6.0	1.6	8.3	1.2	56.7	1.1	3.7	4.2	1	2
Czechoslovakia	9.7	11.4	2.8	12.5	2.0	34.3	5.0	1.1	4.0	1	1
Denmark	10.6	10.8	3.7	25.0	9.9	21.9	4.8	0.7	2.4	2	1
West Germany	8.4	11.6	3.7	11.1	5.4	24.6	6.5	0.8	3.6	2	1
Finland	9.5	4.9	2.7	33.7	5.8	26.3	5.1	1.0	1.4	2	1
France	18.0	9.9	3.3	19.5	5.7	28.1	4.8	2.4	6.5	2	2
Greece	10.2	3.0	2.8	17.6	5.9	41.7	2.2	7.8	6.5	2	2
Hungary	5.3	12.4	2.9	9.7	0.3	40.1	4.0	5.4	4.2	1	1
Ireland	13.9	10.0	4.7	25.8	2.2	24.0	6.2	1.6	2.9	2	1
Italy	9.0	5.1	2.9	13.7	3.4	36.8	2.1	4.3	6.7	2	2
Netherlands	9.5	13.6	3.6	23.4	2.5	22.4	4.2	1.8	3.7	2	1
Norway	9.4	4.7	2.7	23.3	9.7	23.0	4.6	1.6	2.7	2	1
Poland	6.9	10.2	2.7	19.3	3.0	36.1	5.9	2.0	6.6	1	1
Portugal	6.2	3.7	1.1	4.9	14.2	27.0	5.9	4.7	7.9	2	2
Romania	6.2	6.3	1.5	11.1	1.0	49.6	3.1	5.3	2.8	1	2
Spain	7.1	3.4	3.1	8.6	7.0	29.2	5.7	5.9	7.2	2	2
Sweden	9.9	7.8	3.5	24.7	7.5	19.5	3.7	1.4	2.0	2	1
Switzerland	13.1	10.1	3.1	23.8	2.3	25.6	2.8	2.4	4.9	2	1
United Kingdom	17.4	5.7	4.7	20.6	4.3	24.3	4.7	3.4	3.3	2	1
USSR	9.3	4.6	2.1	16.6	3.0	43.6	6.4	3.4	2.9	1	1
East Germany	11.4	12.5	4.1	18.8	3.4	18.6	5.2	1.5	3.8	1	1
Yugoslavia	4.4	5.0	1.2	9.5	0.6	55.9	3.0	5.7	3.2	1	2

Table 6.1: Protein consumption in Europe

Statistics	Red meat	White meat	Eggs	Milk	Fish	Cereals	Starch	Nuts	Vegetables
Sample center	11.93	8.85	3.4	19.94	3.71	39.28	4.89	3.17	4.82
Variation									
Red meat		0.3651	0.2064	0.1354	0.9395	0.3208	0.3851	0.7333	0.3346
White meat			0.1700	0.3198	1.2631	0.5586	0.3001	1.2555	0.4559
Eggs				0.1684	0.8052	0.5679	0.2156	1.1074	0.3936
Milk					0.9611	0.4691	0.3702	1.0476	0.5627
Fish						1.5603	0.6945	2.0827	1.0471
Cereals							0.5937	0.2759	0.2611
Starch								1.1220	0.4313
Nuts									0.4327

Table 6.2: Sample center and off-diagonal compositional covariance matrix.