UNIVERSITY OF IOANNINA
DEPARTMENT OF ECONOMICS
MSc ECONOMIC ANALYSIS

MASTER THESIS

HIERARCHICAL CLUSTER ANALYSIS IN THE EUROPEAN UNION BASED ON THE BEEF WHOLESALE CARCASS PRICE

Author: GEORGIOS BLOUCHOS

Supervisor: ATHANASSIOS STAVRAKOUDIS

January, 2020

Abstract

This study investigates empirically the price situation of the European beef market. Specifically, it utilizes weekly wholesale beef average carcass price data for fifteen European markets and applies the method of hierarchical clustering. In the frame of this application, some of the most fundamental concerns of cluster analysis are presented and discussed thoroughly, while the results of the empirical analysis suggest: First, the DTW distance measure and the choice of the ward linkage method seemed to fit better in the hierarchical agglomerative algorithm concerning our dataset. Second, there is fragmentation and weak connection among the countries of the EU concerning the beef price characteristics. Third, countries playing a major role in the beef market have the highest prices in Europe, while countries less powerful facing lower prices. Fourth, most of the same countries constituting a significant part in the European beef sector face smaller variability concerning their prices. These results relate to the integration and increasing competitiveness which the European Union has set as its goals. However, these are some first indications by simply using hierarchical analysis and further analysis to verify them is necessary.

Acknowledgements

I would like to express my special thanks of gratitude to my supervisor Professor Athanassios Stavrakoudis for his continuous support and cooperation for the fulfillment of this thesis. I am thankful for his guidance, invaluably constructive criticism and his persistence in familiarizing myself with the R statistical program through these years. Also, I would like to thank Professor Dimitrios Panagiotou for his helpful advice on improving this thesis.

Contents

List of Figures **6**

List of Tables **9**

1 The beef market in the European Union **10**
1.1 Introduction . . . 10
1.2 Meat-The product . . . 11
1.3 Beef Cattle . . . 12
1.4 Production . . . 13
1.4.1 World Market . . . 13
1.4.2 Europian Union market . . . 15
1.5 Consumption . . . 18
1.6 Trade . . . 18
1.7 Common Agriculture Policy (CAP) . . . 19

2 Literature Review **21**

3 Data and Methodology **23**
3.1 Data . . . 23
3.2 Methodology . . . 25
3.2.1 Clustering . . . 25
3.2.2 Distance Measures . . . 27
3.2.3 Hierarchical Clustering . . . 30
3.2.4 Tools for time series clustering . . . 32

4 Empirical Analysis and Discussion **34**
4.1 Descriptive Statistics . . . 34
4.1.1 Raw data . . . 35
4.1.2 Log-Returns . . . 40
4.2 Empirical analysis . . . 44
4.2.1 Raw data scenario . . . 46
4.2.2 Scaled data scenario . . . 58

4.2.3 Log-returns data scenario 70

5 Conclusion **82**

Bibliography **85**

A Additional descriptive and empirical results **A88**

B R coding **B107**
B.1 Data downloading and cleaning B107
B.2 Descriptive statistics B109
B.3 Empirical analysis (raw data) B112
B.4 Empirical analysis (scaled data) B116
B.5 Empirical analysis (log-returns) B119
B.6 Appendix figures B123

List of Figures

1.1 World production of meat (2018) 14
1.2 World production of bovine meat (2018) 14
1.3 Pie chart of bovine meat world production (2018) 15
1.4 Beef production in the EU-28 1000(t), 2018 17

4.1 Raw prices over time (weeks) for the fifteen EU beef markets. 35
4.2 Boxplots of the raw prices 37
4.3 Histograms of the raw prices 38
4.4 Density plots of the raw prices 39
4.5 Correlation heatmap (pearson) of the raw prices 40
4.6 Log-return (prices) over time (weeks) for the fifteen EU beef markets. 41
4.7 Histograms of the log-return prices 42
4.8 Density plots of the log-return prices. 43
4.9 Boxplot of the log-return prices 44
4.10 Heatmap based on the euclidean dissimilarity matrix, raw data 46
4.11 Dendrogram of the raw dataset, euclidean distance and ward linkage . 47
4.12 Dendrogram of the raw dataset, euclidean distance and complete linkage 47
4.13 Dendrogram of the raw dataset, euclidean distance and average linkage 48
4.14 Dendrogram of the raw dataset, euclidean distance and single linkage 48
4.15 Heatmap based on the DTW dissimilarity matrix, raw data 49
4.16 Dendrogram of the raw dataset, DTW distance and ward linkage . . . 50
4.17 Dendrogram of the raw dataset, DTW distance and complete linkage 50
4.18 Dendrogram of the raw dataset, DTW distance and average linkage . 51
4.19 Dendrogram of the raw dataset, DTW distance and single linkage . . 51
4.20 Elbow method results based on the raw dataset 52
4.21 Silhouette method results based on the raw dataset 53
4.22 Gap stat method results based on the raw dataset 53
4.23 Side by side comparison between hierarchical clustering with the ward's linkage and the euclidean distance, raw data (left) versus hierarchical clustering with the ward's linkage and the DTW distance, raw data (right) 54

4.24 Final hierarchical clustering for $k = 3$ based on DTW distance and ward linkage method, raw data 55
4.25 Final hierarchical clustering for $k = 4$ based on DTW distance and ward linkage method, raw data 56
4.26 Line graphs grouped in their own clusters, $k = 3$ 57
4.27 Line graphs grouped in their own clusters, $k = 4$ 57
4.28 Heatmap based on the euclidean dissimilarity matrix, scaled data . . 59
4.29 Dendrogram of the scaled dataset, euclidean distance and ward linkage 59
4.30 Dendrogram of the scaled dataset, euclidean distance and complete linkage 60
4.31 Dendrogram of the scaled dataset, euclidean distance and average linkage 60
4.32 Dendrogram of the scaled dataset, euclidean distance and single linkage 61
4.33 Heatmap based on the DTW dissimilarity matrix, scaled data 62
4.34 Dendrogram of the scaled dataset, DTW distance and ward linkage . 62
4.35 Dendrogram of the scaled dataset, DTW distance and complete linkage 63
4.36 Dendrogram of the scaled dataset, DTW distance and average linkage 63
4.37 Dendrogram of the scaled dataset, DTW distance and single linkage . 64
4.38 Elbow method results based on the scaled dataset 65
4.39 Silhouette method results based on the scaled dataset 66
4.40 Gap stat method results based on the scaled dataset 66
4.41 Side by side comparison between hierarchical clustering with the ward's linkage and the euclidean distance, scaled data (left) versus hierarchical clustering with the ward's linkage and the DTW distance, scaled data (right) 67
4.42 Side by side comparison between hierarchical clustering with the ward's linkage and the DTW distance, raw data (left) versus hierarchical clustering with the ward's linkage and the DTW distance, scaled data (right) 67
4.43 Final hierarchical clustering for $k = 2$ based on DTW distance and ward linkage method, scaled data 68
4.44 Final hierarchical clustering for $k = 4$ based on DTW distance and ward linkage method, scaled data 69
4.45 Heatmap based on the euclidean dissimilarity matrix, log-returns . . . 70
4.46 Dendrogram of the log-returns, euclidean distance and ward linkage . 71
4.47 Dendrogram of the log-returns, euclidean distance and complete linkage 71
4.48 Dendrogram of the log-returns, euclidean distance and average linkage 72
4.49 Dendrogram of the log-returns, euclidean distance and single linkage . 72
4.50 Heatmap based on the DTW dissimilarity matrix, log-returns 73
4.51 Dendrogram of the log-returns, DTW distance and ward linkage . . . 74
4.52 Dendrogram of the log-returns, DTW distance and complete linkage . 74
4.53 Dendrogram of the log-returns, DTW distance and average linkage . . 75
4.54 Dendrogram of the log-returns, DTW distance and single linkage . . . 75
4.55 Elbow method results based on the log-returns 76
4.56 Silhouette method results based on the log-returns 77
4.57 Gap stat method results based on the log-returns 77

4.58 Side by side comparison between hierarchical clustering with the ward's linkage and the euclidean distance, log-returns (left) versus hierarchical clustering with the ward's linkage and the DTW distance, log-returns (right) 78
4.59 Final hierarchical clustering for $k = 2$ based on DTW distance and ward linkage method, log-returns 79
4.60 Final hierarchical clustering for $k = 4$ based on DTW distance and ward linkage method, log-returns 80

A2 Bovine livestock in the EU-28, 2018 . . . . . A90
A3 Q-Q plots of the raw prices A91
A4 Q-Q plots of the log-return prices A92
A5 Correlation heatmap (Spearman) of the raw prices A93
A6 Correlation heatmap (Kendall) of the raw prices A94
A7 Final hierarchical clustering for $k = 3$ based on euclidean distance and ward linkage method, raw data A97
A8 Final hierarchical clustering for $k = 4$ based on euclidean distance and ward linkage method, raw data A98
A9 Final hierarchical clustering for $k = 2$ based on euclidean distance and ward linkage method, scaled data A101
A10 Final hierarchical clustering for $k = 4$ based on euclidean distance and ward linkage method, scaled data A102
A11 Final hierarchical clustering for $k = 2$ based on euclidean distance and ward linkage method, log-returns A105
A12 Final hierarchical clustering for $k = 4$ based on euclidean distance and ward linkage method, log-returns A106

List of Tables

1.1 Production of beef 1000(t) in the EU-28 16

4.1 Descriptive Statistics for the raw prices 36
4.2 Descriptive Statistics for the log-return prices 42
4.3 Agglomerative coefficient results for the four linkage criteria (raw data, euclidean distance) 49
4.4 Agglomerative coefficient results for the four linkage criteria (raw data, DTW distance) 52
4.5 Agglomerative coefficient results for the four linkage criteria (scaled data, euclidean distance) 61
4.6 Agglomerative coefficient results for the four linkage criteria (scaled data, DTW distance) 64
4.7 Agglomerative coefficient results for the four linkage criteria (log-returns, euclidean distance) 73
4.8 Agglomerative coefficient results for the four linkage criteria (log-returns, DTW distance) 76

A2 Production of meat: cattle 1000(t) in the EU-28 . . . . . A88
A3 Bovine livestock in the EU-28 (1.000 heads) A89
A4 Dissimilarity matrix based on the euclidean distance, raw dataset . . A95
A5 Dissimilarity matrix based on the DTW distance, raw dataset A96
A6 Dissimilarity matrix based on the euclidean distance, scaled dataset . A99
A7 Dissimilarity matrix based on the DTW distance, scaled dataset . . . A100
A8 Dissimilarity matrix based on the euclidean distance, log-returns . . . A103
A9 Dissimilarity matrix based on the DTW distance, log-returns A104

1 The beef market in the European Union

1.1 Introduction

Although the economic significance of agriculture within the European Union economy had been on a steady decline for the last 50 years, it remains an essential sector and has grown rapidly as a concept in the last years, playing a strategic role in the process of economic development. Nonetheless, agriculture products' importance goes far beyond their simple economic function since they contribute to Europe's regional, cultural and gastronomic identity.

If we go back over some decades, agriculture was connected mostly with the production of basic crops, but nowadays includes other important factors such as forestry, fruit cultivation, poultry and dairy farming. The meat sector is one of the most important in the European Union agriculture and encompasses four main meat types; beef and veal, pig meat, poultry meat and sheep meat/goat meat. Half of all EU farms have livestock and a great number of farmers with ruminant animals are specialist livestock producers.

Meat is a major source of protein and constitutes an important part of the European diet. EU policies in the meat sector are designed to encourage the production of safe, nutritious and affordable meats. Policies, as we shall see, are geared increasingly towards meeting the needs of consumers, livestock producers and the environment in a balanced way.

Following a deductive method of organization, the rest of this section is developed from a general-to-specific order. We start with meat as a commodity and end up in the cattle commodity, studying the rank of meat and bovine market in the world and conclude to the beef market within the European Union, which is our main interest. This will help us understand the European Union's ranking in the world in terms of the main economic sectors and what is happening in the internal.

1.2 Meat-The product

Meat is an important meal and commodity in many parts of the world which constitutes a significant share of a typical diet and has long formed an important part of the European diet.

Meat contains a wide variety of nutrients, including high-value proteins, complex B vitamins, and especially vitamin B12, vitamin D, and iron, zinc, phosphorus and selenium. Meat protein is of high biological value as it contains a complete and well-balanced variety of amino acids, which are the cornerstone for growth and development of the body. Meat is a rich source of iron and although iron is found in all types of white and red meat (beef, pork, chicken, turkey, fish), veal has the highest iron content and is essential for the production of healthy blood, in which oxygen is transported bound to a protein. As with iron, zinc is more readily absorbed from meat than from vegetable foods, thus making meat a necessary source for the intake of this trace element. Zinc is essential for growth and reproduction, as well as for the defense of the body, but also wound healing. Another mineral present in meat is phosphorus. Phosphorus contributes to good dental and bone health, is involved in the burning of sugar for energy production, is an important component of DNA and RNA, plays a role in hormonal and enzymatic regulation, is a component of cell walls and contributes to its maintenance Blood pH. Meat is particularly rich in B vitamins (B1, B3, B6), but mainly in vitamin B12. Vitamin B12 is needed to structure our genetic material, DNA, and thus has many functions in the body, some of which are the production of healthy blood and its contribution to the proper functioning of the nervous system (1),(25).

The so-called 'red meats' (beef/veal and sheepmeat/goatmeat) and 'white meats' (pigmeat and poultrymeat) offer a variety of positive properties and a choice of tastes and textures. Also, meat is a very versatile culinary product and has become a vital element of European cuisine and culture.

Due to a diversity of species, traditions of livestock production and terrain, the EU has a wide variety of livestock types and meat products derived from them and increases the choices for the consumers. Meat products are major beneficiaries of the EU's quality mark schemes and great efforts have been made by the meat production chain to improve the quality of products as well as their labeling and marketing.

The most common sources of meat as mentioned above are domesticated animal species such as cattle, pigs and poultry and to a lesser extent buffalo, sheep and goats. For thousands of years, poultry supplied meat and eggs, cattle, sheep and goats provided meat and milk, and pigs provided a source of meat. These species are the main sources of animal protein for humans.

The meat derived from cattle is known as beef and this is the main market that we will be dealing with throughout this study.

1.3 Beef Cattle

Cattle are domesticated bovine farm animals that are raised as livestock for meat (beef or veal) and for their milk or hides, which are used to make leather. The animals most often included under the term are the Western or European domesticated cattle as well as the Indian and African domesticated cattle. In the terminology used to describe the sex and age of cattle, the male is first a bull calf and if left intact becomes a bull; if castrated he becomes a steer and in about two or three years grows to an ox. The female is first a heifer calf, growing into a heifer and becoming a cow. Males retained for beef production are usually castrated to make them more docile on the range or in feedlots; with males intended for use as working oxen or bullocks, castration is practiced to make them more tractable at work [1].

Cattle farming is usually dependent on the areas offered for grazing but is often found in areas where farming uses new methods, mainly for their high working capacity and their natural fertilization. Breeding occurs extensively in sparsely populated areas and where the climate is dry and intense on fertile, humid and often densely populated areas, such as in the Netherlands, Denmark, etc., where agriculture gives its place to cattle breeders.

The number of bovine animals in the world is constantly increasing and this is due not only to the ever-increasing demand for meat but also to the development of technology, which contributes to the facilitation of transport and the growth of refrigeration facilities which allow for more efficient exchange of products. The use of cattle as commodities has been a point of philosophical controversy, particularly regarding the raising of animals for food. Such issues are compounded by modern concerns about the ethics of industrial factory farming and the contribution of commercial meat production to global warming.

Beef is the meat from the slaughter of mature cattle of at least one year old, while veal is distinguished as meat from bovine animals younger than one year old, which is the flesh of calves.

In Europe, there are mainly two farming methods, the pasture-based or "extensive" production systems and the cereal-based systems. The first takes place generally in the pastoral regions of Europe, where cereal cultivation is more difficult and in the mountainous areas throughout Europe. According to this method, the animals grow at a slower rate, often reaching higher weights and their meat is more mature with a greater taste. The latter is more common in southern and central Europe, where the climate is hotter and there are plenty of cereals respectively. In this method on the contrary, beef animals grow much faster and reach their slaughter weight quicker [2].

Animal diseases are a constant threat to livestock. In particular, this issue disordered the European Union's beef markets in the early 21st century, following

[1] `https://www.britannica.com/topic/agriculture`

[2] The meat sector in the Europian Union - `https://ec.europa.eu/agriculture/publi/fact/meat/2004_en.pdf`

the crises caused by the bovine spongiform encephalopathy (BSE) and foot-and-mouth disease, a fact that had a wider impact on beef consumption. However, the bouts had been successfully managed and led Europe to step up with supportive policies and actions for agriculture and mechanisms to control such cases, providing support to the beef producers.

1.4 Production

1.4.1 World Market

Meat production plays a significant part in the world's economy and contributes to the local, nation and international trade. In recent decades, among many reasons, but mainly because of the growing population, rising urbanization and technological progress, there is an increased demand for livestock products, particularly in developing countries. Consumers there also gain more purchasing power, a fact that increased the demand for meat substantially. Purchasing power is also directly related to consumers' preferences, who require more options in the meat marketplace. As a result, meat production faces many challenges which may lead to multiple paths in the future.

The world meat production for 2018 containing bovine, pig, poultry and ovine meat was estimated to 336.4 million tones, while the corresponding value for 2017 was 332.4. This means that there was an increase of 1.2 %, the fastest growth since 2014. The main countries that are leading the meat production and contributed to this increase are the USA, EU and Russia. India, Mexico and Argentina also play an important role and increased their product to a certain extent. On the contrary, a decrease occurred in Brazil and China, who also have a leading role, with the latter producing the largest meat output worldwide. Figure (1.1) shows the share of each country in world production. As we see, the EU ranks behind China, constituting a major meat producer in global terms accounting for over 14 % of world meat production. Concerning now the various meat categories, bovine stands in second place of the Global meat output. In 2018, the bovine world total meat production was 8.03 million tonnes and recorded the highest rise (+2.1), while poultry meat followed with an increase of (+1.3). Ovine and pigmeat, with the latter holding the largest share, remained almost stable. The USA is the biggest player in the bovine market, while EU28 also owns a remarkable share and it is third in the world ranking [3] (see Figure 1.2).

[3]The numbers and figures (1.1, 1.2) can be found in the Food and Agriculture Organisation of the United Nations (FAOSTAT) and more specifically in the MEAT MARKET REVIEW: Overview of global meat market developments in 2018 - `http://www.fao.org/3/ca3880en/ca3880en.pdf`

Figure 1.1: World production of meat (2018)

Total meat production (thousand tonnes, CWE)			
	2017	2018	Change 2018 over 2017 (%)
World	**332 464**	**336 369**	**1.2**
China	86 887	86 598	-0.3
EU 28	48 163	49 084	1.9
United States	45 772	46 768	2.2
Brazil	27 586	27 579	0.0
Russian Fed.	9 900	10 248	3.5
India	7 256	7 424	2.3
Mexico	6 822	7 028	3.0
Argentina	5 756	5 953	3.4

Source: FAO

Figure 1.2: World production of bovine meat (2018)

Bovine production (thousand tonnes, CWE)			
	2017	2018	Change 2018 over 2017 (%)
World	**69 614**	**71 083**	**2.1**
United States	11 943	12 254	2.6
Brazil	9 550	9 932	4.0
EU 28	7 867	8 032	2.1
China	6 361	6 457	1.5
Argentina	2 842	3 049	7.3
India	2 524	2 536	0.5
Australia	2 149	2 306	7.3
Mexico	1 927	1 979	2.7

Source: FAO

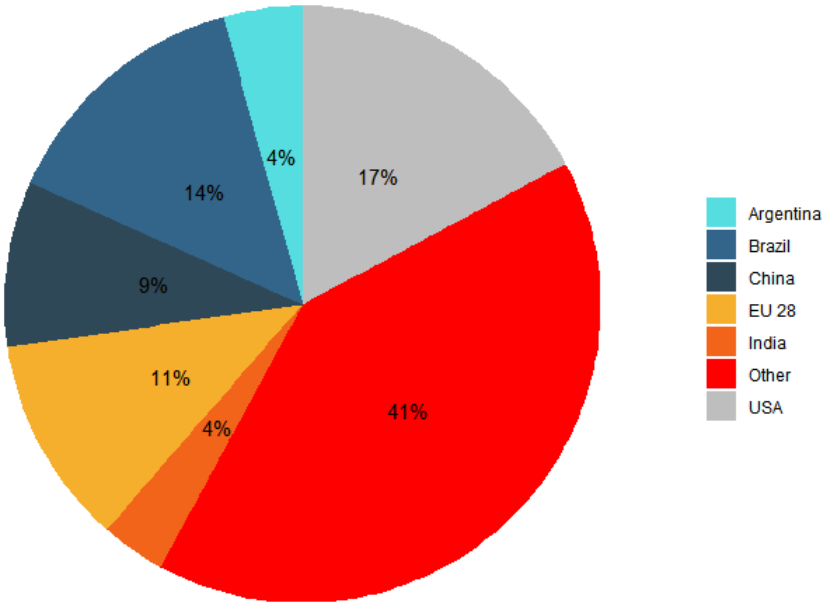


Figure 1.3: Pie chart of bovine meat world production (2018)

Source: FAO(calculated from figure 1.2)

1.4.2 Europian Union market

From the foregoing mentioned, it becomes apparent that the European Union, thanks to its temperate climate and the heterogeneity of its territories, presents a variety of areas for agricultural production. Meat production represents, on average, 40% of the final agricultural production and consists mainly of the four types (pigs, bovine animals, sheep, goats). The main meat output in the EU-28 is pork (23.8 million tonnes in 2018), which produced almost three times the weight of beef/veal (7.9 million tonnes in 2018).

The beef sector is characterised by a diversity of production systems and breeds. As well as specialized livestock farming, there are combined forms of farming which include extensive farming, indoor fattening, specialized meat and veal production, etc., with long production cycles, low income and high production costs. EU produced 7.9 million tonnes of bovine meat (beef and veal), a slightly bigger quantity from 2016 and 2017, when it remained the same (7.8 million tonnes). Almost half of the total beef production in the EU came from France (18.3 %), Germany (15.2 %) and the United Kingdom (13.2 %). The next important countries in beef production were Italy (10.3 %) and Ireland (9 %), while Poland's 8.1 % and Spain's 6 % were also noteworthy [4].

[4]Calculated from table 1.1

Table 1.1: Production of beef 1000(t) in the EU-28

Countries	**2011**	**2012**	**2013**	**2014**	**2015**	**2016**	**2017**	**2018**
EU-28	6.861,18	6.548,32	6.271,03	6.353,92	6.596,19	6.776,22	6.775,78	6.911,36
Belgium	219,40	209,66	195,55	202,16	208,93	215,62	216,81	216,16
Bulgaria	4,11	4,63	4,91	4,28	4,66	5,90	6,69	6,65
Czechia	71,33	64,91	64,12	64,78	67,55	71,21	67,04	70,86
Denmark	102,20	96,90	97,40	99,40	93,10	100,30	93,00	96,60
Germany	1.105,00	1.080,00	1.050,00	1.073,00	1.071,00	1.092,00	1.068,00	1.053,00
Estonia	8,15	7,75	7,70	8,64	9,21	9,05	8,63	8,24
Ireland	545,44	494,60	516,69	580,65	563,26	587,38	615,36	620,47
Greece	47,56	46,99	42,34	37,24	34,14	31,86	34,73	30,61
Spain	359,38	342,10	338,65	340,97	388,00	388,64	386,41	415,52
France	1.340,32	1.266,44	1.201,97	1.219,30	1.251,57	1.262,17	1.244,60	1.267,45
Croatia	47,50	40,90	41,50	39,30	37,50	39,80	37,80	39,41
Italy	880,43	853,81	745,15	607,74	684,25	700,63	651,66	715,93
Cyprus	3,92	4,53	3,71	3,91	4,01	3,88	4,23	4,45
Latvia	16,09	15,32	14,61	15,92	16,31	16,58	15,72	14,96
Lithuania	40,61	39,50	36,32	38,84	43,64	41,75	40,31	39,75
Luxembourg	8,68	8,29	7,78	8,32	8,91	9,25	9,39	9,70
Hungary	25,55	24,11	22,05	22,52	25,75	27,45	26,57	28,45
Malta	1,11	1,11	1,13	1,13	1,03	1,13	1,09	1,04
Netherlands	162,84	158,86	156,68	158,97	157,50	177,73	203,16	197,32
Austria	213,38	213,89	220,12	214,77	222,31	221,37	220,36	227,82
Poland	370,41	362,90	332,91	409,08	467,47	497,72	555,25	561,74
Portugal	73,04	68,70	62,60	59,92	68,09	68,90	68,97	72,01
Romania	22,64	21,92	22,15	22,74	34,94	45,91	46,17	37,66
Slovenia	33,26	31,00	30,28	29,79	31,94	33,84	33,79	32,96
Slovakia	11,19	9,67	9,42	8,73	8,30	8,16	7,68	8,02
Finland	82,28	80,05	80,12	81,95	85,37	85,64	85,07	86,15
Sweden	133,45	120,82	121,66	127,55	129,65	128,59	129,67	134,34
United Kingdom	931,90	878,97	843,52	872,34	877,81	903,77	897,64	914,14

Source: Eurostat (online data code: apro_mt_pann)

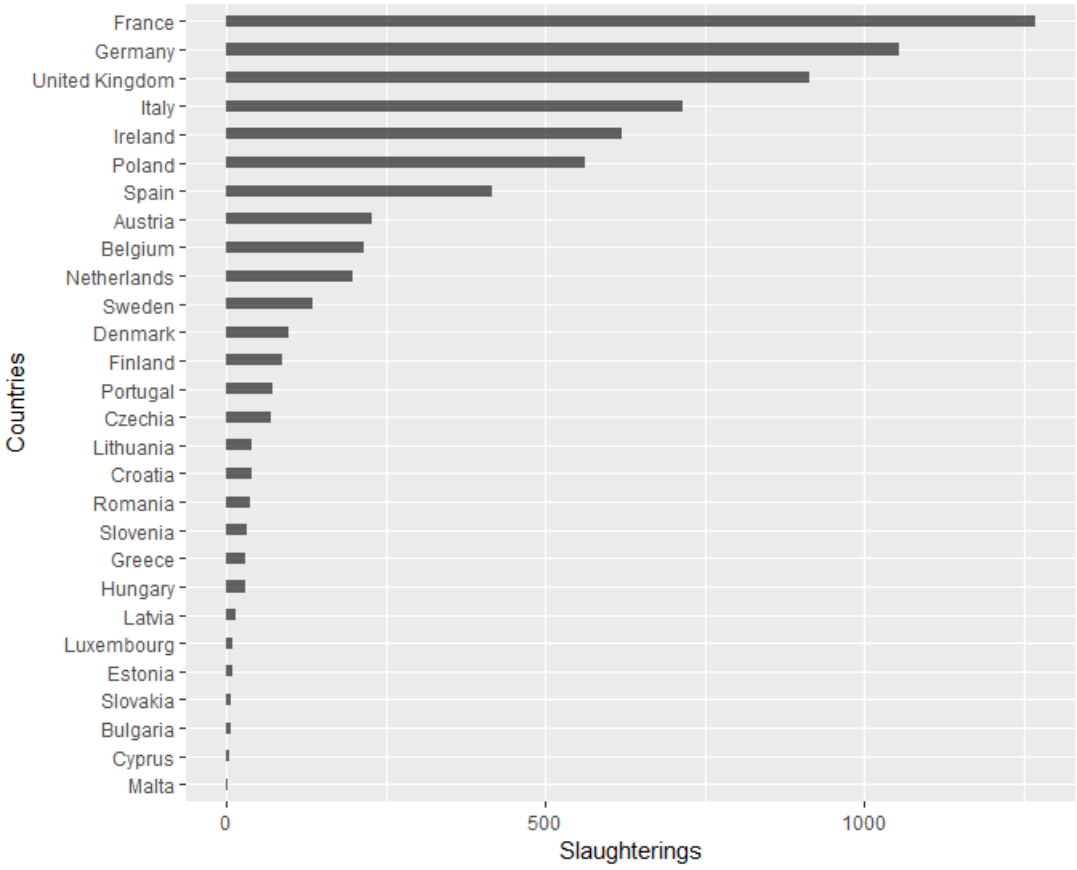


Figure 1.4: Beef production in the EU-28 1000(t), 2018

Source: Eurostat (calculations based on the online data code: apro_mt_pann)

Concerning now the three main producers, Germany is a dairy country with a high share of dairy cows among the total number of cows. More than forty different cattle breeds reflect the regional, climatic and gastronomic differences between the Bavarian Alps in the South and the North and Baltic Seas. Fleckvieh and Braunvieh are the main breeds in the south of Germany, while in the north, German Holsteins predominate. The bovine production system in southern Germany is more extensive than in the north, where production is more concentrated and integrated. The French beef sector is made up of diverse production systems including both dairy herd and suckler herd, and both specialized and mixed breeding systems. The cattle sector in the United Kingdom is typical for its large number of producers; many are either Micro or Small to Medium Enterprises (SME's) and, while the concentration in the sector is changing due to modifications in the subsidy structure, consolidation is still limited. The dairy sector is an important supplier of animals for beef production, but there has been a steady decline in the number of United Kingdom holdings with dairy cows [5].

[5]Evaluation of EU beef labelling rules - `https://ec.europa.eu/agriculture/sites/agriculture/files/evaluation/market-and-income-reports/2015/eu-beef-labelling-rules/fullrep_en.pdf`

1.5 Consumption

In Europe, meat consumption is inseparably linked to economic, environmental and social issues. More specifically, purchasing power and price levels are classic among the key areas that consumers take into account, with strong expectations on issues such as food quality, health value of meat, animal welfare and other traditional features. Various social concerns have been highlighting recent years and seem to be more and more important over time, while environmental matters such as environmentally friendly meat production have a significant role and will continue to do so in the near future (10). Total per capita meat consumption in the EU has barely changed since 2000. What has evolved is the proportion of the basic meats consumed in the Union. Although bovine demand was doing well in the 2000s and having recovered from the crises, there was a decline in consumption stemming from a decline in supply due to the economic crisis. In the latest years, EU meat consumption dominated by pigmeat and beef and veal consumption comes in the third place, having lost a market share from poultry meat, whose demand has almost doubled concerning cattle. As a result, the consumption of beef has declined to about 10% in these two decades (10).

1.6 Trade

International trade for beef has grown remarkably in recent years. The main exporters are Brazil; which leads the ranking of the largest exporter of beef in the world since 2008, the United States, Australia, India and New Zealand [6]. The increase in beef consumption in various Asian countries helps these markets, with China being the largest importer in the world for 2018, mainly due to the increasing demand of its consumers. The rapid growth in Chinese beef imports has dramatically altered global beef flows with several countries now exporting a significant share of total exports to China. It's increasing demand for bovine meat is expected to continue for the next years and as a result, the trade flows are expected to continue to increase. The European Union, although it ranks third in terms of world meat exports behind the United States and Brazil, plays no significant role in the specific sector concerning beef.

However, there is also the sector of beef trade within the European Union, called intra-EU trade, which is the sum of all the quantities sold by all the Member States to the other Member States. More specifically, intra-EU refers to all transactions occurring within the EU and the term is used in the context of external trade, the balance of payments and other similar statistical areas. On the contrary, extra-EU refers to transactions with all countries outside of the EU, which is the rest of the world except for the EU Member States [7]. Internal trade development is a significant subject toward the Europian Union. Volumes traded across borders within the EU market decreased sharply in 1996 and 2000, two years marked by the BSE crisis.

[6] MEAT MARKET REVIEW: Overview of global meat market developments in 2018 - `http://www.fao.org/3/ca3880en/ca3880en.pdf`

[7] https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Glossary:Extra-EU

Total intra-EU trade increased between and after the end of the crisis and continued in the following years with a gradual increase also due to the entry of brand-new members into the Union and the enlargement of the market for internal trade [8]. The extra-EU trade in beef is much lower than the internal trade between the Member States in the most recent years, mainly due to the also reduced production and the significant rise of other world markets over the same period. The European countries playing a major role in the consumption and production of beef within the European Union as mentioned above, also hold a leading part in the internal trade market. Germany, Italy and the United Kingdom are large importers of beef while France and Spain also receive a remarkable share. Ireland stands out with a good percentage in this economic index, as it possesses a special relationship with Great Britain, with a large number of exchanges being taken place from one country to another [9].

1.7 Common Agriculture Policy (CAP)

As stated earlier, the EU tries to set conditions for farmers to fulfill multiple functions, including the principal aim of producing high-quality, safe food. Launched in 1962, the Common Agricultural Policy (CAP) is its unified agricultural policy, a partnership between agriculture and society and between Europe and its farmers [10]. It describes a set of laws and regulations related to agriculture, livestock breeding, movement of agricultural products and all resulting outcomes, such as price stability, product quality, product selection, land use and employment in the agricultural sector. Its main aims are to improve agricultural productivity so that consumers have a stable supply of affordable food and to ensure that EU farmers can make a reasonable living. The CAP is a common policy for all the Member States of the European Union and it is managed and funded at European level from the resources of the EU's budget.

Significant reforms of the CAP have taken place in recent years, most notably in 2003 (the CAP provides income support), 2008 and 2013 (the CAP is reformed to strengthen the competitiveness of the sector, promote sustainable farming and innovation, support jobs and growth in rural areas and move financial assistance towards the productive use of land), which have sought to make EU's agricultural sector more market-oriented and ensure that safe and affordable food continues to be produced while respecting environmental and sustainability concerns [11].

CAP policies relating to meat have also evolved over many years and are in-

[8] Evaluation of EU beef labelling rules - https://ec.europa.eu/agriculture/sites/agriculture/files/evaluation/market-and-income-reports/2015/eu-beef-labelling-rules/fullrep_en.pdf

[9] Evaluation of EU beef labelling rules - https://ec.europa.eu/agriculture/sites/agriculture/files/evaluation/market-and-income-reports/2015/eu-beef-labelling-rules/fullrep_en.pdf

[10] https://ec.europa.eu/info/food-farming-fisheries/key-policies/common-agricultural-policy/cap-glance_en

[11] https://ec.europa.eu/info/food-farming-fisheries/key-policies/common-agricultural-policy/cap-glance_en

creasingly focused specifically on improving the quality of the product, on giving farmers confidence in their future income prospects and on encouraging more environmentally-sustainable farming practices. Reforms specifically in the beef and veal market aim to contribute to market stability, strengthen the competitiveness of the sector, stabilize farm incomes and provide further incentives for producers to move towards more extensive methods of production.

In the past, due to its importance, CAP had occasionally been the subject of much controversy and has received pressure from various sources that led to its major changes-reforms during the length of time. Therefore it has evolved over the years to meet these citizens' requirements and expectations as well as the changing economic circumstances and will continue to evolve, securing benefits to all EU citizens.

2

Literature Review

In this chapter, we present briefly some literature review based on two points. The first is the question of the European integration of the countries, which is of great importance since the results of our empirical analysis are part of this. In the following, we are more specific about the beef sector and what happens mainly to the production hierarchy of the countries and the price index.

The integration of the EU Member States is a major concept and a significant number of papers dealing with Member State price relationships (6). Specifically, the literature focuses on the prices of agricultural products, with some concluding that there is integration in the spatial markets and some concluding the opposite. Meat is our field of interest among the many agricultural products that have been examined so we will focus here. Beef ranks third in the European Union concerning pork, which has been the subject of many studies. Fousekis and Grigoriadis (2019) (6) utilized a sample for the pork market corresponding to ours as it will be described further below, and applying some tools from the Graph theory, they found that prices were related to the natural distance between the Member States and that large countries in significant pork sectors such as production and intra-trade tend to format the prices. Similar conclusions are being sought by the present study too. In another example, Sanjuán and Gil (2001) (11) examined spatial pork and lamb prices relationships in 7 Member States and using multivariate co-integration analysis found a high degree of integration in both markets.

A specific application of the hierarchical analysis method to the European beef market similar to ours from the methodology perspective is the study of Buleca, Kováč and Kočanová (2018) (4), who applied hierarchical analysis to beef production statistics using the euclidean distance as the distance measure and the ward method as the linkage criterion. Once again it is verified that Germany, France, the United Kingdom and Italy account for half of the gross production value of the EU cattle sector. According to Ihle et al., (2017) (13), two are the main attributes of beef prices. The first concerns the fact that beef prices at the consumer level are generally higher than those for other types of meat and secondly that they remain steadily high in recent years. Cattle slaughter prices of different cattle types (steers, young bulls, cows, heifers) increased on average from 2.5 euros in 1991 to almost 3.8 euros per kg in 2015, especially from 2010 to 2015 (13). Probably they will continue to

increase to reach 4.0 euros per kg, so that beef meat and sheep meat will have the same farm prices. This means that beef meat is likely to become even more expensive than pig and poultry meats. The European Union is thus called upon to address this problem, as consumer preferences are matched by the purchasing power and the price level. Also, the emergence of new markets in the world trade of beef makes the market more competitive and the European Union cannot properly repay these high prices. That is why there is an urgent need for market competitiveness and integration in the European Union.

The literature review on the application of the hierarchical algorithm we will use and the different stages of which it consists, is of major importance in this study and it is combined together with the methodology in the next chapter.

3

Data and Methodology

3.1 Data

All data used in the current study are weekly wholesale beef average carcass prices (expressed in Euros per 100kg of carcass weight) for the period 2011-01-03 to 2019-08-26. They have been obtained from the European Commission [1], whose task is to provide EU with statistics at a European level that enable among other comparisons between countries and regions. Agricultural statistics are an important part of its publication and livestock and meat statistics are collected, which includes bovine, pig, sheep and goat livestock; slaughtering statistics on bovine animals, pigs, sheep, goats and poultry; and production forecasts for beef, veal, pig meat, sheep meat and goat meat. The dataset, named "Beef historical weekly prices - 2011 onwards" can be found in the "EU historical series" subsection from the general category "Beef".

As processing the data, we observe that there is a classification of carcasses of bovine animals and price recording [2]. The purpose of the classification is, on the one hand, to pay the producer according to the quality of the carcasses he produces and on the other hand the buyer to choose the quality he wishes and to pay according to the quality he buys. Marketing based on classification contributes substantially to market transparency and its introduction in the market is a major incentive for improving the quality of carcasses. The Community-scale for the classification of carcasses of bovine animals applies to carcasses of bovine animals aged eight months and over and the implementation of this scale is obligatory in all EU Member States. The way of capturing the quality of the carcass is to classify it based on the animal's category, the carcass shape; meaning the degree of muscle mass growth and finally the degree of fattening; indicating the deposition of fat in the carcass.

Starting with the initial, carcasses are classified depending on sex and age of the animal into six categories denoted by the letters Z, A, B, C, D, and E and defined as follows:

[1] https://ec.europa.eu/info/food-farming-fisheries/farming/facts-and-figures/markets/overviews/market-observatories/meat_en

[2] https://ec.europa.eu/info/sites/info/files/food-farming-fisheries/farming/documents/methodology-carcase-remainders_en.pdf

- Z: ("Young cattle"): Carcasses of animals aged from 8 months to less than 12 months.
- A: ("Young bulls"): Carcasses of uncastrated male animals aged from 12 months to less than 24 months.
- B: ("Bulls"): Carcasses of uncastrated male animals aged from 24 months.
- C: ("Steers"): Carcasses of castrated male animals aged from 12 months.
- D: ("Cows"): Carcasses of female animals that have calved.
- E: ("Heifers"): Carcasses of other female animals aged from 12 months.

The shape is defined based on the development of the sides of the carcass, and in particular of the sides of it's most important parts (thigh, back, shoulder blade) in the following categories:

- S: ("Superior"): All profiles extremely convex; exceptional muscle development (double-muscled carcass type).
- E: ("Excellent"): All profiles convex to super-convex; exceptional muscle development.
- U: ("Very Good"): Profiles on the whole convex, very good muscle development.
- R: ("Good"): Profiles on the whole straight; good muscle development.
- O: ("Fair"): Profiles straight to concave; average muscle development.
- P: ("Poor"): All profiles concave to very concave; poor muscle development.

The latter includes five fat classes 1, 2, 3, 4, 5 based on the amount of fat on the outside of the carcass and the inside of the thoracic cavity:

- 1: ("Low"): None up to low-fat cover.
- 2: ("Slight"): Slight fat cover, flesh visible almost everywhere.
- 3: ("Average"): Flesh except for the round and shoulder, almost everywhere covered with fat, slight deposits of fat in the thoracic cavity.
- 4: ("High"): Flesh covered with fat, but on the round and shoulder still partly visible, some distinctive fat deposits in the thoracic cavity.
- 5: ("Very High"): Entire carcass covered with fat; heavy deposits in the thoracic cavity.

Carcass classification applies to all carcasses, all types of shapes and all categories of fat. However, the recording of producer prices for carcasses classified only concerns certain categories of carcasses, namely:

- (Cat Z): U2, U3, R2, R3, O2, O3
- (Cat A): U2, U3, R2, R3, O2, O3
- (Cat B): R3
- (Cat C): U2, U3, U4, R3, R4, O3, O4
- (Cat D): R3, R4, O2, O3, O4, P2, P3
- (Cat E): U2, U3, R2, R3, R4, O2, O3, O4

The market price requested based on the Community grading scale is the price without the value-added tax paid by the supplier of the animal and relates to 100 kg of the carcass, weighed and classified on the hook of the slaughterhouse. The combination we chose for our empirical analysis is ER3 as explained above, in order to consider an animal belonging to the beef category and, with respect to the other 2 categories (carcass shape and degree of fattening), to be of "medium" characteristics.

- E: ("Heifers"): Carcasses of other female animals aged from 12 months.
- R: ("Good"): Profiles on the whole straight; good muscle development.
- 3: ("Average"): Flesh except for the round and shoulder, almost everywhere covered with fat, slight deposits of fat in the thoracic cavity.

Considering then this combination, the data come from 15 member-states; namely, Austria (AT), Belgium (BE) Czechia (CZ), Germany (DE), Denmark (DK), Spain (ES), France (FR), Ireland (IE), Italy (IT), Lithuania (LT), Poland (PL), Portugal (PT), Sweden (SE), Slovenia (SI) and United Kingdom (UK). The selection of the countries was based on two criteria, to include as many countries as possible and to have a sufficient number of observations.

The data were downloaded as a Microsoft Excel file and imported to the programming language R (22). The empirical analysis, as well as data visualization in this study, were exported from the RStudio (24) programming environment. The R coding that was utilized for has been contained as supplementary material in Appendix B.

3.2 Methodology

3.2.1 Clustering

Data mining, also called knowledge discovery, knowledge extraction, data/pattern analysis, information harvesting, etc., is defined as a method used to extract usable data from a larger set of any raw data (5). In particular, data mining is all about discovering unsuspected or previously unknown relationships amongst the data and has applications in multiple fields, like science and research. The invention dates back many decades ago and the area of knowledge mining has been developed

in recent years to address the problem of large volumes of data. It is an important research area with a significant impact on the real world. It involves a set of methods that automate the process of scientific discovery and its uniqueness lies in solving problems with large volumes of data that contain complex and hidden relationships. Tasks performed during data mining are divided into tasks for description and forecasting. Forecasting presupposes the use of various known variables to estimate future unknown values while the description focuses mainly on discover patterns in data that can easily be interpreted and describe them. The contribution of data mining to the science of economics lies in understanding data collections and in the creation and evaluation of a model and its development for any predictions that may be necessary.

Because time series are a ubiquitous and increasingly prevalent type of data, there has been much research effort devoted to time series data mining in recent years. Since data mining has the potential to reveal hidden patterns, the data mining techniques of time series further improve data analysis. Investigations for time series mining focuses on the following processes (5),(16):

- Clustering: Grouping the time series found in the database, based on some similarity or non-similarity measures.
- Classification: Defining a model that can categorize new data.
- Forecasting: Forecasting the $n + 1$ value of a given time series all of its previous points.
- Segmentation: The construction of a model divided into k segments (with k « n) to approximate a time series.
- Summarization: Including methods for describing a subset of data.
- Anomaly Detection: Identification of unusual data records or observations, which raise suspicions by differing significantly from the majority of the data and require further investigation.

In the past few years, tasks such as regression, classification, clustering or segmentation have been extended and modified successfully for time-series databases (Fu, 2011 (7); Bagnall et al., 2016 (2)) . Of the above methods, in this study we are interested in implementing clustering. In many real applications, clustering must be performed on time-series data (LIAO, 2005 (17) ; Fu, 2011 (7)), since it is a common type of dynamic data that naturally arise in many different circumstances, such as economics, finance, medical data, ecology, environmental studies or engineering, just to name a few. The main features of this type of data are its high dimensionality, dynamism, auto-correlation and noisy nature, all of which complicate the study and pattern extraction to a large extent.

Clustering or cluster analysis, which is a descriptive task as explained above, is the main technique used to divide data into groups based on internal and priori unknown schemes inherent of the data (9). In more detail, clustering is the task

aiming to group or classify a set of objects in such a way that objects in the same group (called a cluster) have the same kind of characteristics compared to those in other groups (clusters). In other words, in each group, we want as much as possible internal homogeneity and a great relationship between its elements, while between groups as much as possible heterogeneity and minimum relationship. The original view one has when given multivariate data is unclear and it is very difficult to draw conclusions about them, so with the help of clustering, we try to gain some extra knowledge about our data, such as similarities, presence or absence of features and find out if there are any relationships that characterize them.

3.2.2 Distance Measures

Since clustering is the grouping of similar objects, the choice of how to calculate the similarity/dissimilarity between the two objects is a crucial subject. Most clustering algorithms use metric spaces or distance measures to determine the similarity or dissimilarity between any pair of objects. We consider similarity as the measure that establishes an absolute value of resemblance between two vectors, in principle isolated from the rest of the vectors and without assessing the location inside the solution space (12).

It is useful to denote the distance between two points x and y as: $\mathrm{d}(x, y)$. A valid distance measure should be symmetric and obtains its minimum value (usually zero) in case of identical vectors. The distance measure is called a metric distance measure if it also satisfies the following properties:

$$\mathrm{d}(x, y) \geq 0$$

$$\mathrm{d}(x, y) = 0$$

$$\mathrm{d}(x, y) = \mathrm{d}(y, x)$$

$$\mathrm{d}(x, z) \leq \mathrm{d}(x, y) + \mathrm{d}(y, z)$$

There exists a broad range of measures to compare time series and the choice of the proper dissimilarity measure depends largely on the nature of the clustering, i.e., on determining what the purpose of the grouping is. Many dissimilarity measures between time series have been proposed in the literature. Following Montero and Vilar et al. (2014) (19), they can be grouped into four categories: model-free measures, model-based measures, complexity-based measures and prediction-based measures. This study uses two model-free measures, which include metrics based on the closeness of their values at specific points of time and they are described further below.

3.2.2.1 Euclidean distance

Scientists initially, aiming to objectively determine the similarity between time series, suggested the use of Minkowski distance metric, the general type of which is:

$$\mathrm{d}(x,y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p}$$

The most common distance which is used is the Euclidean distance which occurs of the general type of Minkowski distance, for $p = 2$. Mathematically, the definition of Euclidean distance between two n-dimensional vectors $x(x_1, ..., x_n)$ and $y(y_1, ..., y_n)$ is:

$$\mathrm{d}(x,y) = \left(\sum_{i=1}^{n} |x_i - y_i|^2 \right)^{1/2}$$

Some other widely used measures of dissimilarity or otherwise distance measures that occur for different values of p are the Manhattan ($p = 1$) and Chebyshev ($p = \infty$)

Euclidean distance, although it works well, does not always produce accurate results when the sequences shift with respect to the time axis. In this case another more efficient metric called Dynamic Time Warping can be used and it is discussed in more detail below

From the preceding, it is understood that the Euclidean distance calculates the point by point distance. As a result, its utilization in trying to calculate time series distances requires that they have exactly the same number of observations. Thus, Euclidean distance, although it works well, does not always produce expensive results when the sequences shift with respect to the time axis. This disadvantage has resulted in the invention and implementation of a new method that supports the comparison and calculation of distances between time series of independent dimensions, called Dynamic Time Warping distance.

3.2.2.2 Dynamic Time Warping distance

According to Montero and Vilar (2014) (19), the dynamic time warping (DTW) distance was studied in depth by Sankoff and Kruskal (1983) (26) and proposed to find patterns in time series by Berndt and Clifford (1994) (3). Given the time series Q and C of dimension n and m respectively, to calculate the DTW distance, the warping path must be calculated first. This path consists of a set of consecutive elements of the table, which defines a non-linear matching match between Q and C. That is $W = w_1, ..., w_2$, with $max(m,n) \leq K < m + n - 1$ and $w_k(i,j)$ the k element of W. This path can be dynamically found through the following iteration (21):

$$w(i,j) = \mathrm{d}(q_i, c_i) + min\{w(i-1, j-1), w(i-1, j), w(i, j-1)\},$$

To align the time series with DTW, a dimension table is formed (mxn) where the item (i,j) contains the distance $\mathrm{d}(q_i, c_j)$ between the two points q_i and c_j (where $\mathrm{d}(q_i, c_j) = (q_i - c_j)^2$). Each item ($i$,$j$) in the register corresponds to the alignment

between the two points q_i and c_j. The distortion length is a parameter that calculates the optimum distance specifying the permitted distortion.

Basic restrictions on the Warping Function [3] :

- Boundary Conditions: $i_1 = 1$, $i_k = n$ and $j_1 = 1$, $j_k = m$: The alignment path starts at the bottom left and ends at the top right. This restriction guarantees that the alignment does not consider partially one of the sequences.

- Monotonicity: $i_s - 1 \leq i$ and $j_s - 1 \leq j$: The alignment path does not go back in "time" index. This restriction guarantees that features are not repeated in the alignment.

- Continuity: $i_s - i_{s-1} \leq 1$ and $j_s - j_{s-1} \leq 1$: The alignment path does not jump in "time" index. This restriction guarantees that the alignment does not omit important features.

- Warping Window: $|i_s - j_s| \leq r$ where r > 0 is the window length : A good alignment path is unlikely to wander too far from the diagonal. This restriction guarantees that the alignment does not try to skip different features and gets stuck at similar features.

Many warp paths meet these conditions, but the preferred path is the one that minimizes the cost of distortion:

$$DTW(Q, C) = min\left(\frac{\sqrt{\sum_{k=1}^{k} W_k}}{K}\right)$$

The different approach of the metric DTW compared to that of Euclidean distance, but also of the other point-to-point metrics mentioned above, plays a decisive role in the results obtained from data mining processes. The latter is a special case of DTW, where the sequences are aligned point by point, that is, the i-th point of time series C corresponds to its i-th point of time series Q. Keogh and Ratanamahatana (2005) (15) concluded that the error frequencies in the case of clustering of the time series to be analyzed based on metric DTW, are much lower than those transpiring in the case of time series analysis clustering with the application of the Euclidean distance. The disadvantage of DTW exists in the time and cost of computing, and according to these features may not be the best choice for large databases. The complexity of computing DTW is computationally costly with complexity O(m*n) where m and n represent the length of each sequence.

[3] `http://www.mathcs.emory.edu/~lxiong/cs730_s13/share/slides/searching_sigkdd2012_DTW.pdf`

3.2.3 Hierarchical Clustering

Perhaps the most popular clustering algorithm, hierarchical clustering, is a method that tries to create a hierarchy of groups/clusters in which, as the level in the hierarchy increases, clusters are created by merging the clusters from the next lower level, such that an ordered sequence of groupings is obtained (9). To decide how the merging is performed, a (dis)similarity measure between groups should be specified, in addition to the one that is used to calculate pairwise similarities (as those mentioned above). The partitions represent nonoverlapping clusters and have the property that once two elements become members of the same cluster, they are never again seperated. The researcher has the option of using the entire hierarchy as the solution or selecting a level representing the specific number of clusters of interest.

Algorithms for hierarchical clustering can be agglomerative or divisive (9). In the first method, which is also known as Hierarchical Agglomerative Clustering (HAC), we consider each member of the data as a cluster. Then, more similar memebers are selected and merged based on the similarity measure, creating a new cluster. The process is repeated until all members are contained in a single cluster. Divisive clustering method, also called DIANA, which is an acronym for Divisive Analysis, follows a pattern that is the reverse of the agglomerative technique. It starts with all data in one single cluster and then divide them into two least similar clusters. This is repeated recursively on each cluster until there is one cluster for each member. In this study, we will proceed with Agglomerative Clustering for the rest of the study, since HAC accounts for the majority of hierarchical clustering algorithms while Divisive methods are rarely used.

After selecting a distance metric, it is necessary to determine from where distance is computed. The inter-group dissimilarity is also known as linkage. There are certain approaches which are used to calculate the similarity between two clusters:

- SINGLE LINKAGE: It is also known as minimum method. Here, the distance between one cluster and another cluster is taken to be equal to the shortest distance from any data point of one cluster to any data point in another. That is, distance will be based on similarity of the closest pair of data points. Mathematically, the linkage function – the distance $\mathcal{D}(A, B)$ between clusters A and B – is described by the expression :

  $$\mathcal{D}(A, B) = \min_{a \in A, b \in B} \mathrm{d}(a, b)$$

  where A and B are any two sets of elements considered as clusters, and $\mathrm{d}(a, b)$ denotes the distance between the two elements a and b. An advantage of this approach is that it can separate non-elliptical shapes as long as the gap between two clusters is not small. However, it can cause premature merging of groups with close pairs, even if those groups are quite dissimilar overall.

- COMPLETE LINKAGE: This method is also called the diameter or maximum method. In this method, we consider similarity of the furthest pair. That is,

the distance between one cluster and another cluster is taken to be equal to the longest distance from any member of one cluster to any member of the other cluster. Similarly, mathematically, the linkage function – the distance $\mathcal{D}(A, B)$ between clusters A and B – is described by the expression :

$$\mathcal{D}(A, B) = \max_{a \in A, b \in B} \mathrm{d}(a, b)$$

where A and B are any two sets of elements considered as clusters, and $\mathrm{d}(a, b)$ denotes the distance between the two elements a and b. This approach does well in separating clusters if there is noise between clusters but a disadvantage is that outliers can cause close groups to be merged later than what is optimal.

- AVERAGE LINKAGE: In average linkage, we take the distance between one cluster and another cluster to be equal to the average distance from any member of one cluster to any member of the other cluster. For example, the distance between clusters A and B to the left is equal to the average length each arrow between connecting the points of one cluster to the other. Here, the linkage function – the distance $\mathcal{D}(A, B)$ between clusters A and B – is described by the expression :

$$\mathcal{D}(A, B) = \frac{1}{n_A n_B} \sum_{i=1}^{n_A} \sum_{j=1}^{n_B} \mathrm{d}(a, b)$$

where A and B are any two sets of elements considered as clusters, and $\mathrm{d}(a, b)$ denotes the distance between the two elements a and b. The advantage of the method is that it does not create large length clusters nor do we have such a severe problem with extreme observations. Cons, is that because we have to calculate the average distance between the clusters, it has bigger computational cost. It is also biased towards globular clusters.

- AVERAGE CENTROID: Being a less popular technique, it computes the centroids of two clusters A and B and take the similarity between the two centroids as the similarity between two clusters.

- WARD'S METHOD: Ward's method aims to minimize the total within-cluster variance. At each step the pair of clusters with minimum between-cluster distance are merged. In other words, it forms clusters in a manner that minimizes the loss associated with each cluster. At each step, the union of every possible cluster pair is considered and the two clusters whose merger results in minimum increase in information loss are combined. Here, information loss is defined by Ward in terms of an error sum-of-squares criterion (ESS). Ward's method approach also does well in separating clusters if there is noise between clusters and it is also biased towards globular clusters.

Finaly, the created hierarchy can be visualized as a binary tree of clusters created based on the connection criterion. Such a plot is called a dendrogram and it is in the form of an inverted tree. Dendrograms begin with each object in a separate cluster. At each step, the two most similar clusters are joined into a single new cluster. Once fused, objects are never separated. Usually, the horizontal axis of the dendrogram represents the objects and clusters while the vertical axis the distance or dissimilarity between clusters. The dendrogram is fairly simple to interpret and the reverse visualization can also happen. The dendrogram does not directly imply a certain number of clusters, but one can be induced. One option is to visually evaluate the dendrogram to assess the height at which the largest change in dissimilarity occurs, consequently cutting the dendrogram at said height and extracting the clusters that are created. Another option is to specify the number of clusters that are desired, and cut the dendrogram in such a way that the chosen number is obtained. In the latter case, several cuts can be made, and validity indices can be used to decide which value yields better performance.

3.2.4 Tools for time series clustering

Although hierarchical clustering provides a fully connected dendrogram representing the cluster relationships, sometimes we need to choose the preferred number of clusters to extract. The optimal number of clusters is somehow subjective and depends on the method used for measuring similarities and the parameters used for partitioning. So defining different methods for determining the optimal number of clusters in hierarchical clustering is necessary to complete the process. These methods include direct methods such as the elbow and silhouette method or other statistical testing methods. An example of the latter is the gap statistic. In addition to elbow, silhouette and gap statistic methods, there are more than thirty other indices and methods that have been published, but we will remain from now to these three [4] :

- ELBOW METHOD: It is probably the most well-known method, in which the sum of squares at each number of clusters is calculated and graphed. After that, we look for a change of slope from steep to shallow (an elbow) to determine the optimal number of clusters. More specific, the Elbow method looks at the total WSS (within-cluster sum of square) as a function of the number of clusters. One should choose a number of clusters so that adding another cluster doesn't improve much better the total WSS.

- SILHOUETTE METHOD (23): Another illustration that can help determine the optimal number of clusters is called the silhouette method. The mean silhouette method calculates the average silhouette of observations for different values of k. The optimal number of k groups is the one that maximizes the mean silhouette over a range of possible values.

[4] https://towardsdatascience.com/10-tips-for-choosing-the-optimal-number-of-clusters-277e93d72d92

- GAP STATISTIC (28): The gap statistic compares the total within intra-cluster variation for different values of k with their expected values under null reference distribution of the data. The estimate of the optimal clusters will be a value that maximizes the gap statistic.

4 Empirical Analysis and Discussion

4.1 Descriptive Statistics

In this section, we give a brief presentation of our sample, describe the basic features of our observations that will lead to some initial results and together with some helpful presentation of diagrams, give them a visual comprehensible form.

4.1.1 Raw data

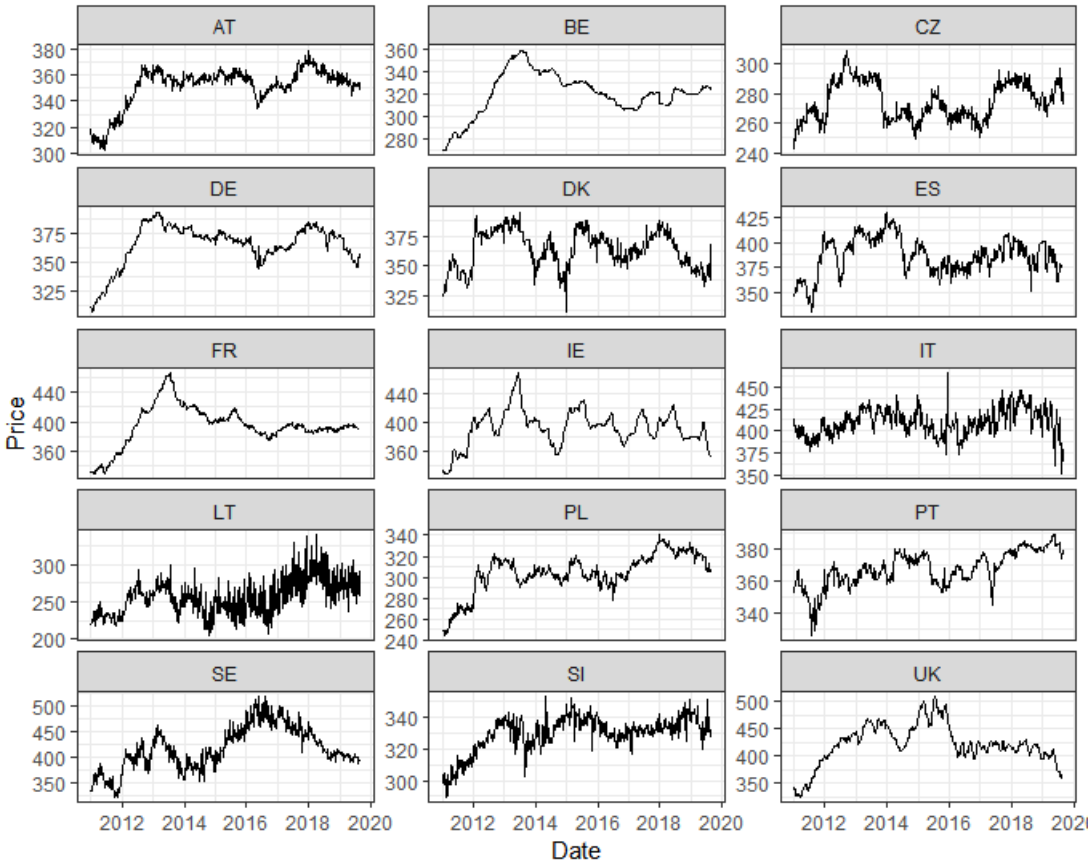


Figure 4.1: Raw prices over time (weeks) for the fifteen EU beef markets.

Figure (4.1) shows the evolution of the time series for the 15 EU markets. We observe that prices in some countries fluctuate similarly while in others the opposite happens. In almost all countries, prices are increasing until 2012-2013, reflecting the steady increase in the following ten years after the price drop in 2001-2003, due to the mad cow and foot-and-mouth disease crises. Over the period 2014 to 2018 beef prices stay relatively firm despite the Russian import ban and the restructuring in the EU dairy sector.

Table (4.1) presents some useful descriptive statistics, concerning the measures of location, the measures of dispersion and the coefficients of skewness and kurtosis. These are very significant variables since check for normality can be done if compare the mean and the median values or estimate the coefficient of skewness and kurtosis. Other ways to examine normality are the use of graphs (histograms, normal Q-Q plots and boxplots) and the application of appropriate statistical tests.

Table 4.1: Descriptive Statistics for the raw prices

Country	N	Mean	Median	Min	Max	Range	SD	Skewness	Kurtosis
DE	452	365.630	370.480	307.020	393.720	86.700	17.625	-1.237	1.297
FR	452	395.199	394.000	329.000	466.000	137.000	27.195	-0.162	0.740
UK	452	422.124	420.783	325.340	509.013	183.673	36.615	-0.323	0.426
IT	452	409.798	409.805	351.040	465.390	114.350	15.783	-0.100	0.295
IE	452	392.817	394.850	329.330	467.940	138.610	24.697	-0.073	0.751
PL	452	304.205	305.490	244.192	340.261	96.069	18.957	-1.008	1.008
ES	452	386.605	386.750	330.531	430.412	99.881	18.316	-0.311	-0.179
AT	452	351.443	355.560	302.180	379.000	76.820	15.212	-1.522	1.755
PT	452	367.804	368.250	326.200	389.000	62.800	11.082	-0.567	0.304
SI	452	329.434	331.295	289.620	352.570	62.950	11.120	-1.024	1.001
SE	452	415.774	409.301	321.131	517.025	195.894	42.084	0.107	-0.525
BE	447	321.068	322.000	270.000	359.500	89.500	19.340	-0.424	0.102
CZ	452	274.670	272.340	241.563	309.219	67.656	12.916	0.205	-0.981
DK	452	363.406	364.737	310.762	395.328	84.566	15.769	-0.307	-0.709
LT	446	256.064	254.030	205.103	339.570	134.467	25.152	0.524	-0.047

Firstly, we observe that 13 countries have a complete set of observations (N = 452), while 2 countries contain missing values. These are Belgium (NAs = 5) and Lithuania (NAs = 6). The coefficients of mean and median of a continuous variable can be used for an initial assessment of normality. In particular, the closer the mean and the median are, the more it is likely that the quantitative variable will follow the normal distribution. Then, the standard deviation shows how much variation from the mean exists. It represents a "typical" deviation from the mean. A low standard deviation indicates that the data points tend to be very close to the mean while a high standard deviation indicates that the data points are spread out over a large range of values. From the table (4.1), the time series with the smallest (SD) are the most predictable and follow the flattest course among the European countries.

Next, we have estimated the coefficients of skewness and kurtosis, which offer valuable information on whether or not a continuous variable follows the normal distribution. In more detail, the skewness coefficient takes values from -3 to 3, with values from [-1,1] denote the existence of a normal distribution and values from [-1,-3] or from [1,3] denote no normal distribution. According to the table, most of the countries seem to be in the [-1,1] interval. Sweden, Czechia, and Lithuania have positive skewness, meaning the mass of the distribution is concentrated on the left. The rest countries on the contrary have left-skewed distributions, so the left tail of their distributions is longer.

As we stated before, diagrams are essential tools that communicate information and explain statistical data. Figure (4.2) shows classic horizontal boxplots containing all the countries, where a nice summary of our variables is presented. Note that the white marks on the boxplot represent the mean of the beef prices so that we have a good visualization of the previous mention about the measures of location and their comparison.

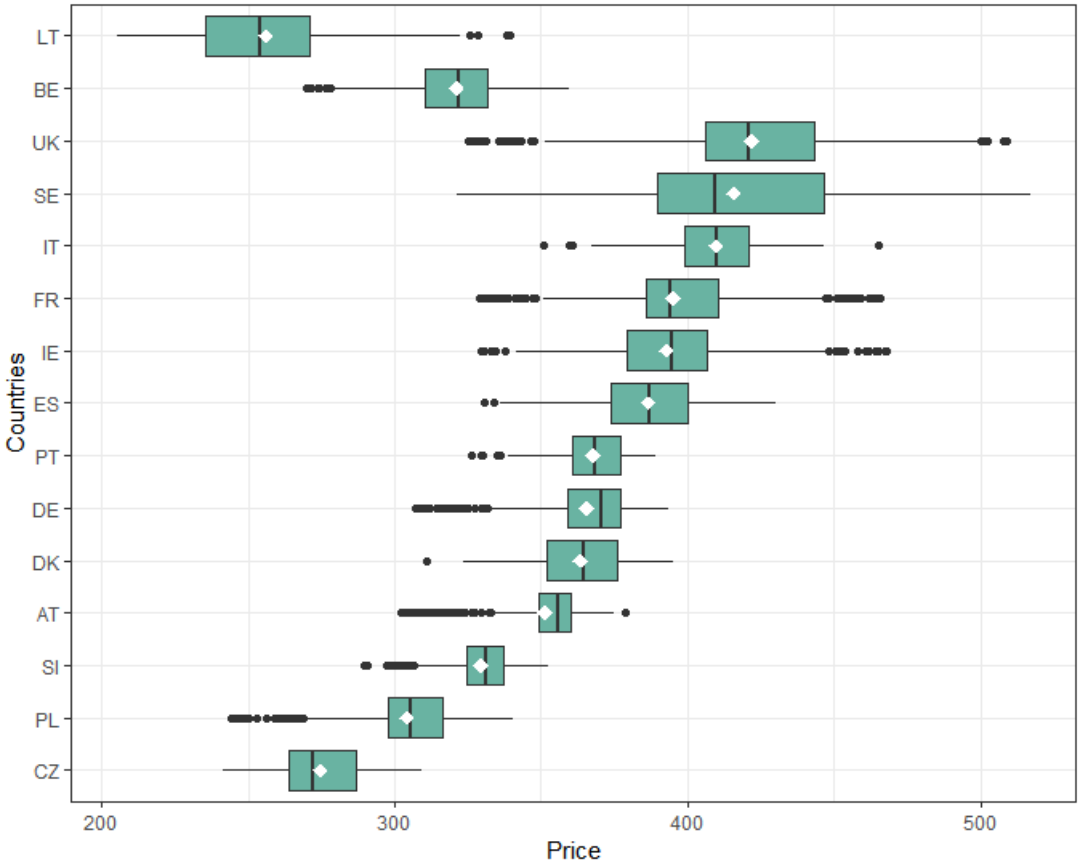


Figure 4.2: Boxplots of the raw prices

Figures (4.3) and (4.4) depict the histograms and density plots (a smoothed version of the histograms) for our variables respectively to have a graphical representation of the distribution and to assist our briefly description above for coefficients such as standard deviation, skewness and kurtosis.

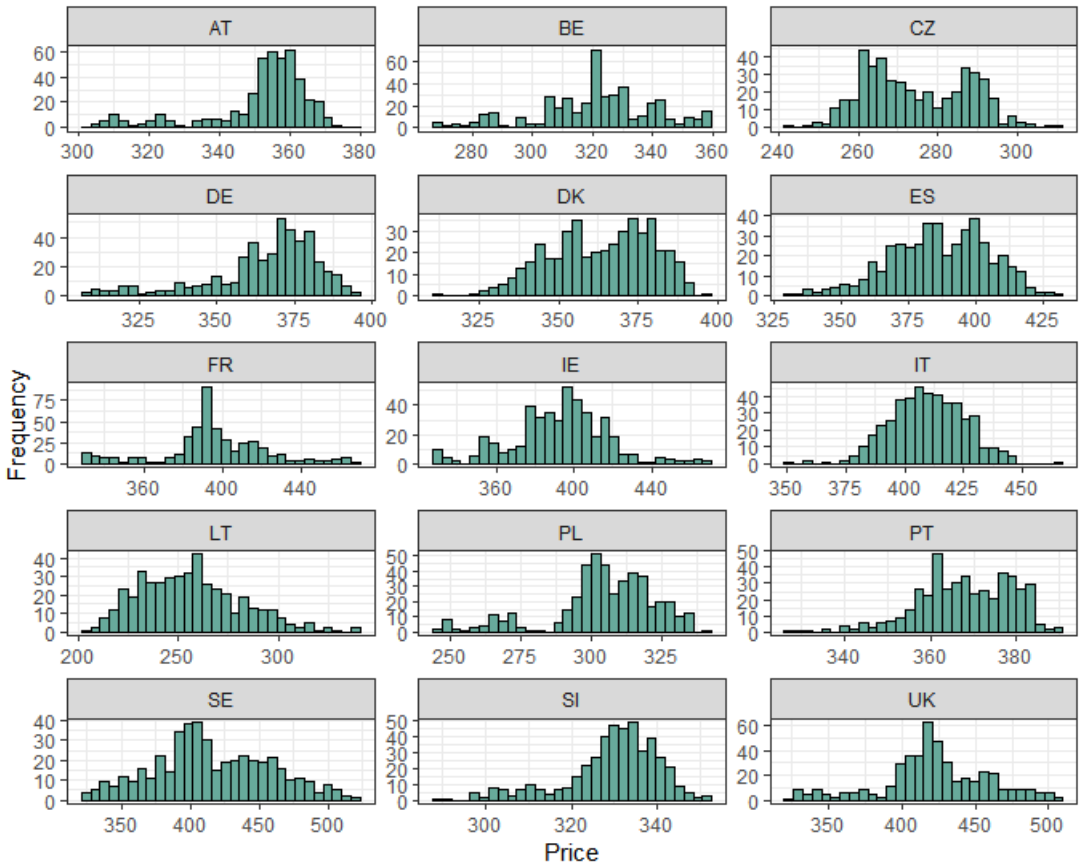


Figure 4.3: Histograms of the raw prices

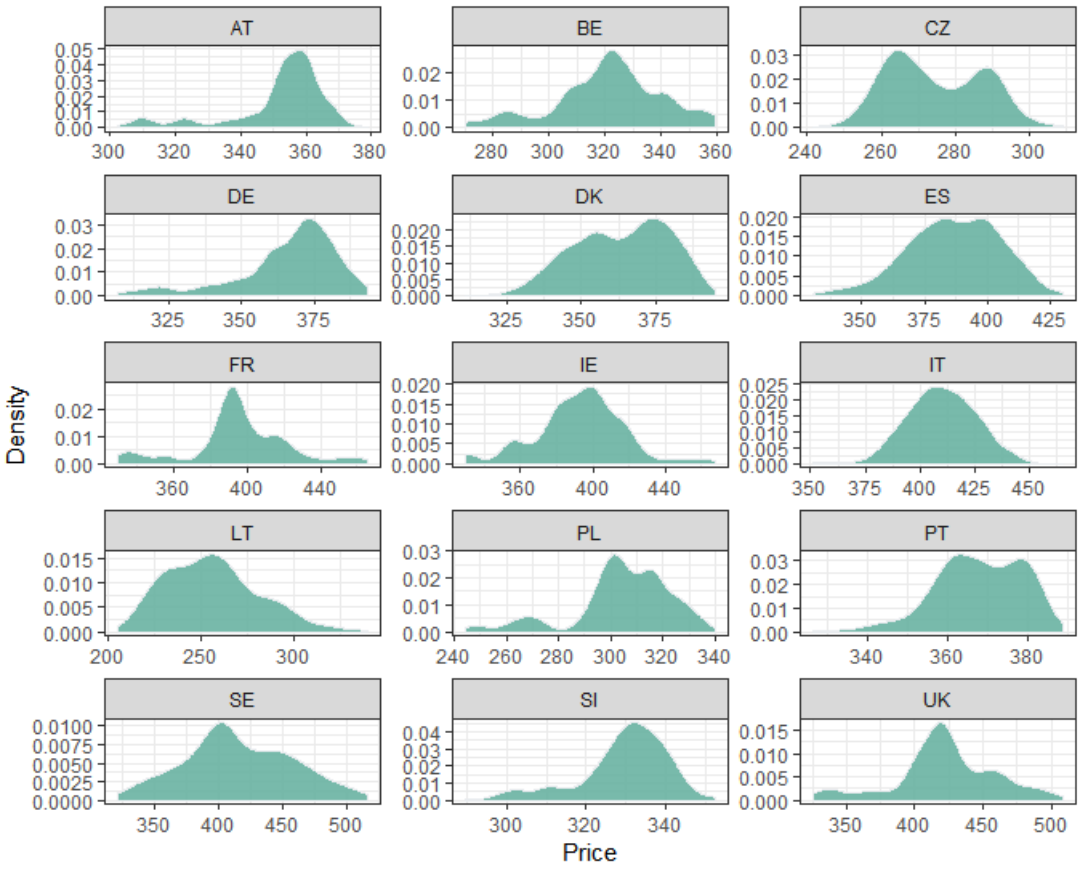


Figure 4.4: Density plots of the raw prices

Last but not least, we use the correlation coefficient as a measure of association to help us take a first look and quickly identify the most correlated variables. The correlation coefficient (also known as the Pearson correlation coefficient) measures how well two variables are related in a linear (straight line) fashion. It is usually called r and lies between -1 and +1. A value of r = -1 means that the two variables are exactly negatively correlated and a value of r = +1 means that the two variables are exactly positively correlated. A value of r = 0, means that the two variables are not linearly related. There are several ways for visualizing a correlation matrix. A good way to quickly check correlations among our variables is by visualizing the correlation matrix as a heatmap (figure 4.5). There are also other different methods for correlation analysis such as Spearman (figure A5) and Kendall (figure A6), which are non-parametric rank-based correlation tests and presented in the same way in the Appendix A.

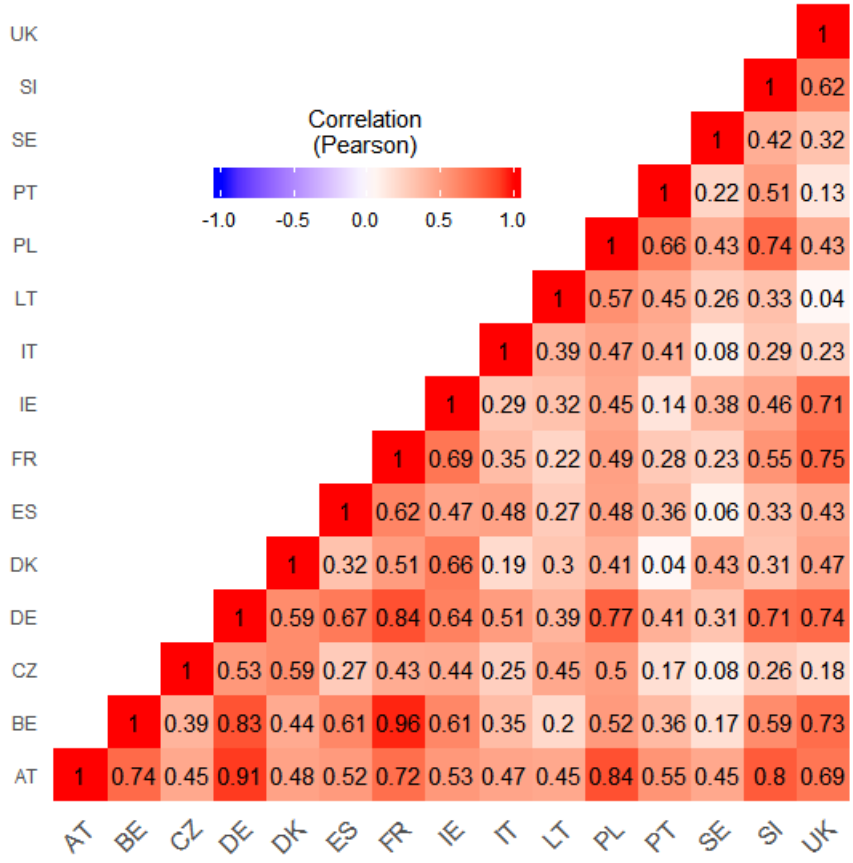


Figure 4.5: Correlation heatmap (pearson) of the raw prices

4.1.2 Log-Returns

For practical purposes, as price series in levels are mostly non-stationary and price differences are stationary, the bibliography always concentrates on log price-returns because they simply eliminate the non-stationary properties of the data, making them more stable. The price change (P_{ct}) is calculated as $P_{ct} = log(P_t/P_{t-1})$, where P_t and P_{t-1} are current and one period lagged weekly spot prices respectively.

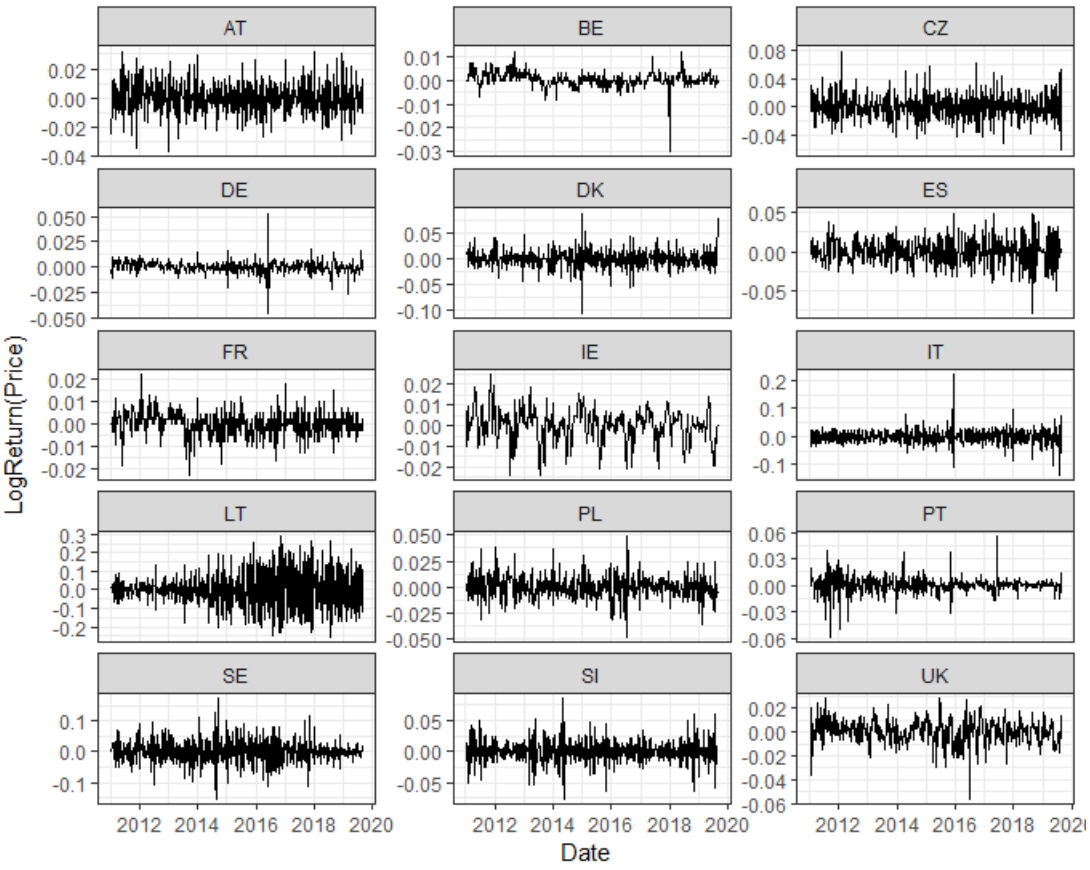


Figure 4.6: Log-return (prices) over time (weeks) for the fifteen EU beef markets.

An interesting analysis is to plot once again the weekly log return prices of the fifteen time series. It is verified that the log-return graphs (figure 4.6) present an oscillation around the value of zero. It is also possible to observe the presence of volatility clustering at some moments and periodic clustering of high and low returns, suggesting that maybe the log return process is not completely independent.

Table (4.2) shows the main descriptive statistics of the weekly log return series. As we can see, the mean of log returns is zero and the min and max are also approximately around zero. Some countries present greater volatility, as perceived by the greater standard deviation mean (SD) they have (e.g., Italy, Sweden, Lithuania). The skewness values showed in the table indicate that the distributions of the return prices are more symmetrical than earlier and from figures (4.7) and (4.8), we understand that we now have more bell-shaped density distributions. However, excess of kurtosis is being presented in some variables, as we expected, which in the literature are conventionally known as 'narrow peak' and 'fat tails', being characterized as leptokurtic distributions.

Table 4.2: Descriptive Statistics for the log-return prices

Country	N	Mean	Median	Min	Max	Range	SD	Skewness	Kurtosis
DE	451	0.000	0.000	-0.046	0.053	0.099	0.006	0.198	17.960
FR	451	0.000	0.000	-0.023	0.022	0.045	0.006	-0.206	1.310
UK	451	0.000	0.001	-0.056	0.028	0.085	0.010	-0.459	1.988
IT	451	0.000	0.000	-0.139	0.221	0.360	0.027	0.631	10.887
IE	451	0.000	0.001	-0.024	0.024	0.048	0.008	-0.389	0.656
PL	451	0.000	0.001	-0.048	0.048	0.096	0.012	0.105	1.597
ES	451	0.000	0.000	-0.080	0.050	0.130	0.017	-0.171	0.977
AT	451	0.000	0.000	-0.037	0.032	0.069	0.011	0.020	0.155
PT	451	0.000	0.000	-0.059	0.055	0.114	0.010	-0.395	8.910
SI	451	0.000	0.000	-0.076	0.085	0.161	0.019	-0.083	1.977
SE	451	0.000	0.000	-0.153	0.172	0.325	0.039	0.119	1.451
BE	443	0.000	0.000	-0.030	0.012	0.042	0.003	-1.682	20.936
CZ	451	0.000	0.000	-0.062	0.078	0.140	0.019	0.281	0.786
DK	451	0.000	0.000	-0.105	0.088	0.193	0.018	-0.032	4.657
LT	439	0.000	0.003	-0.255	0.286	0.541	0.109	0.063	-0.432

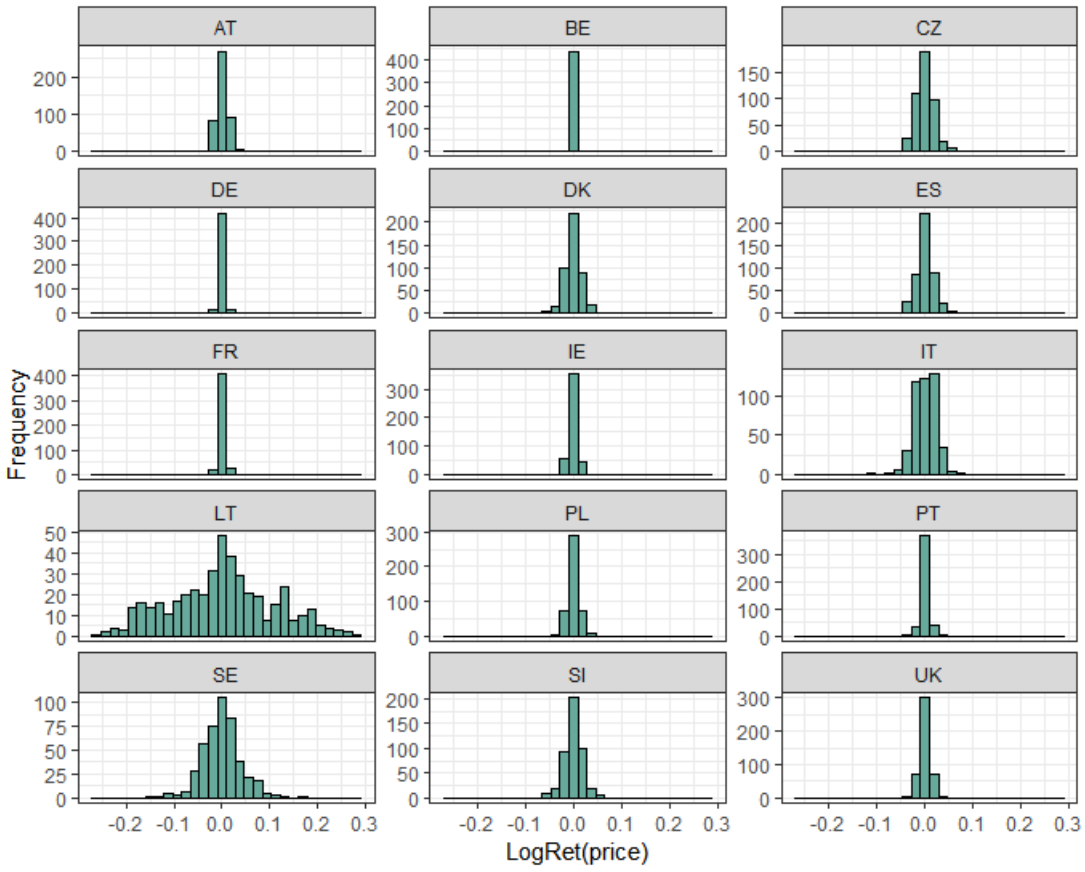


Figure 4.7: Histograms of the log-return prices

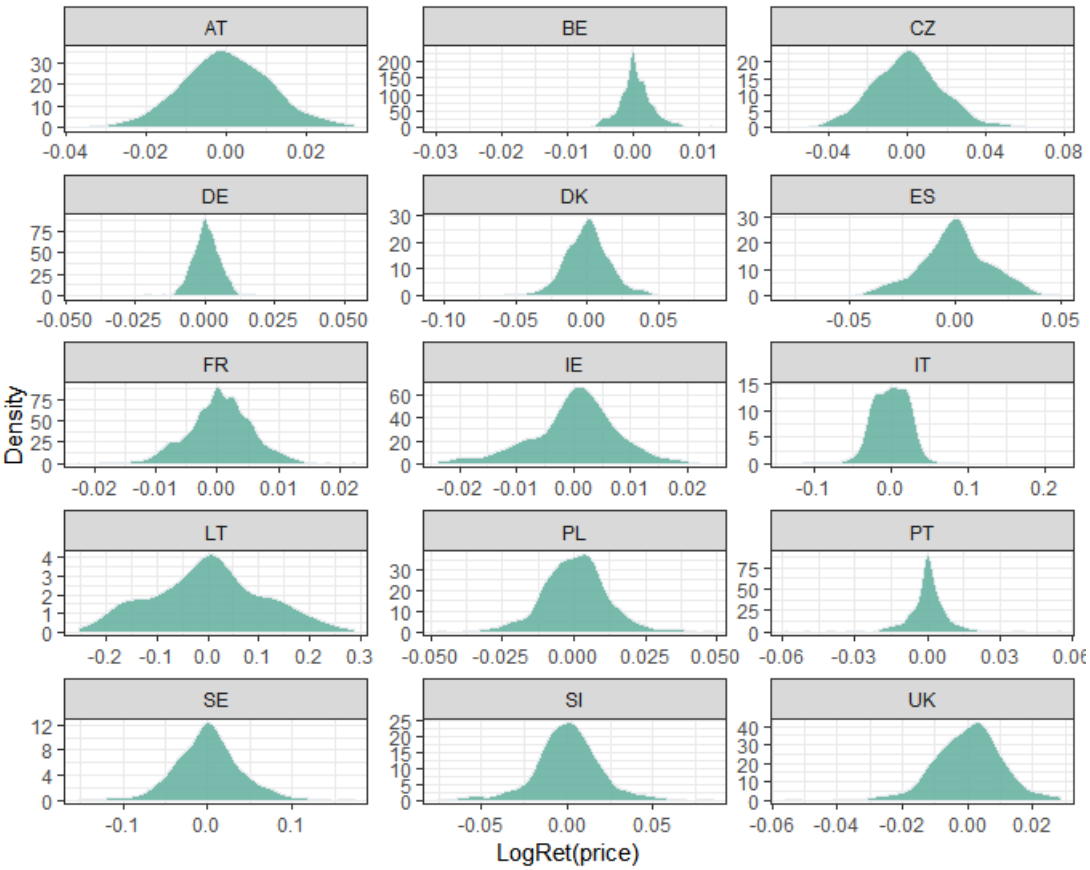


Figure 4.8: Density plots of the log-return prices.

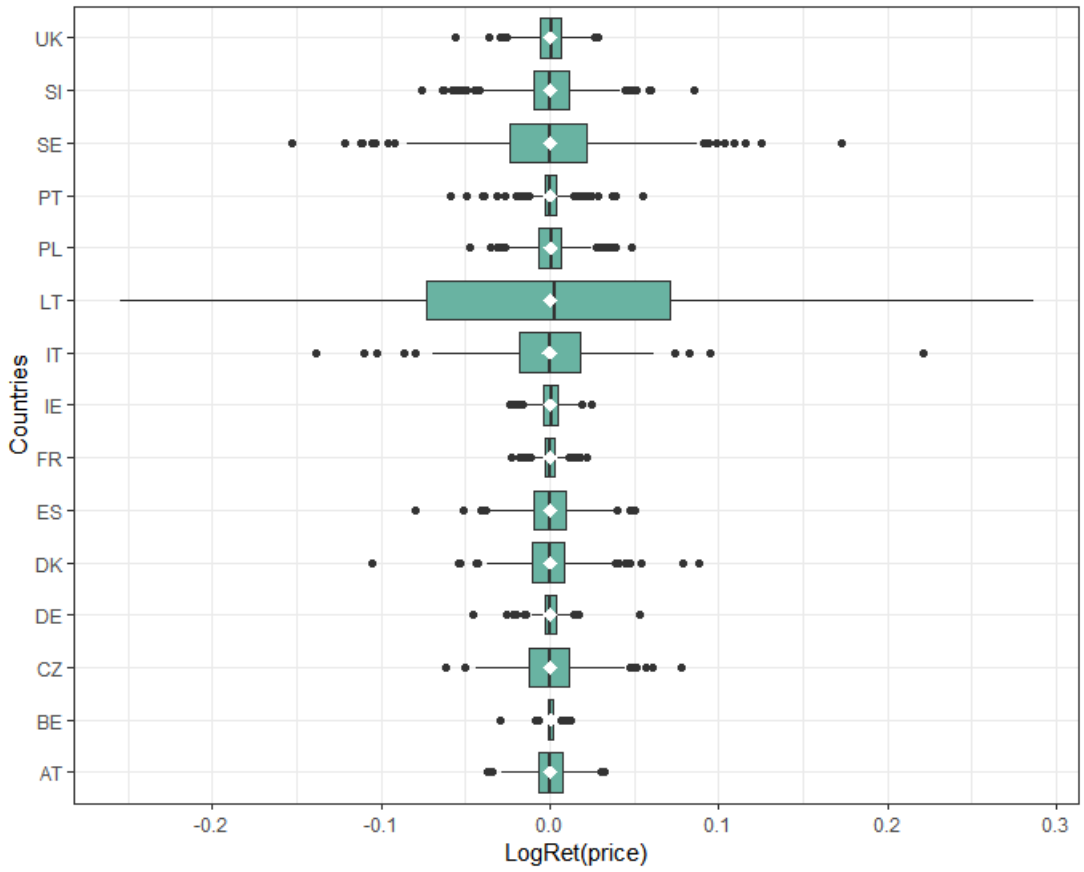


Figure 4.9: Boxplot of the log-return prices

As previously in the raw prices subsection, the boxplots (figure 4.9) are listed above. Once again, the white marks correspond to the means of the log-return prices. In addition, in Appendix A, there are also other graphs such as Q-Q plots, which completely satisfy the descriptive statistics and data visualization sectors for the variables, both for the raw and log-return prices.

4.2 Empirical analysis

For the empirical analysis, to find the similarities in observations and group the selected beef markets in the European Union, we illustrated hierarchical agglomerative clustering as described in chapter 3. We chose this method because it is mainly used in the bibliography and other empirical studies compared to the divisive hierarchical clustering. From now on, and for the rest of this study, we will refer to the hierarchical agglomerative clustering simple as hierarchical clustering. Since this method is not automated but is subject to some degree of subjectivity in its process, we endeavor to make a comprehensive presentation and through some methods, extract conclusions about how the countries are grouped.

Concerning data transformations, among the different practices that can be implemented and taking into account the type of data we have, we demonstrated

hierarchical clustering to three different data sets. The first concerns raw data, that is, basically non transformed data, as presented in detail in the descriptive statistics. Then the data needed to be standardized and so we have the second set of data. To perform cluster analysis in R, generally, the data should be standardized to make variables comparable. Standardization consists of transforming the variables such that they have mean zero and standard deviation one. Thus, our clustering algorithm won't depend on an arbitrary variable unit. The third and last set of data includes a logarithmic transformation and more specifically the log-returns as previously described in the descriptive statistics. For each data type, we converted the data sets into a format that can be quickly inspected by R and removed all the NA values present in the data, since the distances require that at least one variable have non-missing values for each pair of rows. Conclusively, that left us with 441 observations for both raw and scaled data and 431 for the log-returns.

Once we have defined the type of algorithm and the data sets, the next crucial parameter is the choice of the distance measure. The choice of distance measures is very important, as it has a strong influence on the clustering results. Several different distance measures could be used and one must consider the data at hand and the assumptions of each measure to select the appropriate method. In the present study, we applied the Euclidean and Dynamic Time Warping (DTW) distances as previously described in the methodology section. The first was chosen because it is the most commonly used distance measure for clustering and since our time series are of equal length, we overcome euclidean distance limitations. DTW, whose advantages have been described above, is also often used since it groups time series according to their patterns or shapes even if these patterns are not synchronized.

In the next phase, after selecting the distance metric, it is necessary to determine the linkage criterion. Many linkage criteria have been developed and as with distance measures, the choice should be made based on theoretical considerations from the domain of application. Where there are no clear theoretical justifications for the choice of linkage criterion, ward's method is the sensible default. However, we will include and test all four main linkage criteria namely Average, Single, Complete and Ward.

To summarize, the hierarchical algorithm was implemented with all possible combinations of data sets, distance measures and linkage criteria as they selected and already explained. This means we demonstrated 3 (data sets) x 2 (distance measures) x 4 (linkage criteria) = 24 different dendrograms, from which we tried to come up with an ideal dendrogram for each set of data and through some methods, extract the appropriate number of clusters in each case. Remember once again the subjectivity that underlies the whole process. There are different packages and functions available in R for computing hierarchical clustering. For this study, we used the stats (22) and cluster (18) packages, while factoextra (14) package was very useful for the clustering visualization. Other papers related to R which were also helpful to this study were (20), (8), (27), (19).

4.2.1 Raw data scenario

In this section, we present all the results of the hierarchical clustering for the raw dataset. Starting by taking into consideration the euclidean distance, table (4.10) shows an initial visualization of the dissimilarity matrix. In this plot, the blue color corresponds to the small distance while the red color indicates the large distance between the countries. Observing the heatmap, we see that similar observations are close to one another and a first conclusion as it seems is that 3 or 4 clusters of countries will be created. In Appendix A, it is also cited the distance matrix presenting the dissimilarity values which we feed into the algorithm. Specifying then all four agglomerative methods as described above (i.e. "ward", "complete", "average", "single"), we plotted the four corresponding dendrograms (figures 4.11, 4.12, 4.13, 4.14).

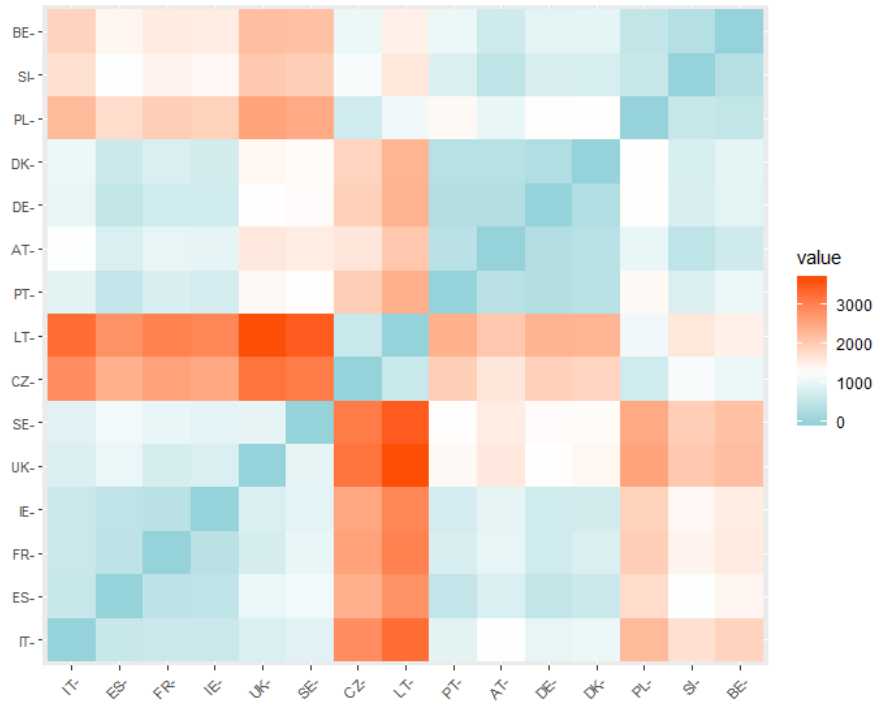


Figure 4.10: Heatmap based on the euclidean dissimilarity matrix, raw data

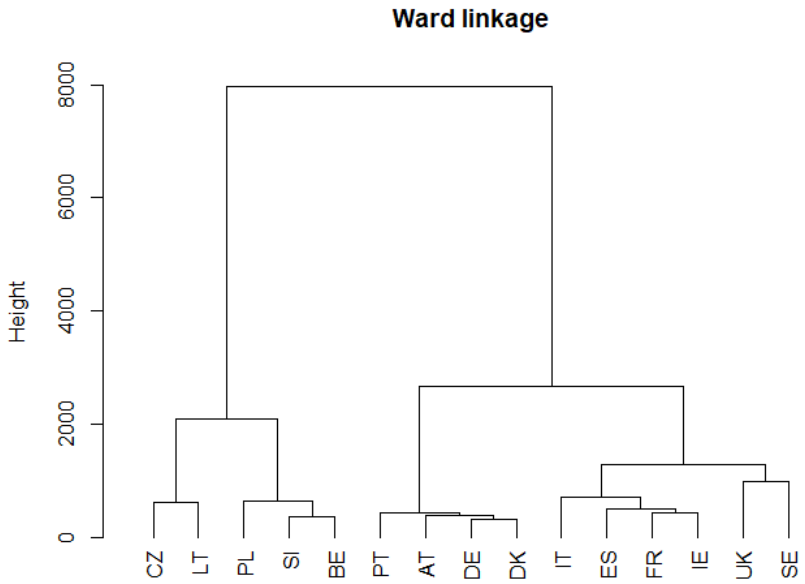


Figure 4.11: Dendrogram of the raw dataset, euclidean distance and ward linkage

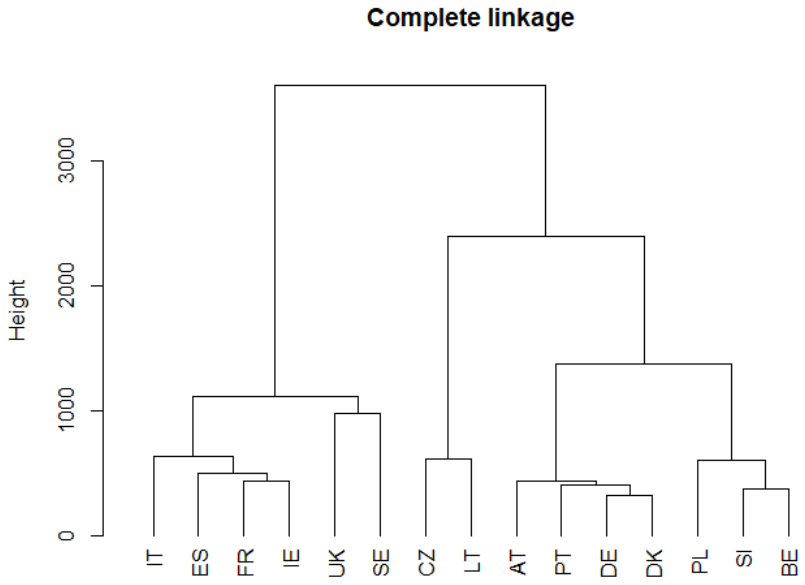


Figure 4.12: Dendrogram of the raw dataset, euclidean distance and complete linkage

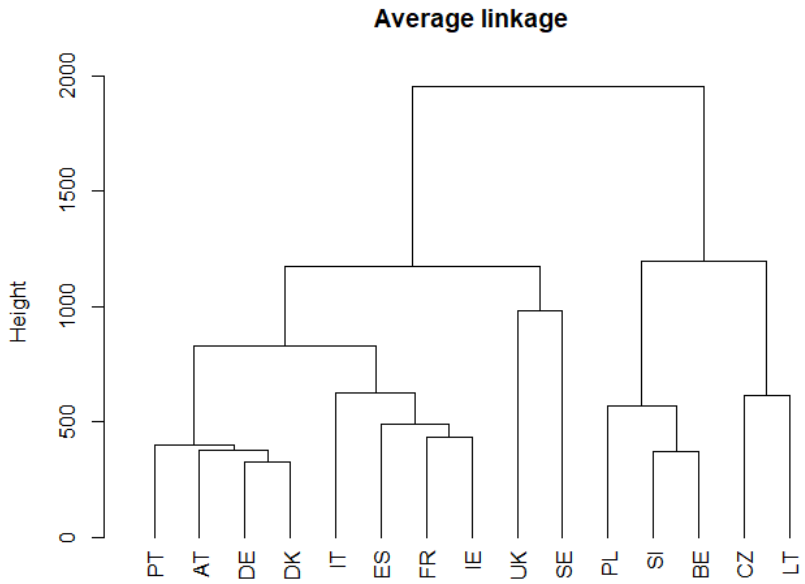


Figure 4.13: Dendrogram of the raw dataset, euclidean distance and average linkage

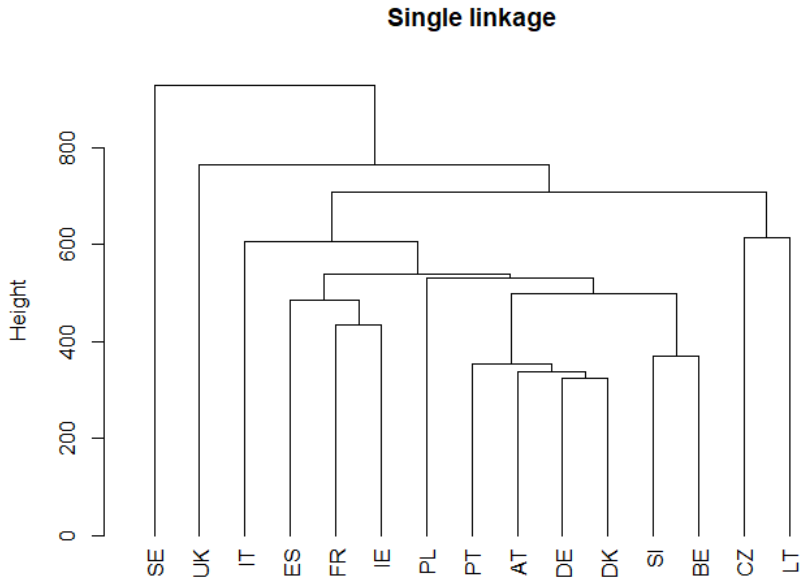


Figure 4.14: Dendrogram of the raw dataset, euclidean distance and single linkage

Having a first look at the figures (4.11, 4.12, 4.13, 4.14), we observe that the ward and complete methods seem to create more accurate clusters of the European Union

countries, while the other two do not work so well. A way to measure the amount of clustering structure found is the agglomerative coefficient, using the agnes function. This coefficient takes values from 0 to 1, with values closer to 1 indicate a strong clustering structure. As we computed it for the four different agglomerative methods assessed (table 4.3), ward's method seems to identify the strongest clustering structure.

Table 4.3: Agglomerative coefficient results for the four linkage criteria (raw data, euclidean distance)

Average	Single	Complete	Ward
0.7299530	0.4619372	0.8516763	0.8785674

Next, we present the corresponding graphs obtained using the DTW distance in the algorithm. Figure (4.15) depicts the similar heatmap representation of the dissimilarity matrix (also indicated in Appendix A), while figures (4.16, 4.17, 4.18, 4.19) are the four diagrams derived from the same four different linkage criteria. Having calculated once again the agglomerative coefficients (see table 4.4), the ward method seems to be the most fitting again, while complete method exported almost the same results. Single linkage criterion is once again the least appropriate for our data set.

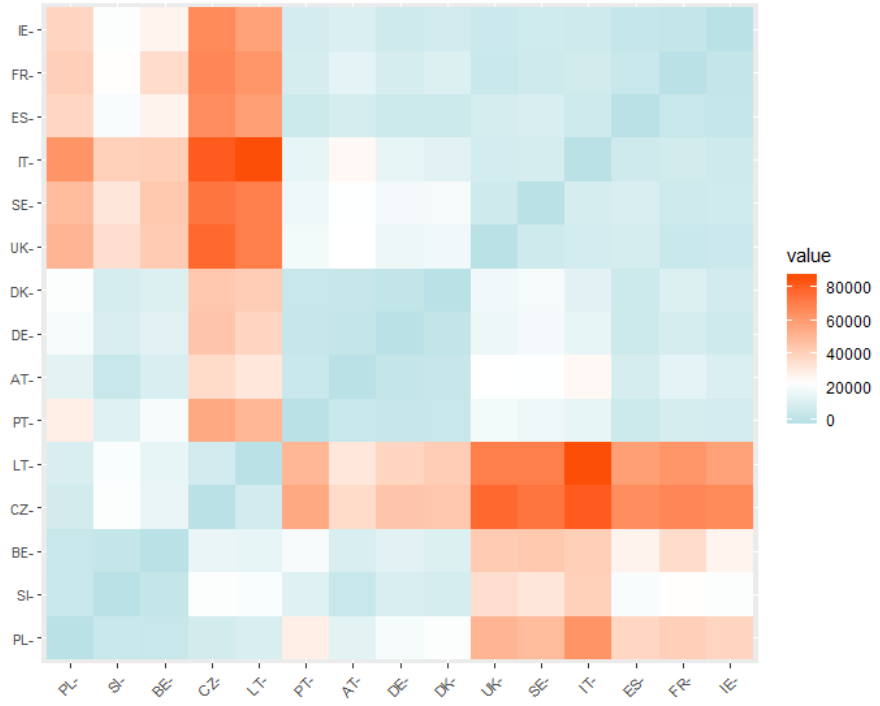


Figure 4.15: Heatmap based on the DTW dissimilarity matrix, raw data

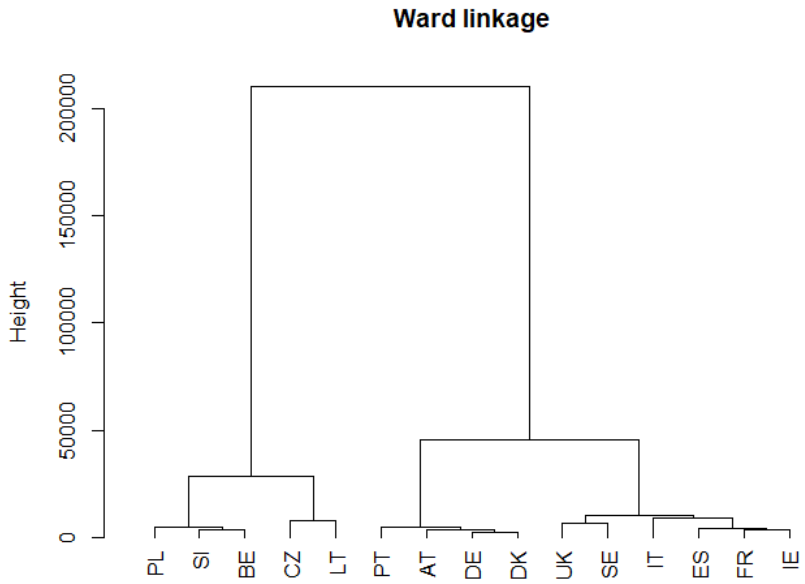


Figure 4.16: Dendrogram of the raw dataset, DTW distance and ward linkage

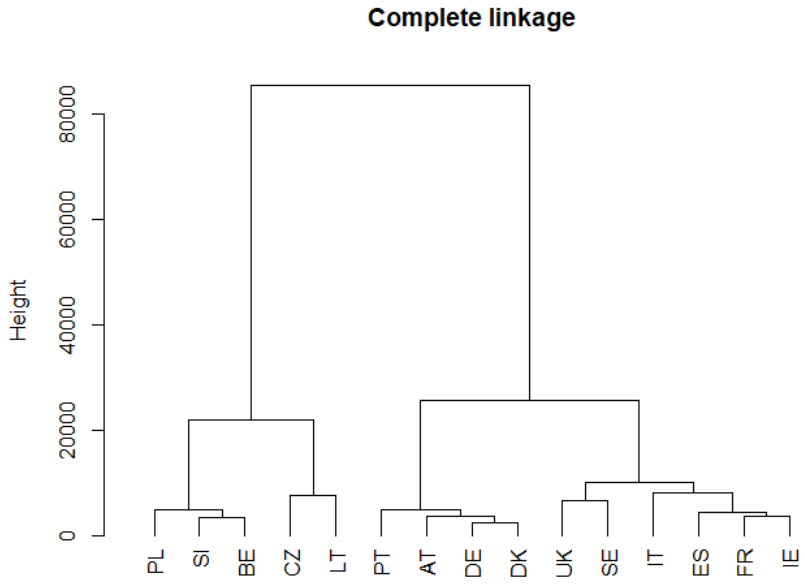


Figure 4.17: Dendrogram of the raw dataset, DTW distance and complete linkage

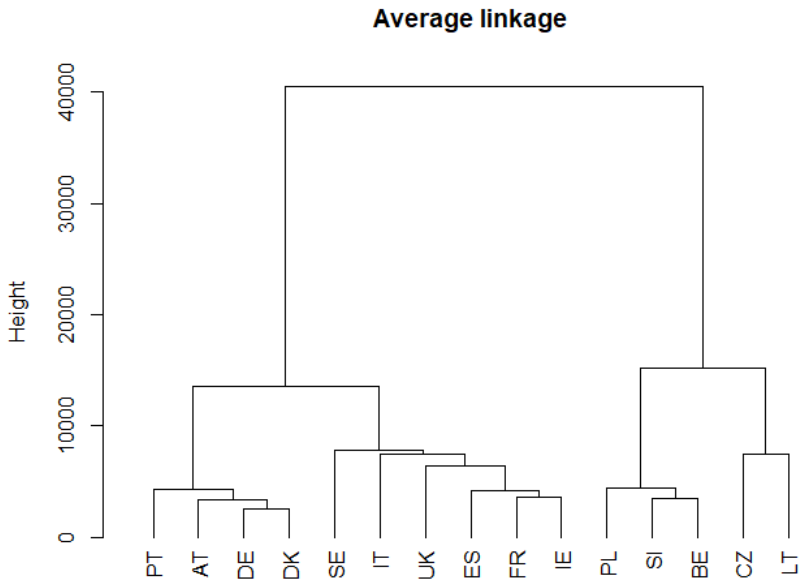


Figure 4.18: Dendrogram of the raw dataset, DTW distance and average linkage

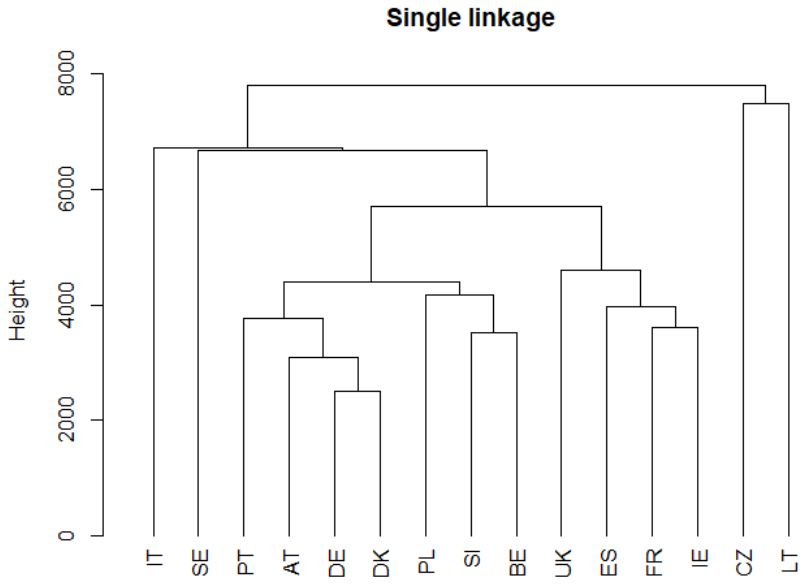


Figure 4.19: Dendrogram of the raw dataset, DTW distance and single linkage

Table 4.4: Agglomerative coefficient results for the four linkage criteria (raw data, DTW distance)

Average	Single	Complete	Ward
0.8811299	0.4261642	0.9427119	0.9571770

From the preceding estimates, it becomes clear that for the both different distance measures, the ward method seems as partly expected to be the most suitable to proceed further in our calculations. Once we have extracted the trees and selected those that apply to us (4.11, 4.16), the next step is to calculate the appropriate number of clusters. The height of the cut to the dendrogram controls the number of clusters obtained. Determining the optimal number of clusters in a data set is a fundamental issue that requires us to specify the number of clusters (k) to be generated. The following figures (4.20, 4.21, 4.22) show the results provided by the elbow, silhouette, and gap statistic methods as explained in the methodology. As we observe, there is no definitively clear optimal number of clusters in this case; the first suggests $k = 4$, the second $k = 3$ and the last $k = 1$. We continued our analysis for $k = 4$ as proposed by the 1st method, which is the most popular among the three and for $k = 3$ as suggested by the silhouette method. We suspected according to the trees and the heatmaps above that one of the two is the right choice but we performed a complete diagrammatic presentation for both.

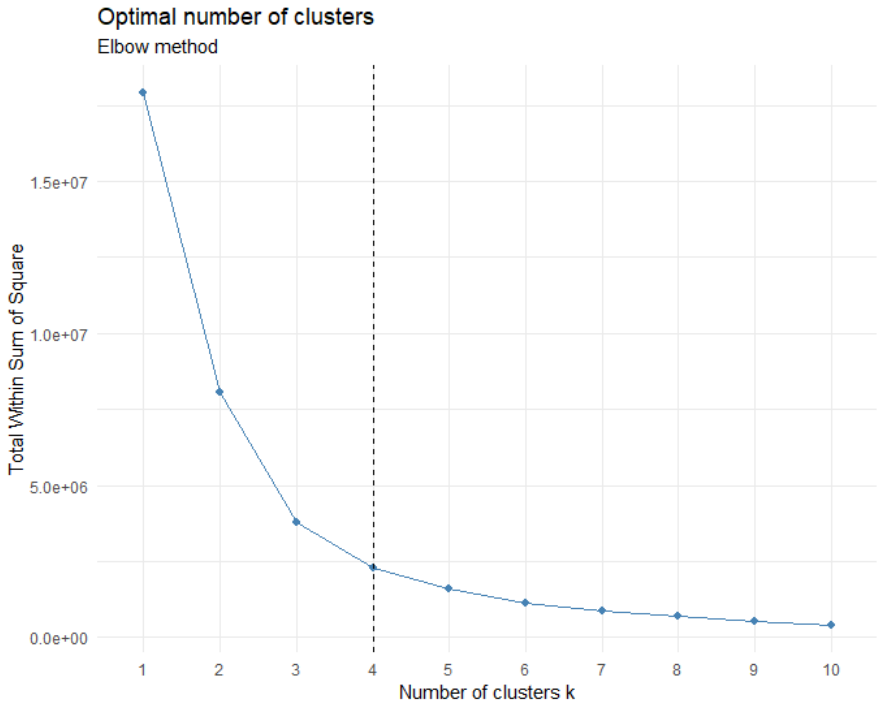


Figure 4.20: Elbow method results based on the raw dataset

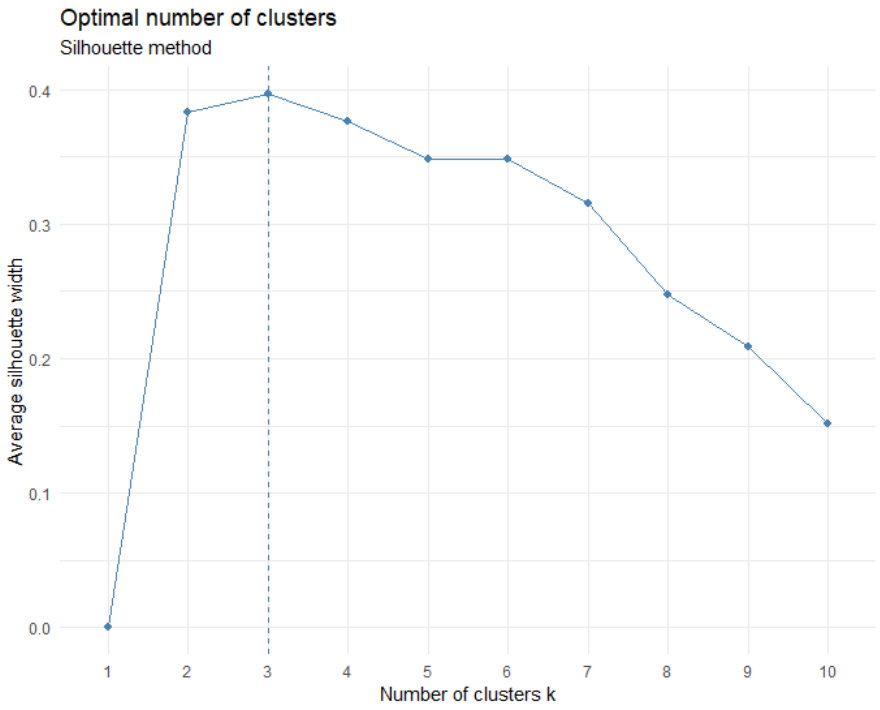


Figure 4.21: Silhouette method results based on the raw dataset

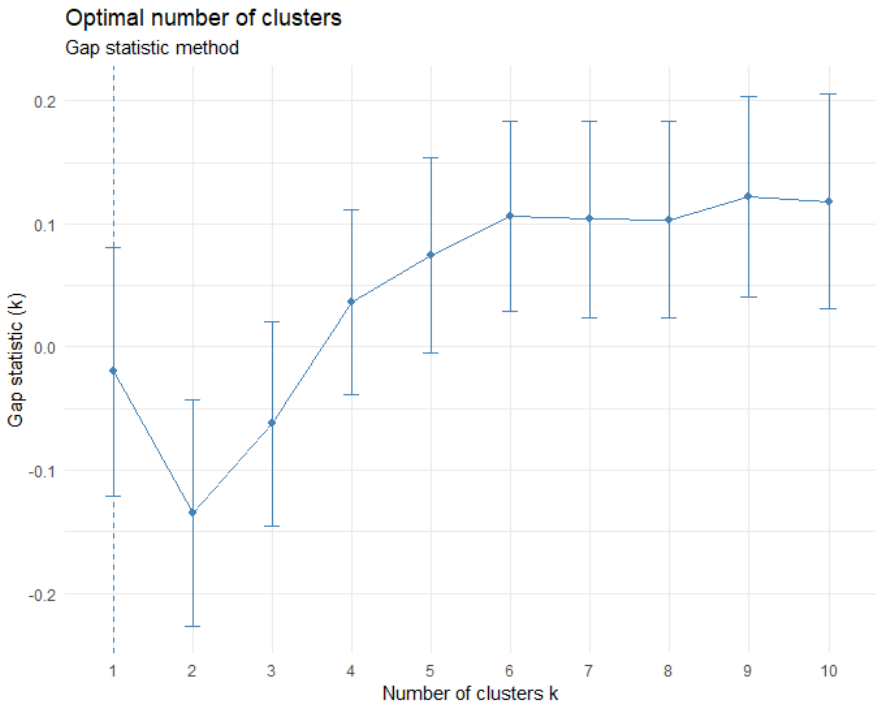


Figure 4.22: Gap stat method results based on the raw dataset

Summarizing our results so far, the ward method turned out to be the most suitable for use in the algorithm. Figure (4.23) compares side by side hierarchical

clustering with the ward's linkage and the euclidean distance versus hierarchical clustering with the ward's linkage and the DTW distance, with their labels connected by lines. The quality of the alignment of the two trees can be measured using the function entanglement. Entanglement is a measure between 1 (full entanglement) and 0 (no entanglement). A lower entanglement coefficient such as 0.13 in our case corresponds to very good alignment.

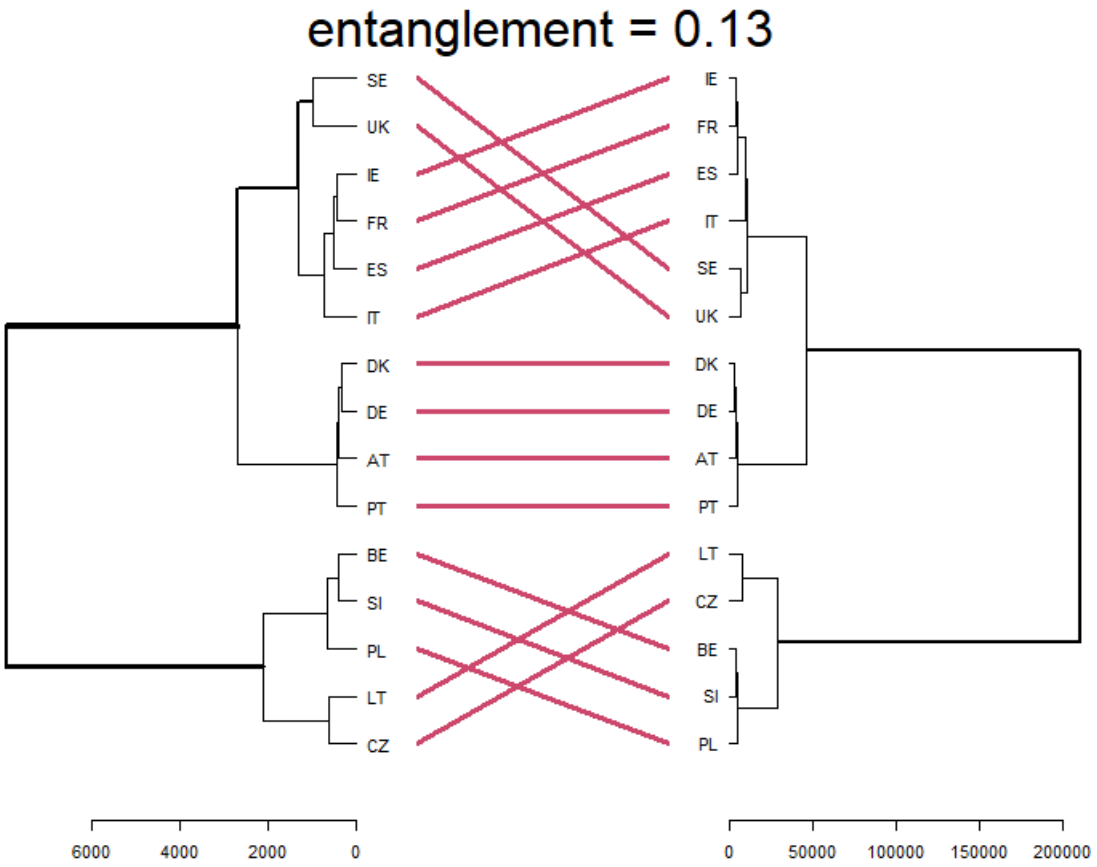
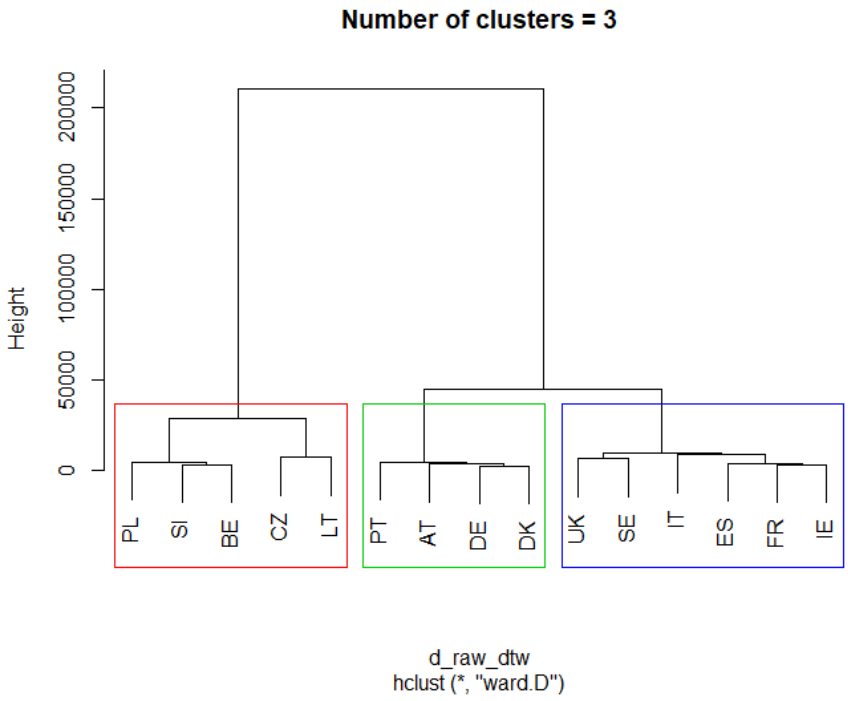
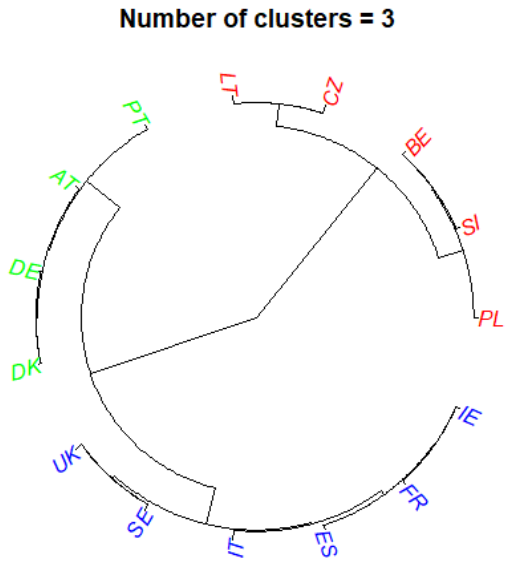


Figure 4.23: Side by side comparison between hierarchical clustering with the ward's linkage and the euclidean distance, raw data (left) versus hierarchical clustering with the ward's linkage and the DTW distance, raw data (right)

Finally, concerning the agglomerative and entanglement coefficients as well as the dendrograms' results, we present the final trees based on the DTW distance measure for $k = 3, 4$ (figures 4.24, 4.25) which seems to be the most appropriate in this raw dataset. Both distances seem to work very well in our case, but DTW stands out a bit more since the euclidean is a special case of the first. The corresponding dendrograms for the euclidean distance are cited in Appendix A.

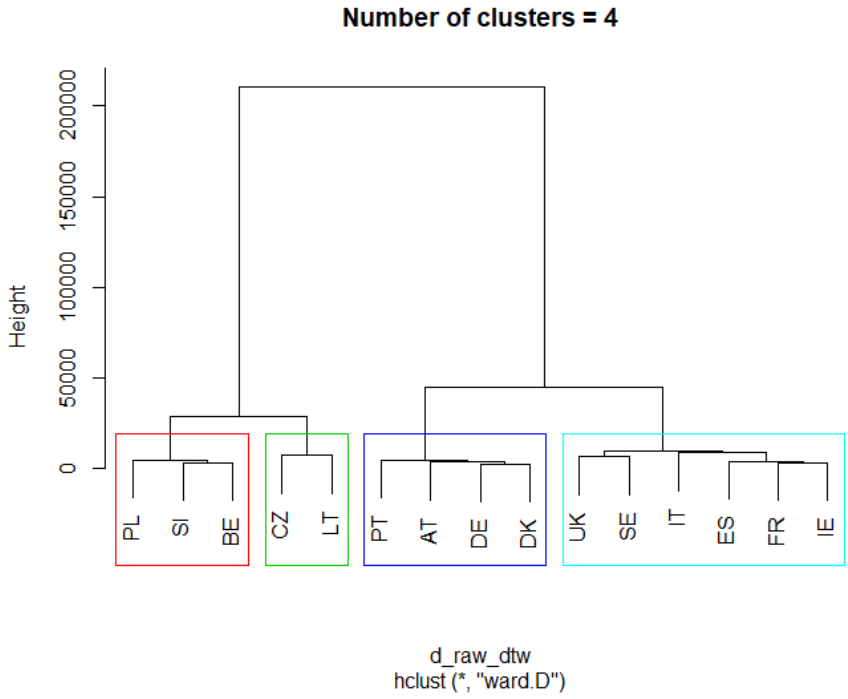


(a) Customization

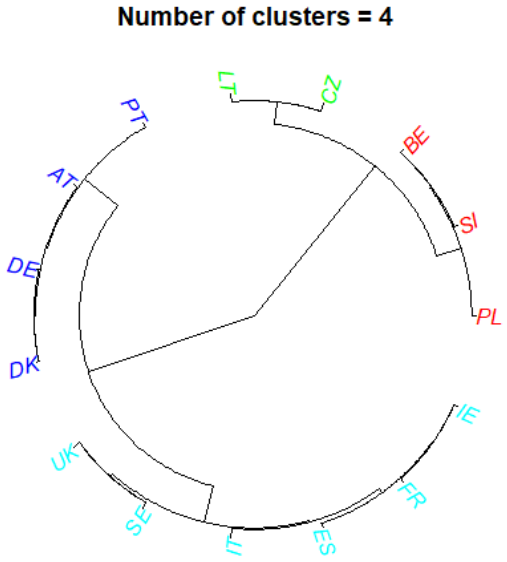


(b) Customization

Figure 4.24: Final hierarchical clustering for $k = 3$ based on DTW distance and ward linkage method, raw data



(a) Customization



(b) Customization

Figure 4.25: Final hierarchical clustering for $k = 4$ based on DTW distance and ward linkage method, raw data

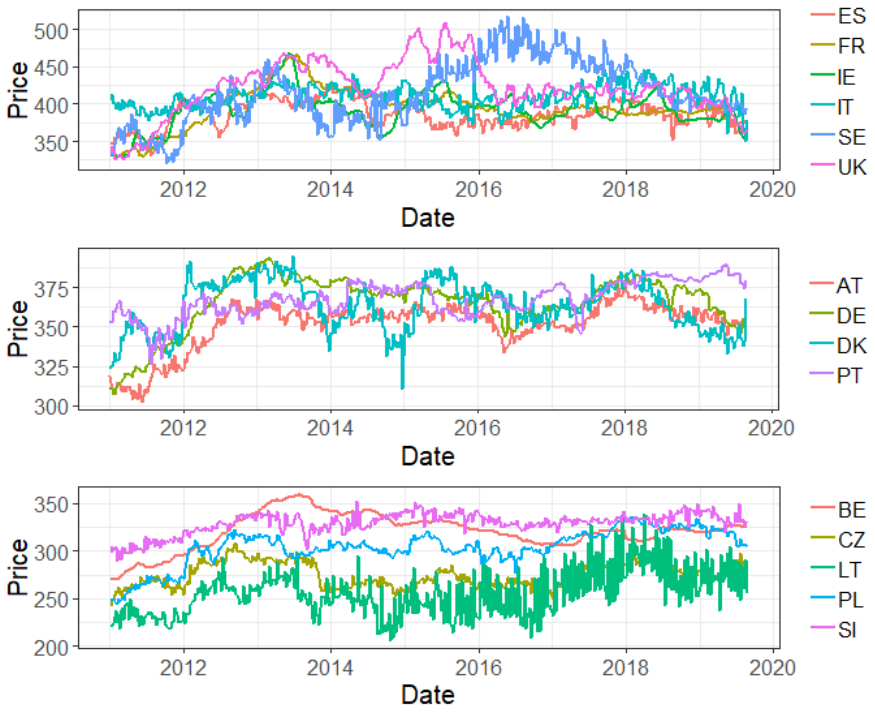


Figure 4.26: Line graphs grouped in their own clusters, $k = 3$

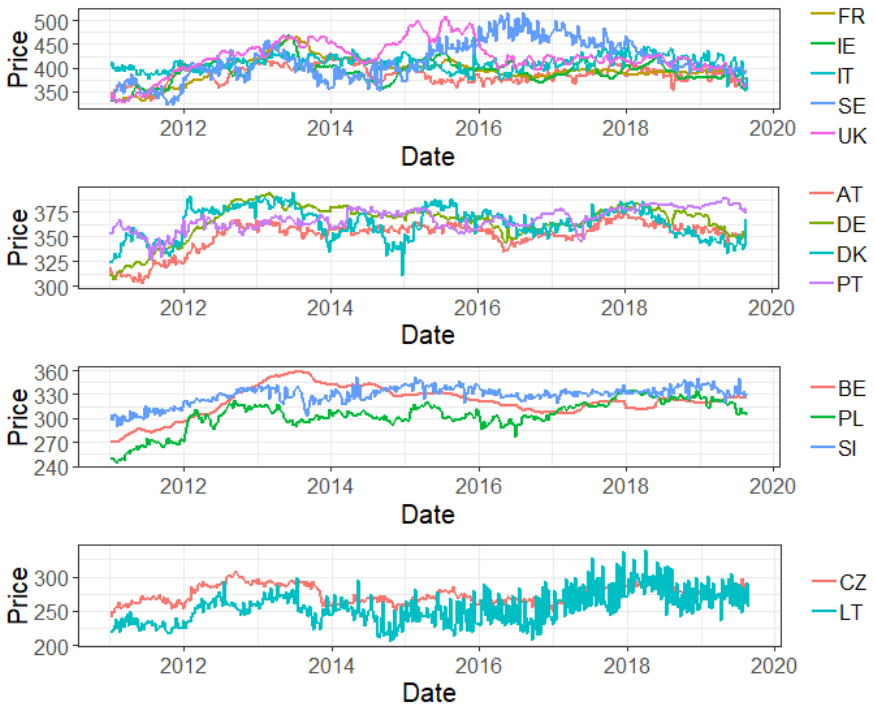


Figure 4.27: Line graphs grouped in their own clusters, $k = 4$

Figures (4.26, 4.27) show how the time series are categorized after the definition of clusters. Considering $k = 3$, the largest cluster (blue) contains 6 countries namely

{ United Kingdom(UK), Sweden (SE), Italy (IT), Espania (ES), France (FR) and Ireland (IE) }. Apart from Sweden, which is somewhere in the middle regarding beef production, the rest of the countries are some of those having a leading role in the European market in economic sectors such as production, consumption and intra-trade. Besides, figure (4.26) shows that this cluster contains the countries with the highest weekly average carcass prices. The central group (green) consists of 4 countries namely { Portugal (PT), Austria (AU), Germany (DE) and Denmark (DK) }. Germany, as we see, has a central position in the tree which is expected as it is one of the largest producers and consumers in the EU and also plays a significant role in international trade. The other two countries in the group, despite their smaller size, appear to be directly linked to Germany since they also have a significant share in the beef production against many other countries in the Union, but also maybe because of their geographical connection. Together with Portugal, their price index seems to be somewhere in the middle and smaller than the cluster's that includes the big markets (blue). The last cluster contains the last five countries namely { Poland (PL), Slovenia (SI), Belgium (BE), Czechia (CZ) and Lithuania (LT) }. Poland is one of the seven largest producers but seems to have the lowest price index among them. Belgium is also an important market which seems to have differences concerning its neighboring and leading markets in Europe, while the other three are countries with less or insignificant production. The only difference we have when cutting the tree to a height which creates 4 clusters is the creation of a fourth cluster that separates Czechia and Lithuania from the previous group to which they used to belong.

4.2.2 Scaled data scenario

The second set of data that was studied in the same way, relates to the first used above in the analysis but applying scaling to it. Remember once again that scaling/standardization means each variable has a mean zero and standard deviation one. This is done to avoid the clustering algorithm to depend on an arbitrary variable unit. Our goal anew is to come up with a tree diagram with a specific distance measure and a specific linkage criterion and ultimately export the number of clusters that best categorize the countries. Following the same line of reasoning, starting with the euclidean distance, we extracted the four diagrams for the four different linkage criteria (see 4.29, 4.30, 4.31, 4.32), which are accompanied by the representation of the dissimilarity matrix in the form of a heatmap (4.28) and the calculation of the agglomerative coefficient (table 4.5). The diagrammatic representation of the trees combined with this coefficient again suggests the ward method as the ideal link criterion for our scaled dataset.

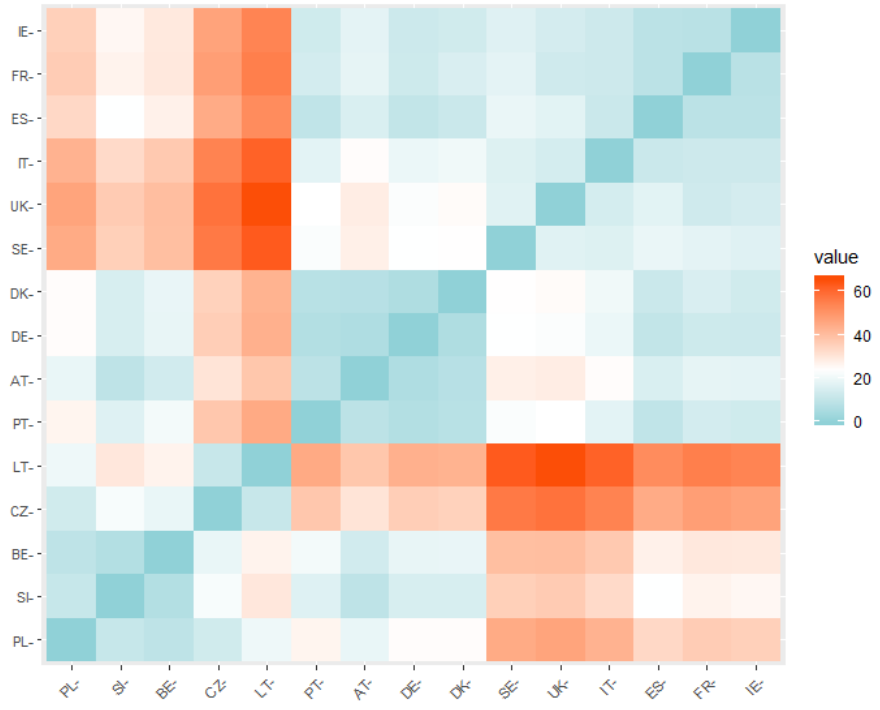


Figure 4.28: Heatmap based on the euclidean dissimilarity matrix, scaled data

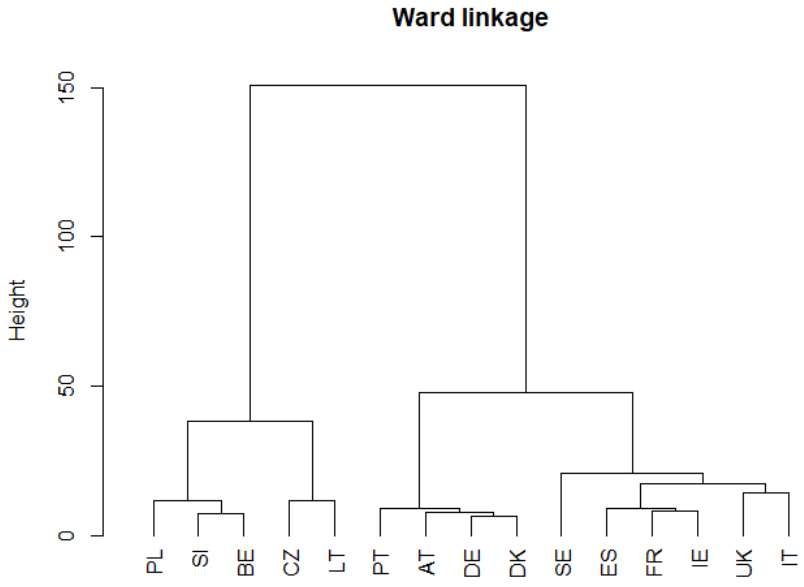


Figure 4.29: Dendrogram of the scaled dataset, euclidean distance and ward linkage

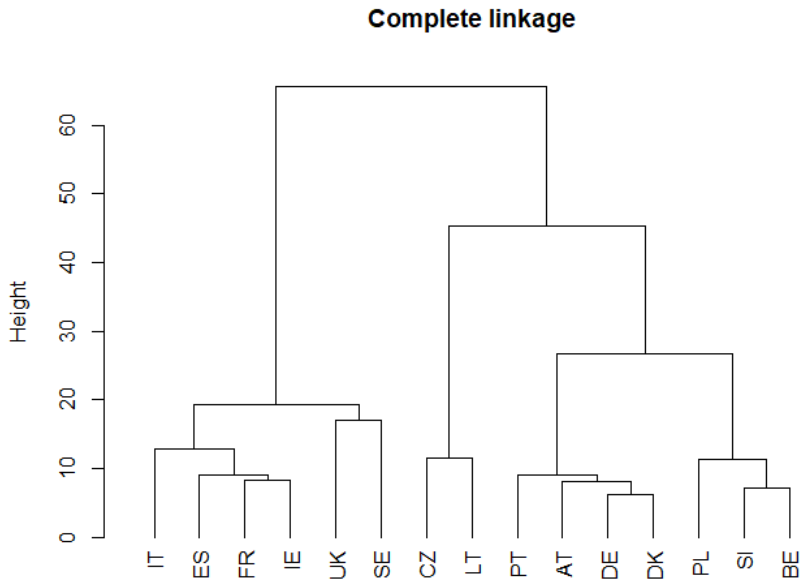


Figure 4.30: Dendrogram of the scaled dataset, euclidean distance and complete linkage

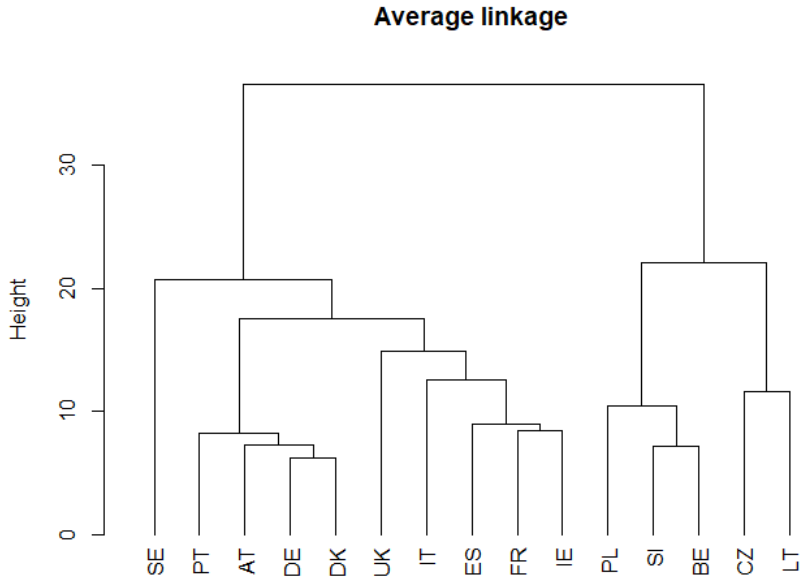


Figure 4.31: Dendrogram of the scaled dataset, euclidean distance and average linkage

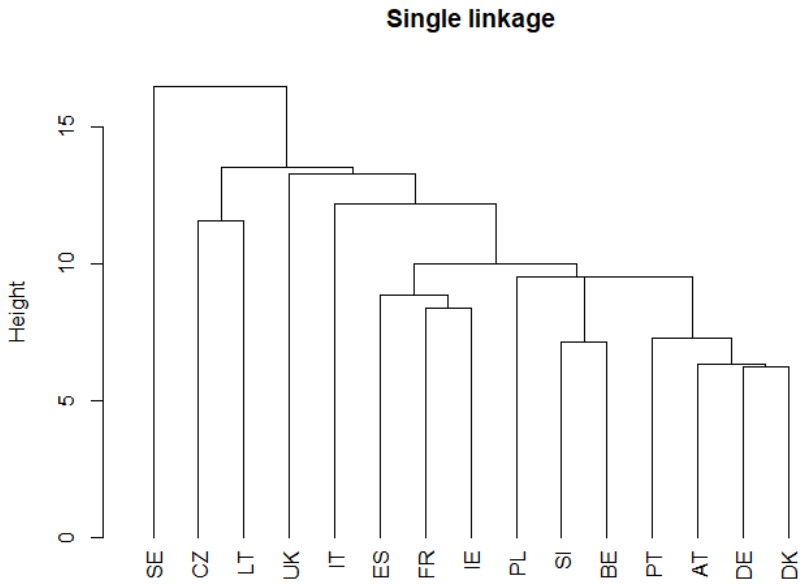


Figure 4.32: Dendrogram of the scaled dataset, euclidean distance and single linkage

Table 4.5: Agglomerative coefficient results for the four linkage criteria (scaled data, euclidean distance)

Average	Single	Complete	Ward
0.7266208	0.4310072	0.8462735	0.8874635

Following are the same graphs that correspond to using DTW as the distance measure. The ward method seems to be the ideal one to use and even has the highest agglomerative coefficient (4.6) we have encountered so far. On the contrary, the use of single method seems to cluster the observations in a weaker way, which is to be expected according to its disadvantage of creating many successive clusters.

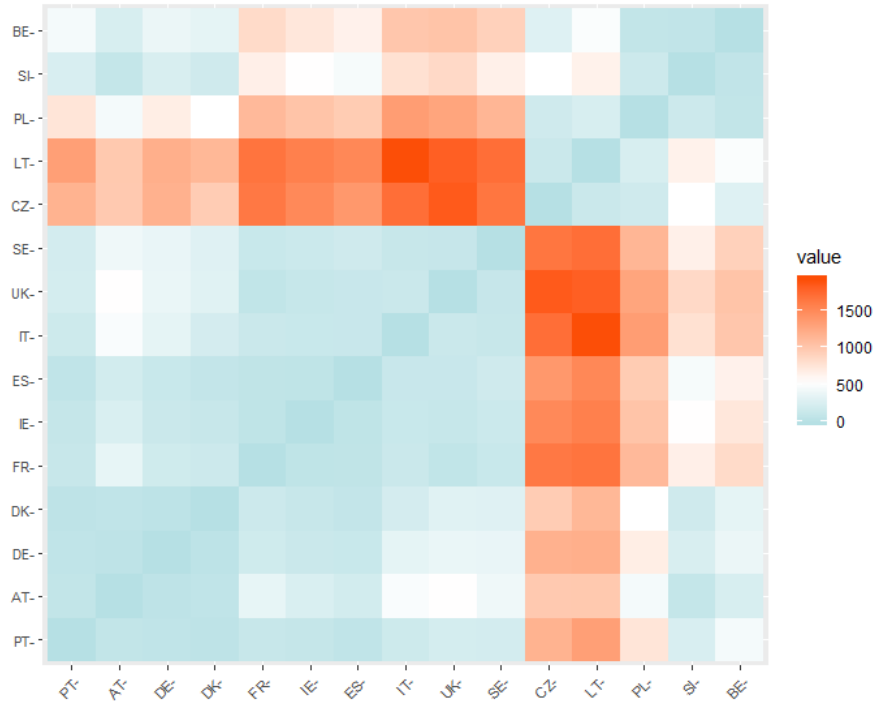


Figure 4.33: Heatmap based on the DTW dissimilarity matrix, scaled data

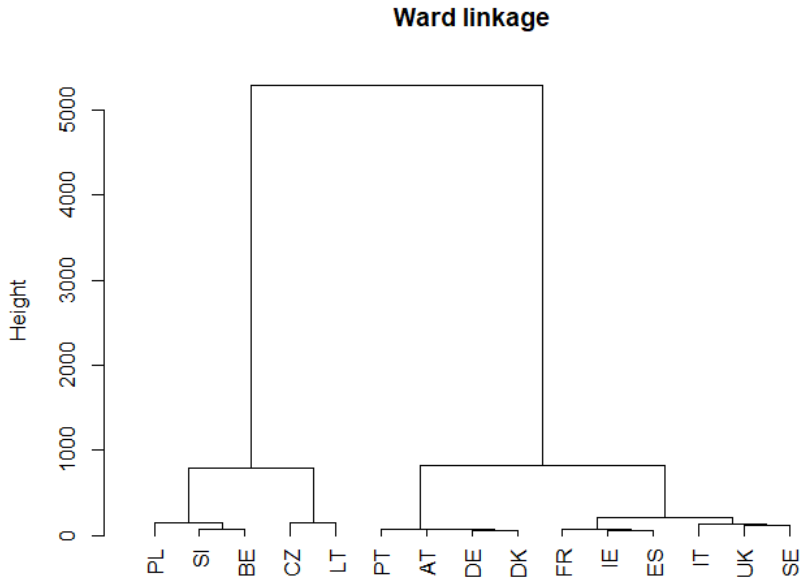


Figure 4.34: Dendrogram of the scaled dataset, DTW distance and ward linkage

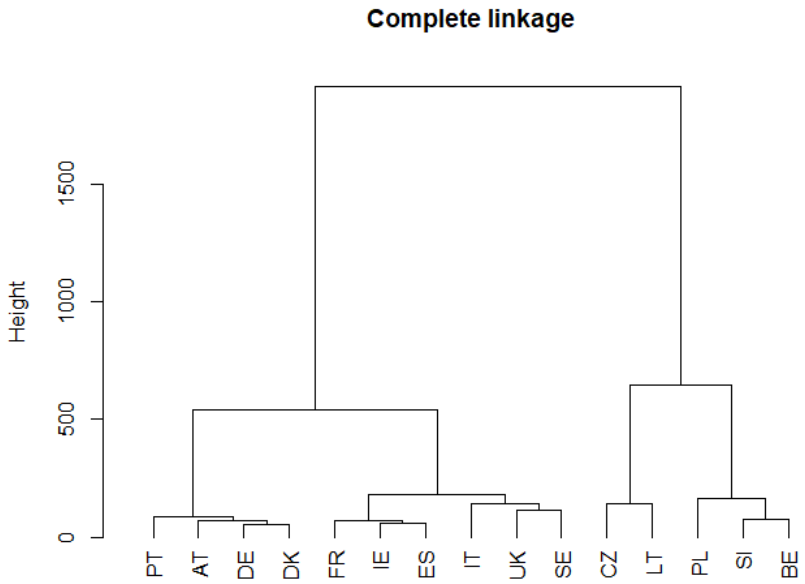


Figure 4.35: Dendrogram of the scaled dataset, DTW distance and complete linkage

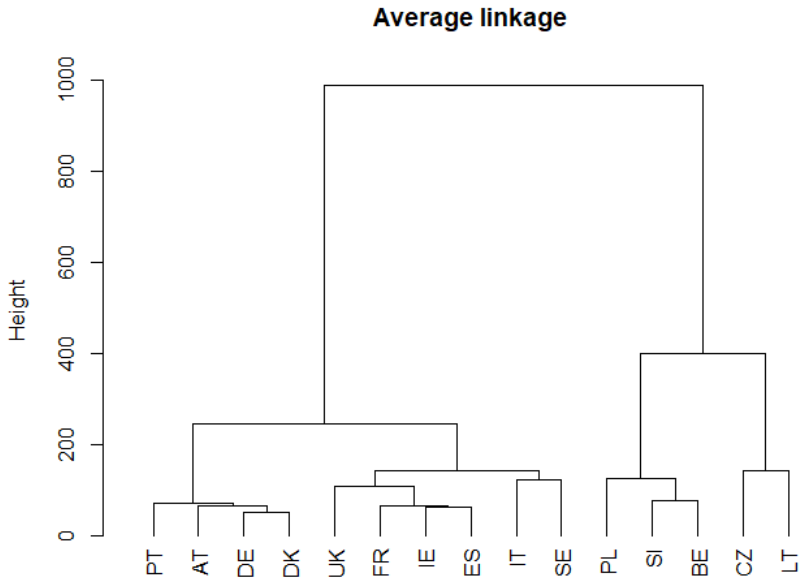


Figure 4.36: Dendrogram of the scaled dataset, DTW distance and average linkage

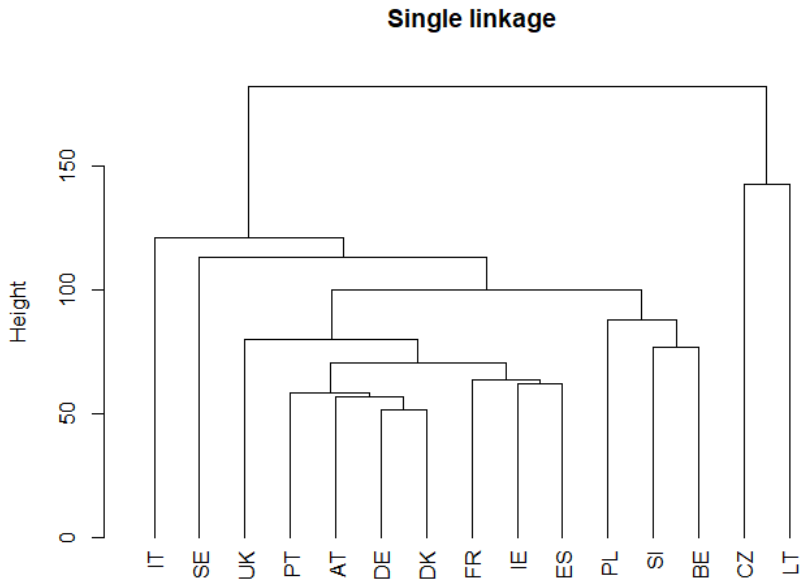


Figure 4.37: Dendrogram of the scaled dataset, DTW distance and single linkage

Table 4.6: Agglomerative coefficient results for the four linkage criteria (scaled data, DTW distance)

Average	Single	Complete	Ward
0.9094206	0.5437331	0.9503966	0.9664953

Applying the elbow, silhouette and gap stat methods, we note that for the first, the suggested number of clusters is 4, for the second 2 and for the last 1. The only difference concerning the previous set (raw) lies in the fact that the silhouette method proposes minus one number of clusters. When comparing the two dendrograms for the two different distance measures in terms of the ward linkage method, the entanglement coefficient and the graph visualization indicate that both measures are quite reliable to be used and suggest very similar clusters. If we take into account figure (4.34) including the DTW distance and compare it to the tree we ended up in the raw set (figure 4.16), we also see that both extract related results. Since we have again ambiguous results concerning the appropriate number of clusters, we present further below the final trees based on the DTW distance measure and the ward method for $k = 2, 4$. The two resulting trees conclude in almost the same structures as those we analyzed earlier in the raw dataset. If we consider the number of clusters to be 2, the first (red) consists of { Poland, Slovenia, Belgium, Chechia and Lithuania }, while the second (green) includes { Portugal, Austria, Germany, Denmark, France, Ireland, Espania, Italy, United Kingdom and Sweden

}. In essence, the two left clusters for the previous tree in the raw data for $k = 4$, merge into one as also the two on the right. For $k = 4$, their clusters are the same as we ended up with for the same number in the raw dataset, with the only difference being that there is simply a rearrangement of places in the right bigger cluster. In conclusion, both datasets yield similar results, with the latter slightly improving the predictive accuracy, due to its purpose of not allowing a particular feature to be affected by a large numerical value range.

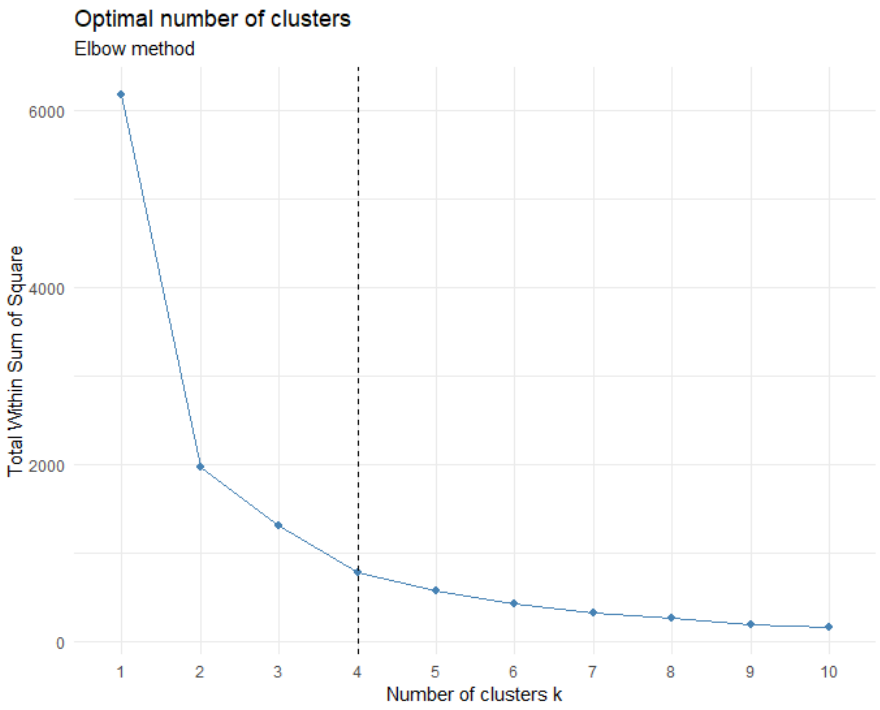


Figure 4.38: Elbow method results based on the scaled dataset

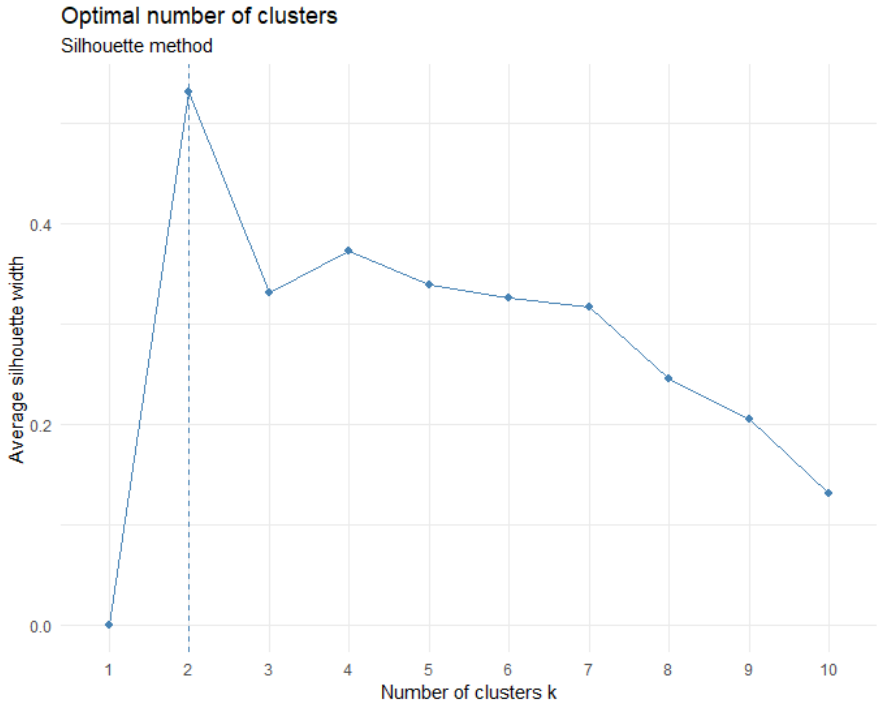


Figure 4.39: Silhouette method results based on the scaled dataset

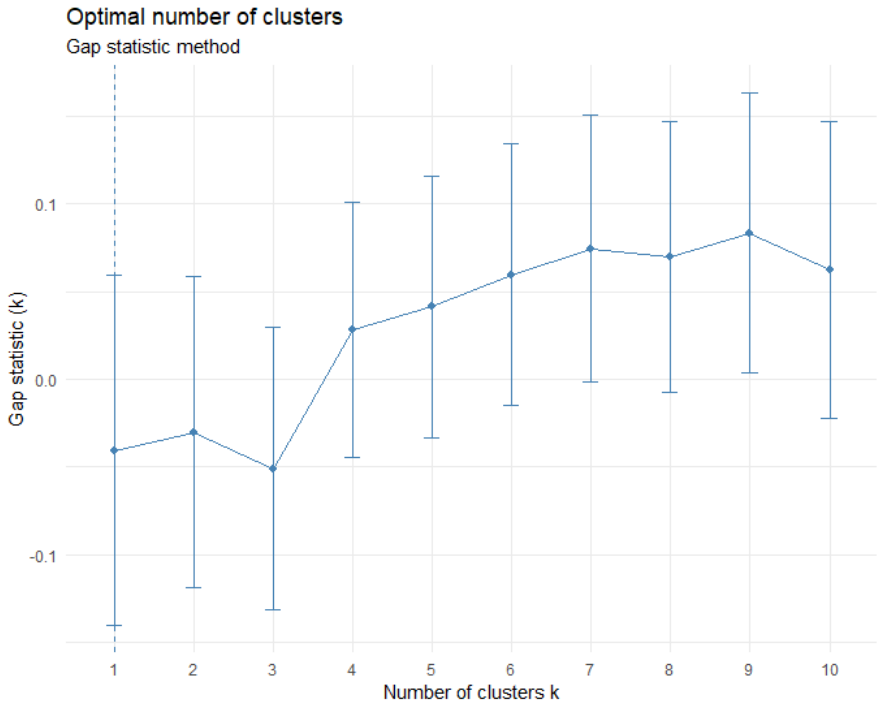


Figure 4.40: Gap stat method results based on the scaled dataset

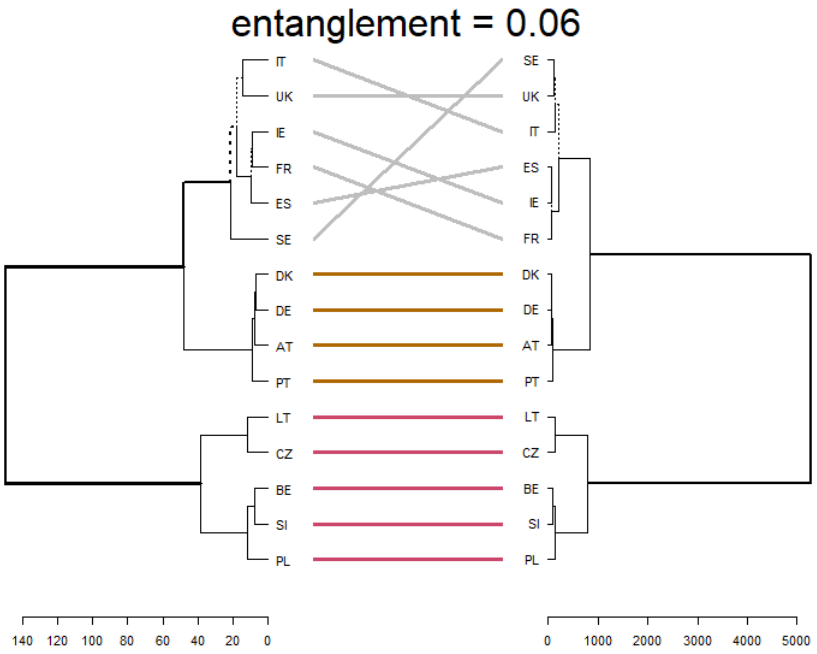


Figure 4.41: Side by side comparison between hierarchical clustering with the ward's linkage and the euclidean distance, scaled data (left) versus hierarchical clustering with the ward's linkage and the DTW distance, scaled data (right)

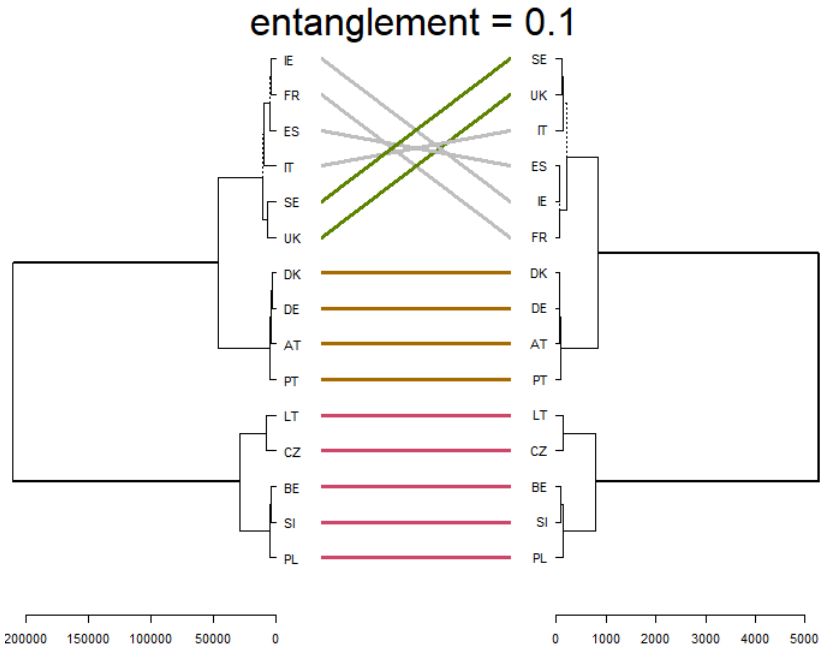
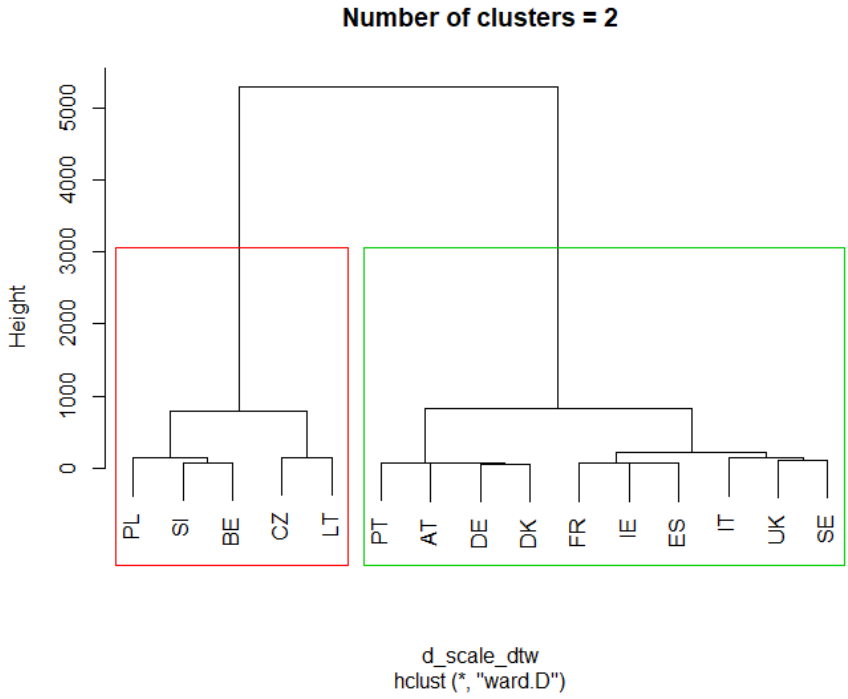
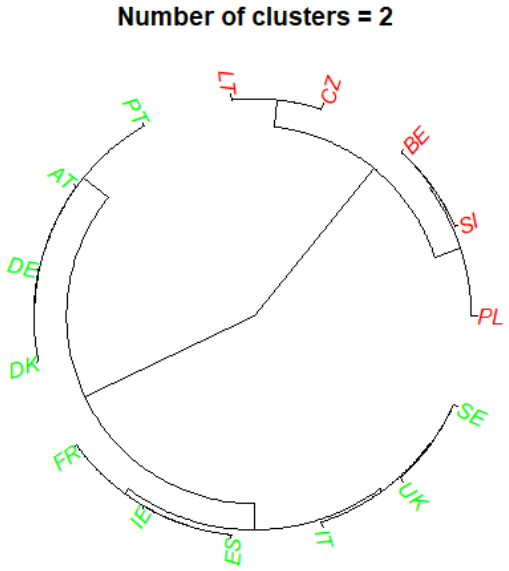


Figure 4.42: Side by side comparison between hierarchical clustering with the ward's linkage and the DTW distance, raw data (left) versus hierarchical clustering with the ward's linkage and the DTW distance, scaled data (right)

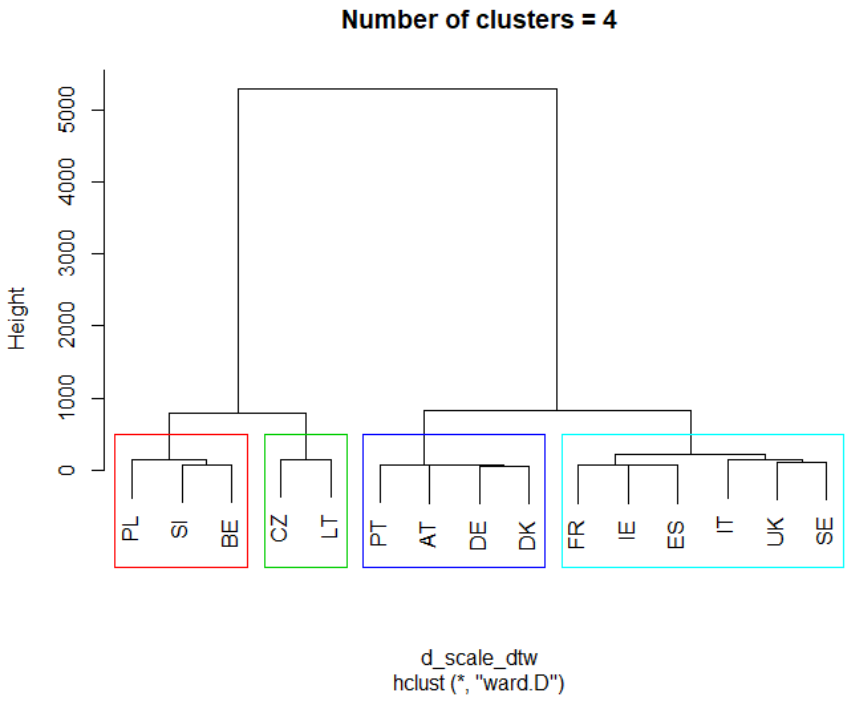


(a) Customization 1

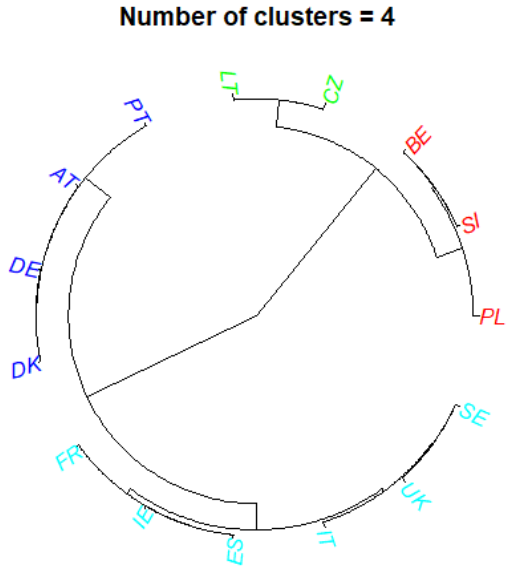


(b) Customization 2

Figure 4.43: Final hierarchical clustering for $k = 2$ based on DTW distance and ward linkage method, scaled data



(a) Customization 1



(b) Customization 2

Figure 4.44: Final hierarchical clustering for $k = 4$ based on DTW distance and ward linkage method, scaled data

4.2.3 Log-returns data scenario

The last hierarchical analysis was applied to the dataset with the logarithmic returns we created and discussed in the descriptive statistics. The previous two have returned similar results and we wanted to see if any of them are confirmed here as well. Anyway, this study is all about a general presentation, testing many cases to come to some conclusions. Observing the heatmap displaying the diagrammatic representation of the Euclidean dissimilarity matrix, we can already picture some first conclusions. More specifically, we see that Lithuania is a cluster of its own since it differs completely from all other countries. Also, we note that Sweden and Italy also have longer distances from the rest countries, whose distances among them are close to zero. Figures (4.46, 4.47, 4.48, 4.49) present the four trees for the euclidean distance concerning the four different linkage methods. In this case, all the methods seem to generate related results, but the ward method appears to stand out a little more like the most appropriate one. The matrix once again is cited in the Appendix A.

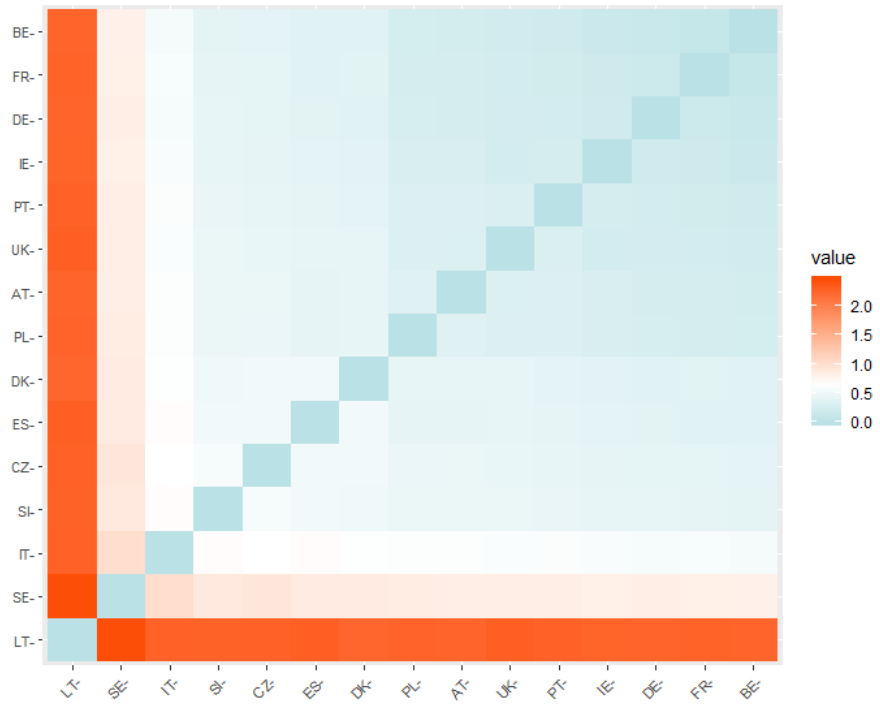


Figure 4.45: Heatmap based on the euclidean dissimilarity matrix, log-returns

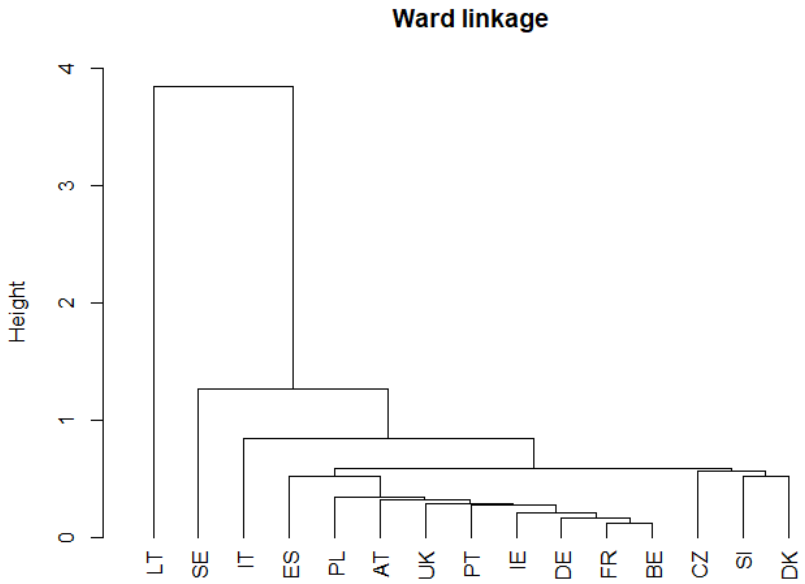


Figure 4.46: Dendrogram of the log-returns, euclidean distance and ward linkage

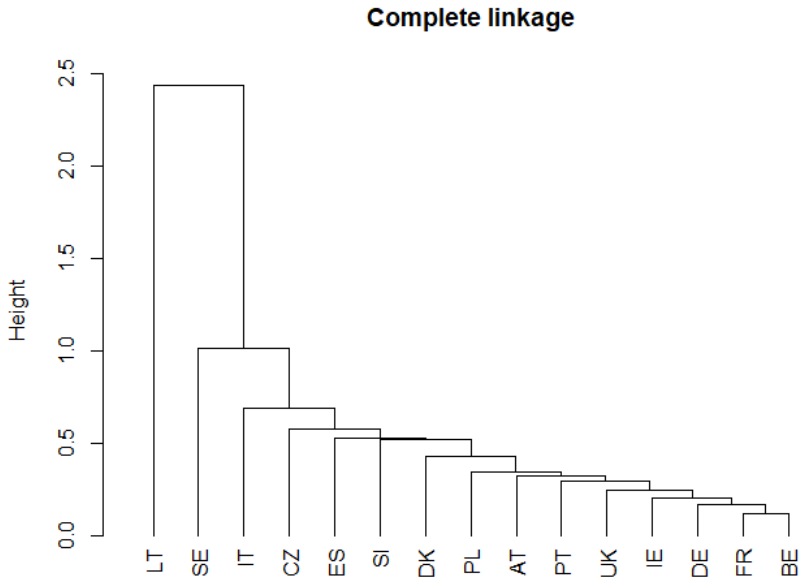


Figure 4.47: Dendrogram of the log-returns, euclidean distance and complete linkage

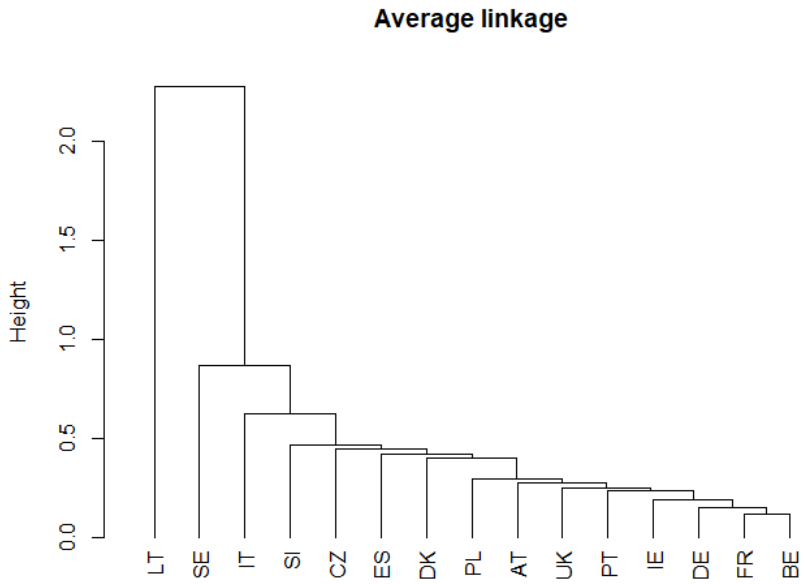


Figure 4.48: Dendrogram of the log-returns, euclidean distance and average linkage

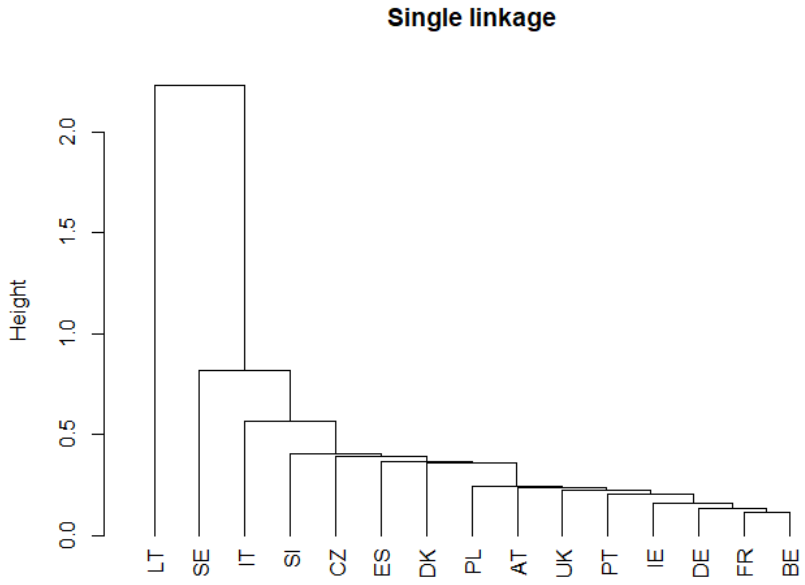


Figure 4.49: Dendrogram of the log-returns, euclidean distance and single linkage

Table 4.7: Agglomerative coefficient results for the four linkage criteria (log-returns, euclidean distance)

Average	Single	Complete	Ward
0.7914302	0.8032368	0.7809020	0.8084142

The corresponding diagrams for the DTW distance are being presented below and they suggest the same things as when using the euclidean distance. The elbow method (4.55) indicates once again the number of clusters to be 4, the silhouette method (4.56) to be 2 and the gap stat method (4.57) is the one that differs, suggesting the number to be 8. Figures (4.59, 4.60) show the final trees for $k = 2, 4$ for the DTW distance and the ward linkage criteria.

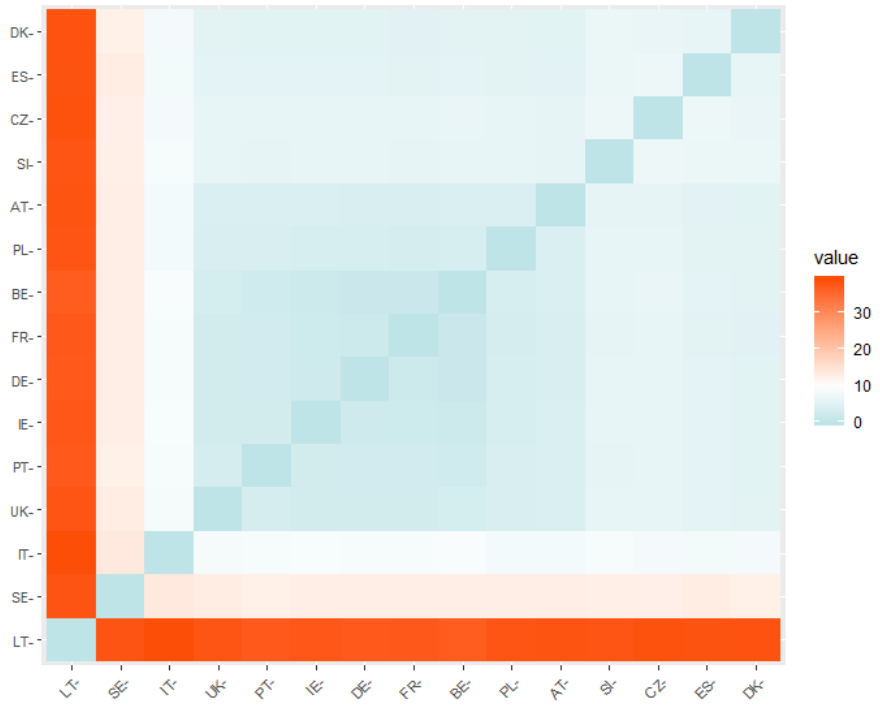


Figure 4.50: Heatmap based on the DTW dissimilarity matrix, log-returns

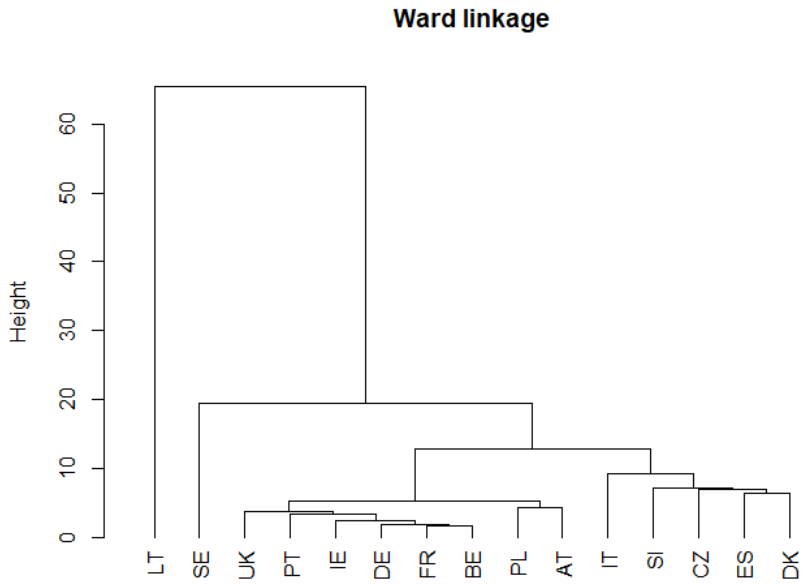


Figure 4.51: Dendrogram of the log-returns, DTW distance and ward linkage

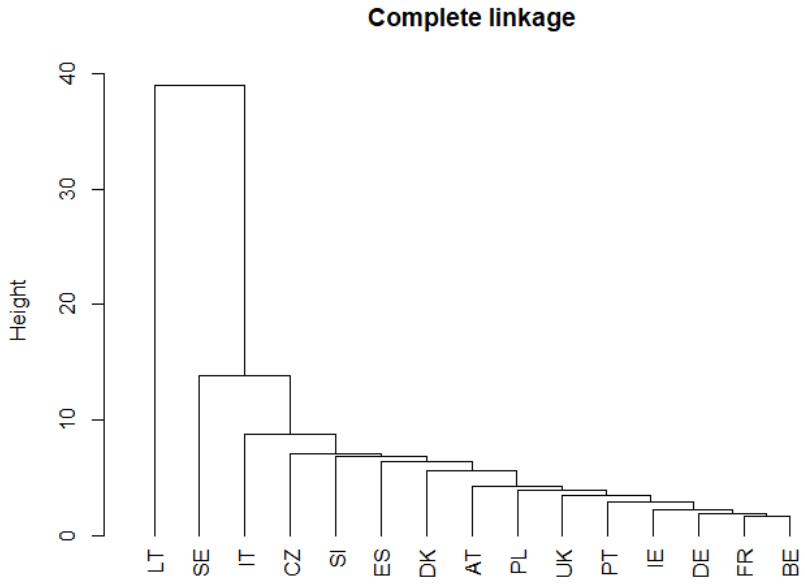


Figure 4.52: Dendrogram of the log-returns, DTW distance and complete linkage

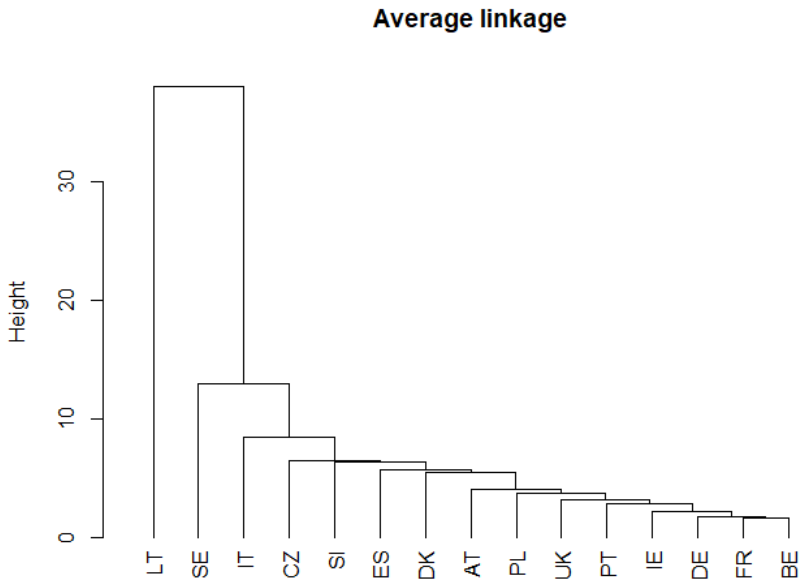


Figure 4.53: Dendrogram of the log-returns, DTW distance and average linkage

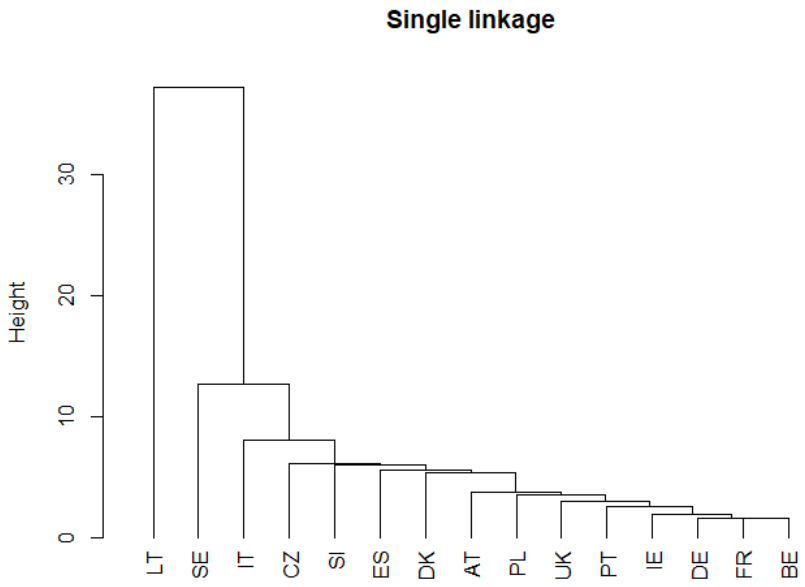


Figure 4.54: Dendrogram of the log-returns, DTW distance and single linkage

Table 4.8: Agglomerative coefficient results for the four linkage criteria (log-returns, DTW distance)

Average	Single	Complete	Ward
0.8172778	0.8198719	0.8127533	0.8342533

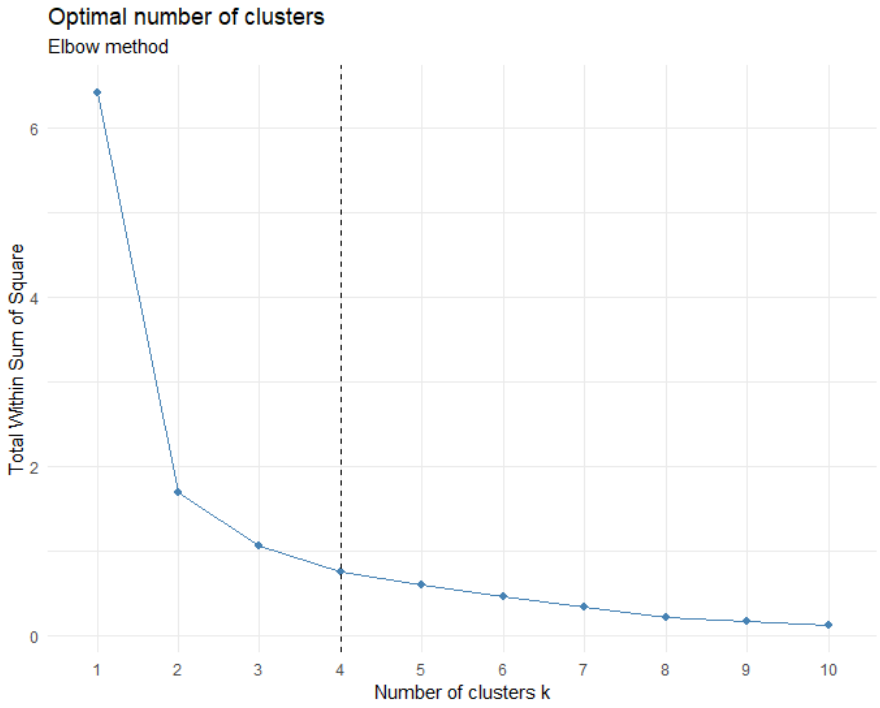


Figure 4.55: Elbow method results based on the log-returns

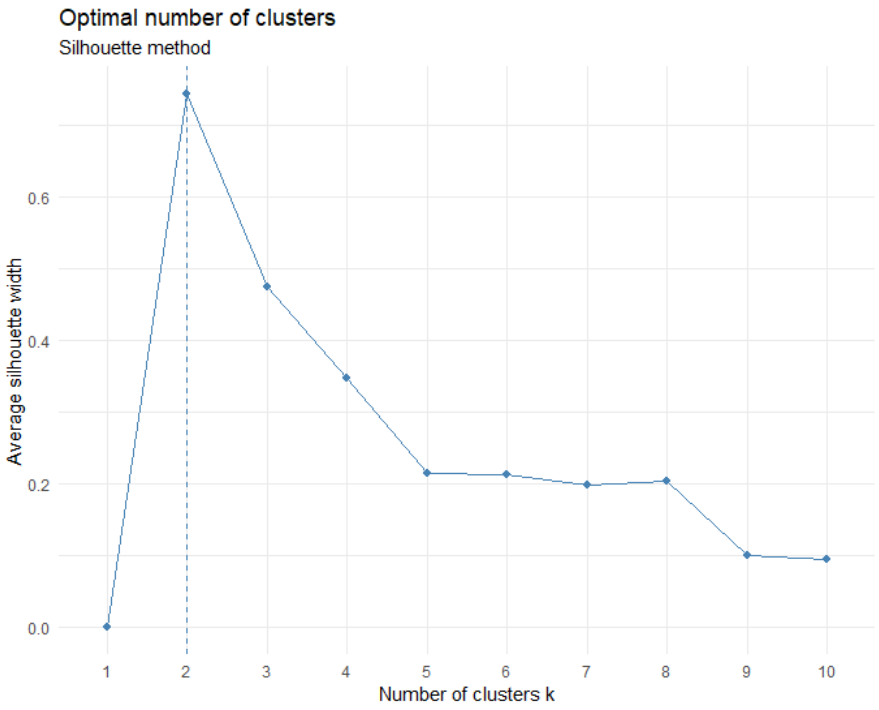


Figure 4.56: Silhouette method results based on the log-returns

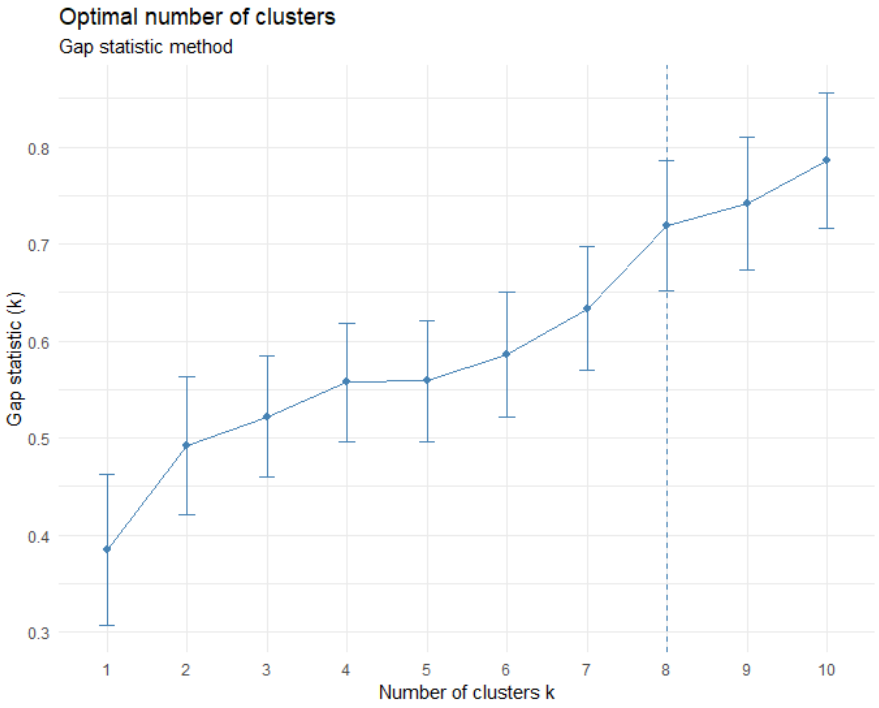


Figure 4.57: Gap stat method results based on the log-returns

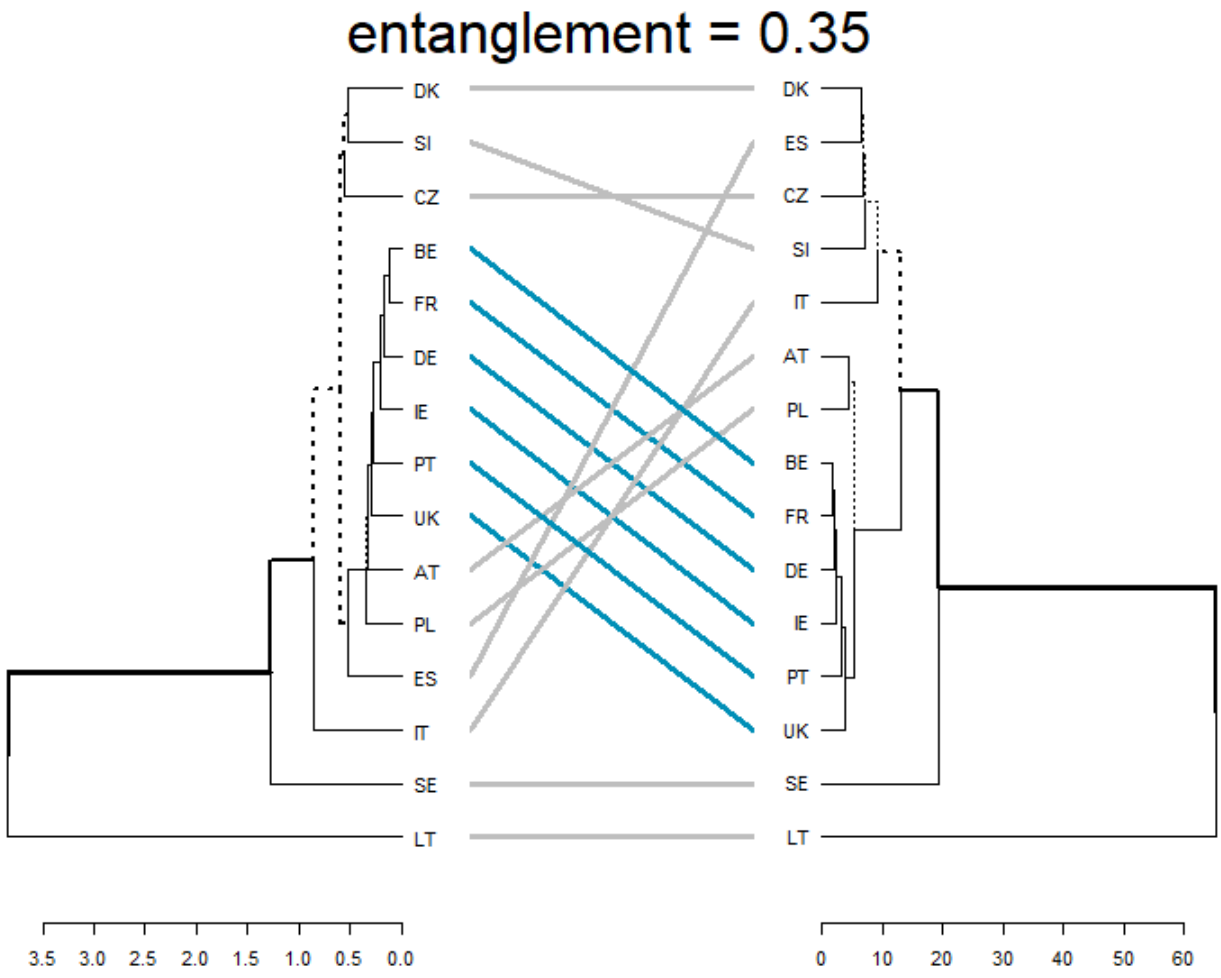
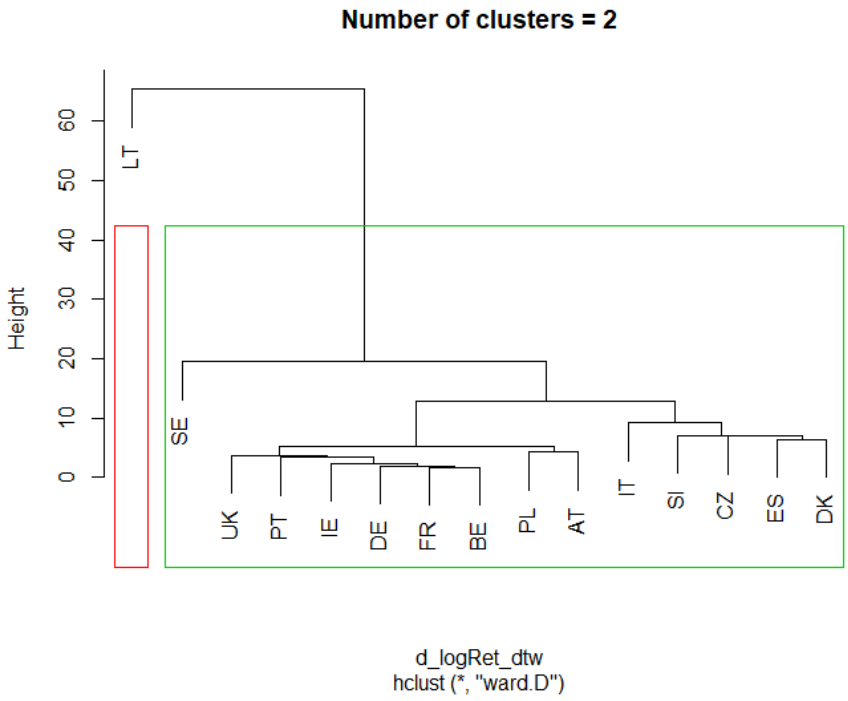
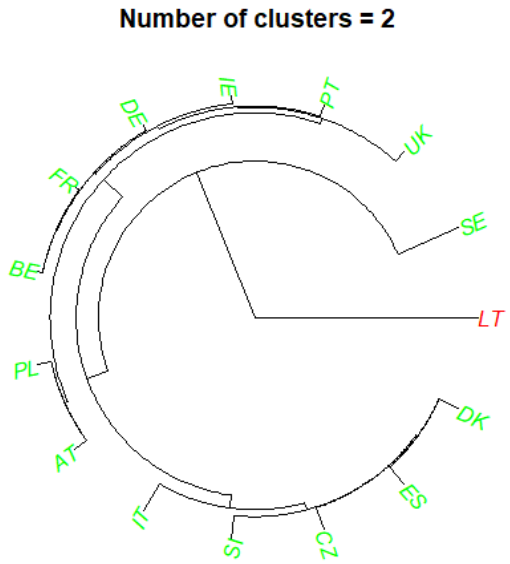


Figure 4.58: Side by side comparison between hierarchical clustering with the ward's linkage and the euclidean distance, log-returns (left) versus hierarchical clustering with the ward's linkage and the DTW distance, log-returns (right)

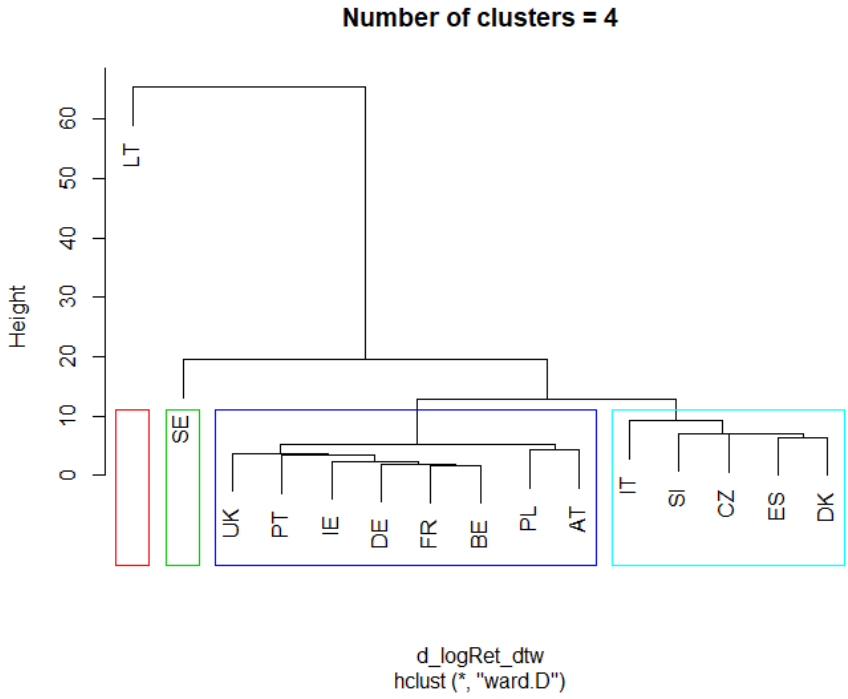


(a) Customization

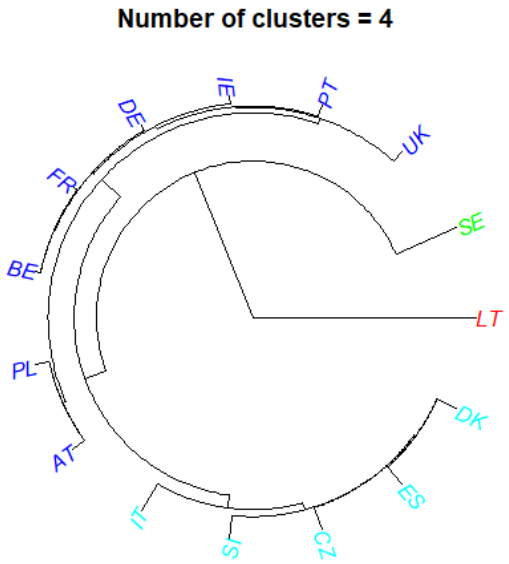


(b) Customization

Figure 4.59: Final hierarchical clustering for $k = 2$ based on DTW distance and ward linkage method, log-returns



(a) Customization



(b) Customization

Figure 4.60: Final hierarchical clustering for $k = 4$ based on DTW distance and ward linkage method, log-returns

For this particular dataset, we observe that we do not have the same results as

in the two previous samples which were considerably similar. This is to be expected and is due to the character of the transformation, which makes the time series more stationary and gives them this wiggly shape we examined in the descriptive statistics. The characteristic confirmed and indicated by the hierarchical analysis relates to the volatility and variability of the selected countries of the European Union. In every tree we take into account, Lithuania is the most isolated object, a fact that we have also discovered from the heatmap. This means that the country in question shows the greatest variability in the relative price changes. For $k = 4$, Sweden is the next followed by similar features, also constituting a single cluster and to a lesser degree accompanies Italy, which is however grouped with { Slovenia, Chechia, Espania and Denmark }. The rest of the countries are distinguished by less volatility and are mainly the countries that have a leading role in the European beef market. Finally, the two distance measures do not yield so similar results here as figure (4.58) argues, since the Euclidean distance for $k = 4$ isolates Italy into a separate cluster on its own and mixes slightly the rest time series.

5 Conclusion

The European Union is the third-largest producer of beef in the world, but it is at crossroads in terms of prospects. The appearance of modern consumption features, different social reforms and the prevalence of new beef export markets (e.g. Brazil, USA, Australia, India) are bringing Europe up with key decisions to make about its future. From an economic point of view, a major problem of this sector is the heterogeneity across countries in multiple sectors such as production, consumption, farm size, as well as costs and prices. The present study attempts to take a prime look at the weekly wholesale carcass prices distribution and group fifteen countries by applying hierarchical agglomerative analysis. This method was chosen among others such as the k-means, because it does not need to specify the number of clusters in advance and also produces a distinct hierarchical representation of the clusters. One major disadvantage is that it is not extremely suitable for a large number of observations, a fact that we did not encounter in our sample. However, by applying this method, we come across the different components of the algorithm which are subjectively controlled to their choice, with the testing of different cases being necessary. In particular, we need to concern about:

- What kind of transformation do we need to perform in the data?
- What dissimilarity measure should be used?
- What type of linkage should be used?
- Where should we cut the dendrogram to determine the efficient number of clusters?

The above queries are required to be resolved as efficiently as possible by anyone who is called upon to apply hierarchical analysis. Each hypothesis has an impact on the final outcome and the conclusions we reach, as well as each hypothesis may indicate data aspects. For this reason, we did various examinations that were presented in the empirical part, but also wanted to come to the one that best describes our data and draws the best conclusions. The conclusions and results for each step of the algorithm are presented respectively:

- For the data transformation, the raw data showed highly similar results to the corresponding sample in the standardized form. According to the literature, the scaled dataset is the one that stands out and is also proposed in this study for more accurate results, since standardization transforms observations to a comparable scale and does not reflect samples that may be dominated by variables with large values. The log-returns also yielded useful conclusions, after classifying the data according to the variability in the relative changes of the prices.

- It is always better to choose a distance measure based on accurate observation of the data and the purposes of the analysis. The euclidean and DTW distances used here also showed similar results, especially in the raw and scaled scenarios, mainly because our time series were of the same length. DTW was slightly better in the analysis since euclidean is a special case of the first. Finally, transforming the prices to log-returns removes the nonstationarity of the originals series by taking differences and therefore other dissimilarity measures constructed under the stationarity assumption can be used for further investigation.

- The hierarchical algorithm is also sensitive to the chosen linkage criterion since each linkage method has different systematic biases in the way it groups observations. Out of the four criteria described and used in the study, the ward's method returned stronger clustering structures and was used in the algorithm while the single linkage criterion yielded the weakest.

- The most important problem that exists in hierarchical clustering, and also in this particular application, is that the number of clusters that we have to present is not clear. However, an attempt was made to find it in conjunction with heatmaps visualizing the distance measure matrices. The ideal number seems to range from 2 to 4 clusters, with 4 appearing to probably be the ideal fit. However, all cases are presented in the analysis for a complete comparison.

After comparing all possible interconnections according to the above characteristics and presenting a complete view, empirical analysis of the distribution of carcass prices of the selected beef markets suggest:

- The diagrammatic representation of the trees with or without the clusters in which the countries are classified demonstrates the similarity between each pair of the comprised countries, since the tree itself can propose results in combination with heatmaps.

- Heterogeneity seems to be confirmed and fragmentation also appears to be prevalent in the European Union beef market as prices in European countries are unevenly distributed.

- Prices vary between countries and appear to be related to the countries' market share in production and intra-trade. More specifically, countries with a leading role and high levels of production in Europe were ranked together having the highest prices, while more distant markets had lower prices (Poland being the exception).

- The leading countries appear to be more stable according to their prices and seem to present a less important variability, with their time series being more predictable compared to other more insignificant countries in the market.

- With less certainty, the physical distance between spatial markets may be related to price similarity. We think this might be the case by looking at Germany for example being categorized and associated with Denmark and Austria or Ireland also being categorized with England since they are also significant strategic partners. However, we cannot be sure and it would be better to have more markets that were not included due to lacking observations, so that we can have a better perspective.

The application of cluster analysis to time series data can lead to very useful conclusions. It would be interesting for further analysis to consider:

- The use of the specific variables recommended in this study in the rest meat markets in Europe or further study of this sample through different sophisticated techniques that will probably overcome the missing values and will include more countries.

- The use of additional distance measures.

- The use of additional clustering algorithms such as partitioning algorithms, since some initial conclusions are presented in this study and so the reader can proceed in further calculations and tests.

- The use of further statistical tests to determine the appropriate number of clusters so that we can confidently conclude at a specific number that efficiently cuts the dendrogram.

Bibliography

[1] Jan Annigan. Why is meat important? `http://healthyeating.sfgate.com/meat-important-7213.html`. last modified December 07, 2018.

[2] Anthony Bagnall, Aaron Bostrom, James Large, and Jason Lines. The great time series classification bake off: An experimental evaluation of recently proposed algorithms. extended version. *arXiv preprint arXiv:1602.01711*, 2016.

[3] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA, 1994.

[4] Ján Buleca, Viliam Kováč, and Denisa Kočanová. Cluster analysis of beef production distribution in europe. *Potravinarstvo Slovak Journal of Food Sciences*, 12(1):789–797, 2018.

[5] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. From data mining to knowledge discovery in databases. *AI magazine*, 17(3):37–37, 1996.

[6] Grigoriadis V. Fousekis, P. Integration and hierarchy of national pork meat markets in the eu: An empirical analysis from the vantage of graph theory. *German Journal of Agricultural Economics*, 2019.

[7] Tak-chung Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181, 2011.

[8] Toni Giorgino et al. Computing and visualizing dynamic time warping alignments in r: the dtw package. *Journal of statistical Software*, 31(7):1–24, 2009.

[9] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[10] Jean-François Hocquette, Marie-Pierre Ellies-Oury, Michel Lherm, Christele Pineau, Claus Deblitz, and Linda Farmer. Current situation and future

prospects for beef production in europe—a review. *Asian-Australasian journal of animal sciences*, 31(7):1017, 2018.

[11] Sanjuán A. I and J. M. Gil. Price transmission analysis: a flexible methodological approach applied to european pork and lamb markets. *Applied Economics*, 33(1):123–131, 2001.

[12] Félix Iglesias and Wolfgang Kastner. Analysis of similarity measures in times series clustering for the discovery of building energy patterns. *Energies*, 6(2):579–597, 2013.

[13] Rico Ihle, LKE Dries, RA Jongeneel, TJ Venus, and JHH Wesseler. Research for agri committee-the eu cattle sector: challenges and opportunities-milk and meat: study. Technical report, European Parliament, 2017.

[14] Alboukadel Kassambara and Fabian Mundt. *factoextra: Extract and Visualize the Results of Multivariate Data Analyses*, 2019. R package version 1.0.6.

[15] Eamonn Keogh and Chotirat Ann Ratanamahatana. Exact indexing of dynamic time warping. *Knowledge and information systems*, 7(3):358–386, 2005.

[16] Eamonn J. Keogh and Shruti Kasetty. On the need for time series data mining benchmarks: a survey and empirical demonstration. In *KDD*, pages 102–111, 2002.

[17] T Warren Liao. Clustering of time series data—a survey. *Pattern recognition*, 38(11):1857–1874, 2005.

[18] Martin Maechler, Peter Rousseeuw, Anja Struyf, Mia Hubert, and Kurt Hornik. *cluster: Cluster Analysis Basics and Extensions*, 2019. R package version 2.1.0 — For new features, see the 'Changelog' file (in the package source).

[19] Pablo Montero, José A Vilar, et al. Tsclust: An r package for time series clustering. *Journal of Statistical Software*, 62(1):1–43, 2014.

[20] Usue Mori, Alexander Mendiburu, and Jose A Lozano. Distance measures for time series in r: The tsdist package. *R journal*, 8(2):451–459, 2016.

[21] Abdullah Mueen and Eamonn Keogh. Extracting optimal performance from dynamic time warping. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2129–2130. ACM, 2016.

[22] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018.

[23] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[24] RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio,

Inc., Boston, MA, 2016.

[25] Julie Ryschawy, Catherine Disenhaus, Sophie Bertrand, Gilles Allaire, Olivier Aznar, Sylvain Plantureux, Etienne Josien, Caroline Guinot, Jacques Lasseur, Christophe Perrot, et al. Assessing multiple goods and services derived from livestock farming on a nation-wide gradient. *animal*, 11(10):1861–1872, 2017.

[26] David Sankoff and JB Kruskal. Time warps, string edits, and macromolecules: The theory and practice of sequence comparison. 1983.

[27] Alexis Sarda-Espinosa. dtwclust: Time series clustering along with optimizations for the dynamic time warping distance. *R: A language and environment for statistical computing, version*, 5(0), 2016.

[28] Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

A

Additional descriptive and empirical results

Table A2: Production of meat: cattle 1000(t) in the EU-28

Countries	2012	2013	2014	2015	2016	2017	2018
EU-28	7.579,52	7.267,97	:	7.585,17	7.800,02	7.802,83	7.931,69
Belgium	262,28	249,91	257,67	267,88	278,36	281,54	277,31
Bulgaria	5,32	5,68	4,80	5,29	6,66	7,44	7,22
Czechia	65,71	64,83	65,53	68,29	71,93	67,72	71,58
Denmark	125,40	125,20	125,60	120,60	129,40	124,00	129,20
Germany	1.140,00	1.106,00	1.128,00	1.124,00	1.148,00	1.124,00	1.102,00
Estonia	7,96	7,88	:	9,62	9,43	9,00	8,56
Ireland	495,40	517,57	581,81	564,14	588,36	617,02	622,54
Greece	56,16	50,12	46,04	41,92	40,17	44,11	39,63
Spain	591,38	580,84	578,60	626,10	637,01	643,86	669,01
France	1.477,69	1.404,49	1.419,16	1.452,77	1.464,15	1.442,18	1.460,00
Croatia	46,78	47,27	44,42	42,26	44,43	42,20	43,78
Italy	981,12	855,32	709,43	788,28	809,66	756,42	809,22
Cyprus	5,31	4,57	4,60	5,74	7,04	8,31	5,28
Latvia	16,37	15,67	17,00	17,36	17,70	16,75	15,87
Lithuania	39,95	36,77	39,26	44,13	42,29	40,88	40,28
Luxembourg	8,47	7,95	8,48	9,08	9,42	9,54	9,87
Hungary	24,71	22,64	23,11	26,39	28,07	27,21	29,15
Malta	1,11	1,13	1,13	1,03	1,15	1,12	1,07
Netherlands	373,44	379,10	376,18	382,52	416,06	438,87	459,21
Austria	221,12	227,20	221,64	228,75	227,44	226,09	233,46
Poland	371,00	339,02	412,66	471,01	501,46	558,58	564,72
Portugal	92,99	84,09	79,84	88,62	91,10	91,09	93,79
Romania	28,82	29,28	29,20	44,47	57,53	59,14	49,92
Slovenia	33,09	32,10	31,57	33,58	35,66	35,79	34,87
Slovakia	9,76	9,53	8,83	8,40	8,29	7,79	8,11
Finland	80,37	80,42	82,32	85,76	86,37	85,39	86,48
Sweden	135,25	135,73	141,95	143,98	131,25	132,07	136,87
United Kingdom	882,56	847,66	877,58	883,21	911,66	904,73	922,70

Source: Eurostat (online data code: apro_mt_pann)

Table A3: Bovine livestock in the EU-28 (1.000 heads)

Countries	2011	2012	2013	2014	2015	2016	2017	2018
EU-28	87.054,22	87.296,95	87.734,43	88.405,62	89.138,27	89.134,16	88.818,84	87.400,29
Belgium	2.471,60	2.438,18	2.441,32	2.477,24	2.503,26	2.501,35	2.385,99	2.398,09
Bulgaria	567,53	535,32	585,55	562,36	561,04	570,14	552,92	542,12
Czechia	1.339,48	1.321,06	1.332,08	1.373,07	1.366,33	1.339,60	1.366,36	1.365,24
Denmark	1.612,00	1.607,00	1.583,00	1.553,00	1.566,00	1.554,00	1.558,00	1.530,00
Germany	12.527,84	12.506,77	12.685,99	12.742,19	12.635,46	12.466,59	12.281,20	11.949,09
Estonia	238,30	246,00	261,40	264,70	256,20	248,20	250,90	251,90
Ireland	5.925,32	6.253,24	6.309,05	6.243,05	6.422,23	6.613,43	6.673,59	6.593,49
Greece	681,00	685,00	653,00	659,00	582,00	554,00	556,00	542,00
Spain	5.923,11	5.812,61	5.802,22	6.078,73	6.182,91	6.317,64	6.465,75	6.510,59
France	19.129,00	19.052,00	19.129,00	19.271,00	19.406,00	19.004,00	18.975,48	18.563,23
Croatia	446,50	452,00	442,00	441,00	441,00	444,00	451,00	414,00
Italy	6.251,93	6.251,93	6.249,29	6.125,42	6.155,81	6.314,89	6.349,81	6.311,16
Cyprus	56,92	56,92	57,08	59,54	58,86	63,14	67,03	70,82
Latvia	380,61	393,10	406,49	422,02	419,08	412,31	405,82	395,33
Lithuania	752,40	729,20	713,50	736,60	722,60	694,80	676,90	653,50
Luxembourg	188,09	188,30	198,24	201,15	200,64	202,41	198,07	194,39
Hungary	697,00	760,00	782,00	802,00	821,00	852,00	870,00	885,00
Malta	15,07	15,59	15,22	14,88	15,02	14,36	14,18	14,12
Netherlands	3.912,00	3.985,00	4.090,00	4.169,00	4.315,00	4.294,00	4.030,00	3.690,00
Austria	1.976,53	1.955,62	1.958,28	1.961,20	1.957,61	1.954,39	1.943,48	1.912,81
Poland	5.500,94	5.520,35	5.589,54	5.660,27	5.762,50	5.970,20	6.035,70	6.183,30
Portugal	1.519,11	1.497,55	1.470,50	1.548,61	1.605,86	1.635,01	1.670,02	1.632,42
Romania	1.988,90	2.009,10	2.022,40	2.068,90	2.092,40	2.049,70	2.011,10	1.977,20
Slovenia	462,30	460,06	460,58	468,25	484,19	488,60	479,61	476,81
Slovakia	463,36	471,08	467,82	465,54	457,46	446,11	439,83	438,86
Finland	902,68	901,39	903,36	907,40	903,41	887,25	874,52	859,38
Sweden	1.449,73	1.443,58	1.443,52	1.436,49	1.428,40	1.436,05	1.448,59	1.435,45
United Kingdom	9.675,00	9.749,00	9.682,00	9.693,00	9.816,00	9.806,00	9.787,00	9.610,00

Source: Eurostat (online data code: apro_mt_lscatl)

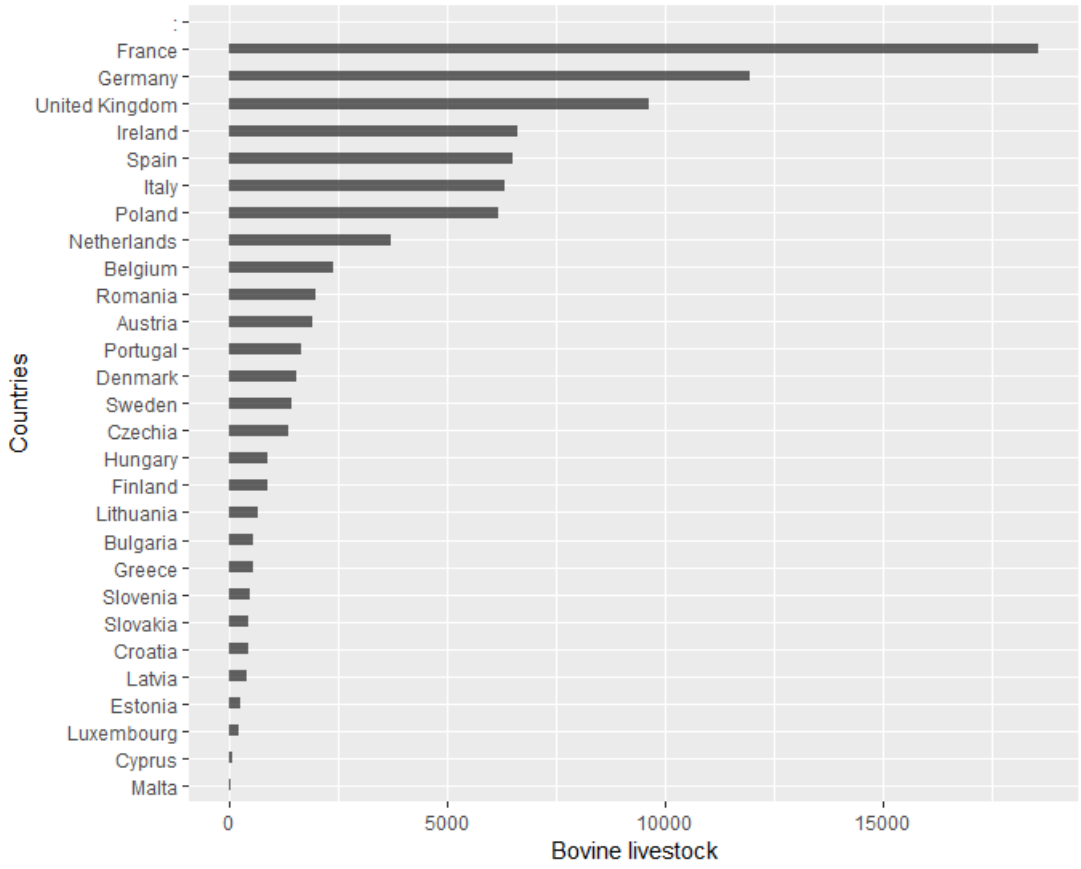


Figure A2: Bovine livestock in the EU-28, 2018

Source: Eurostat (calculations based on the online data code: apro_mt_lscatl)

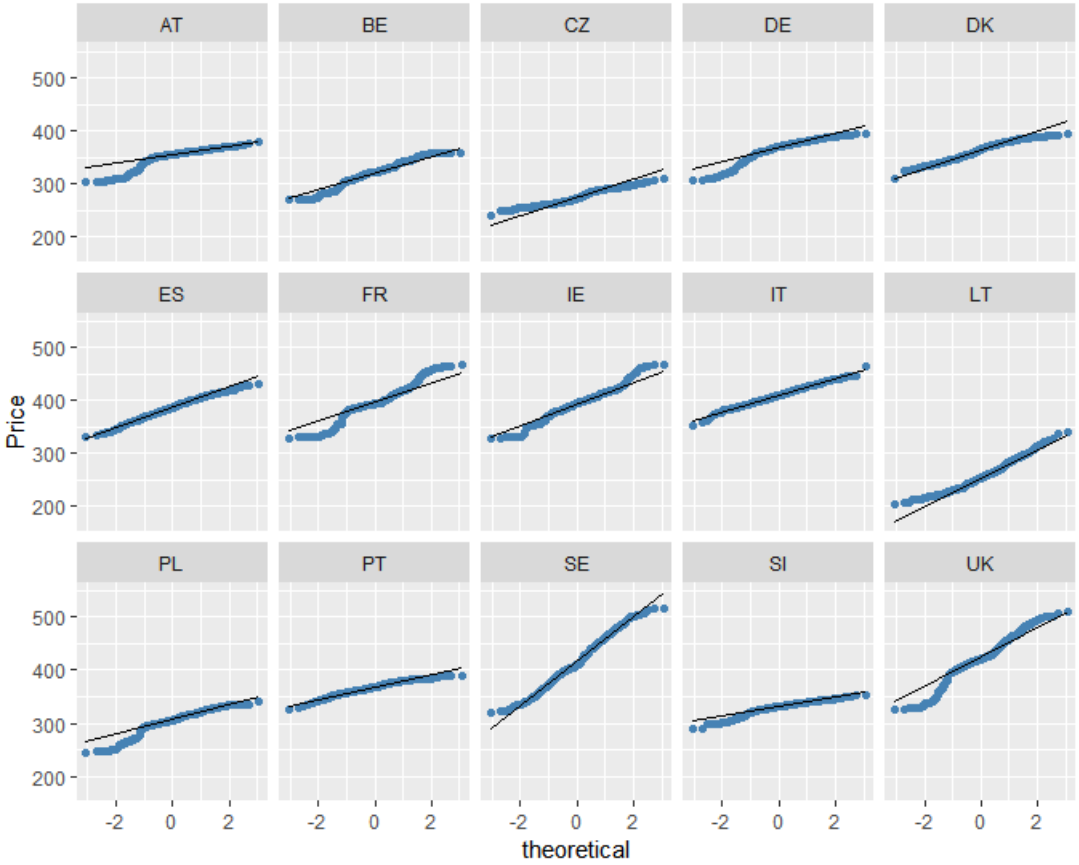


Figure A3: Q-Q plots of the raw prices

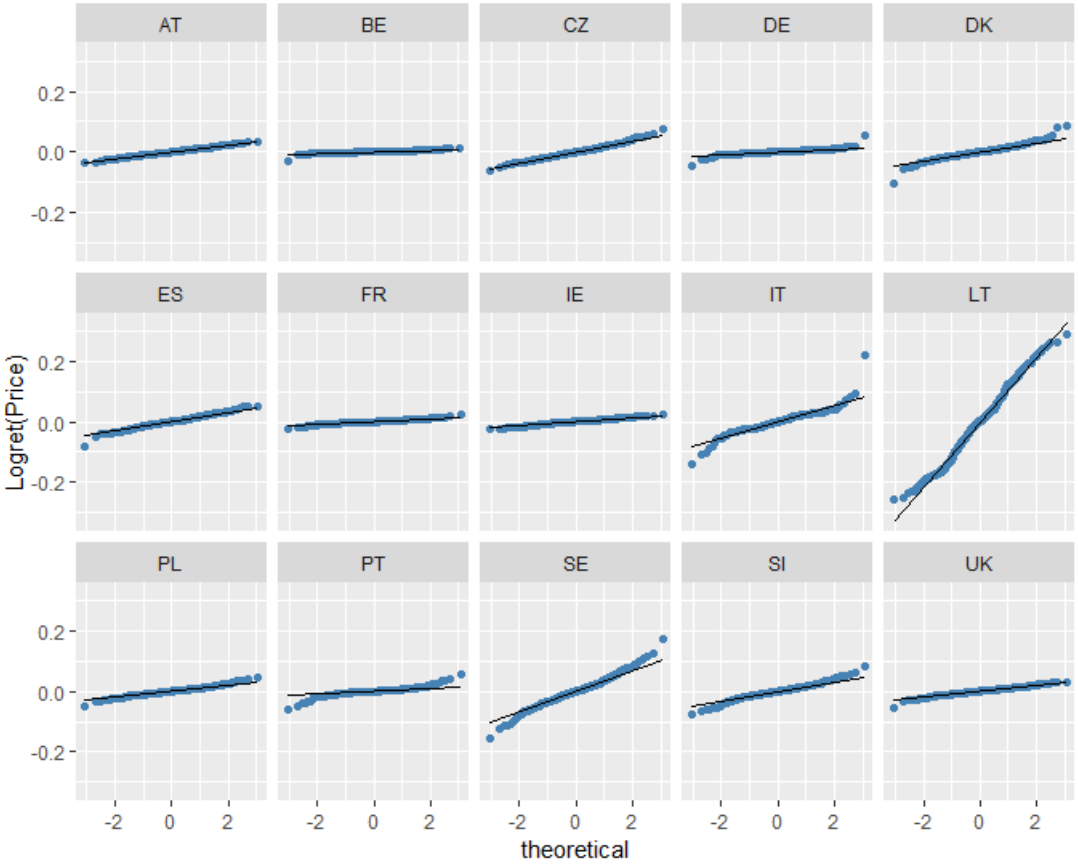


Figure A4: Q-Q plots of the log-return prices

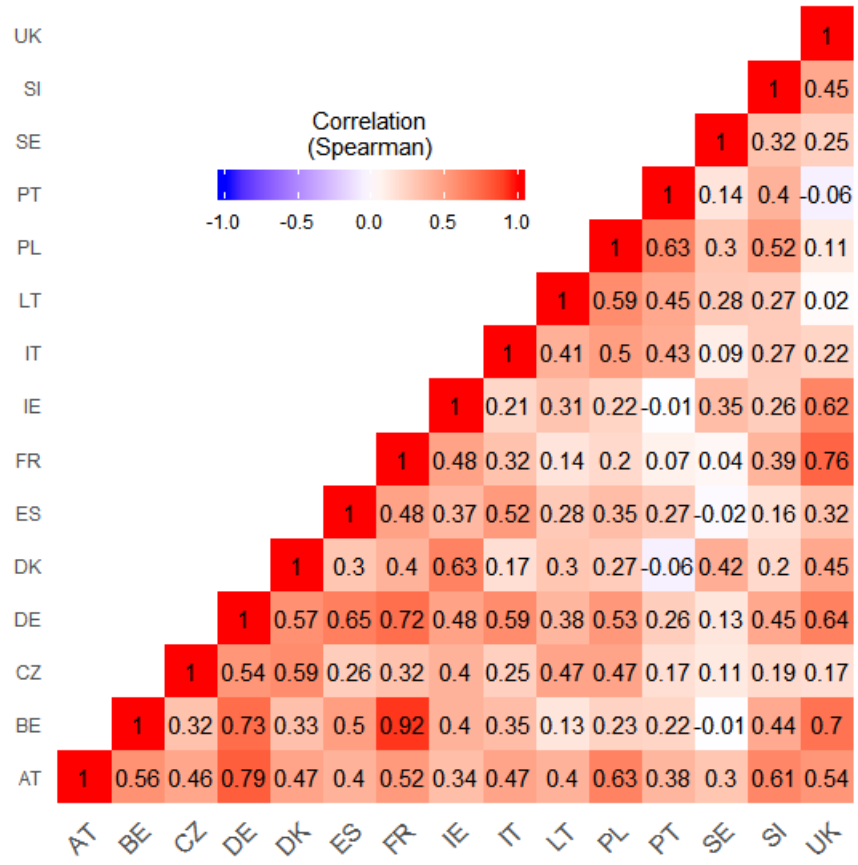


Figure A5: Correlation heatmap (Spearman) of the raw prices

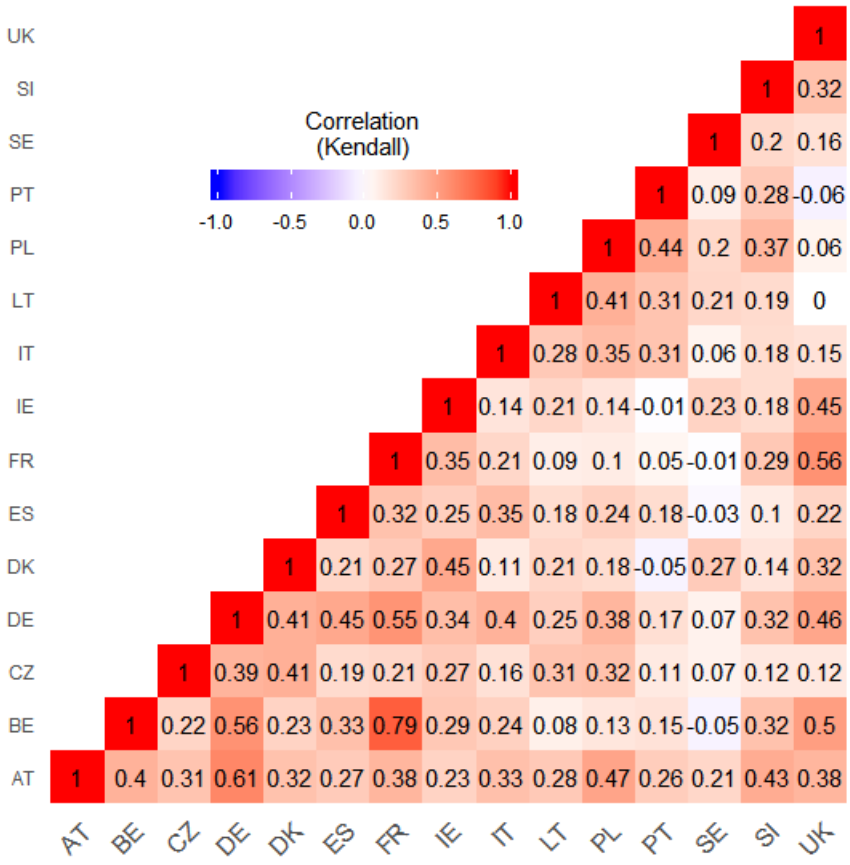


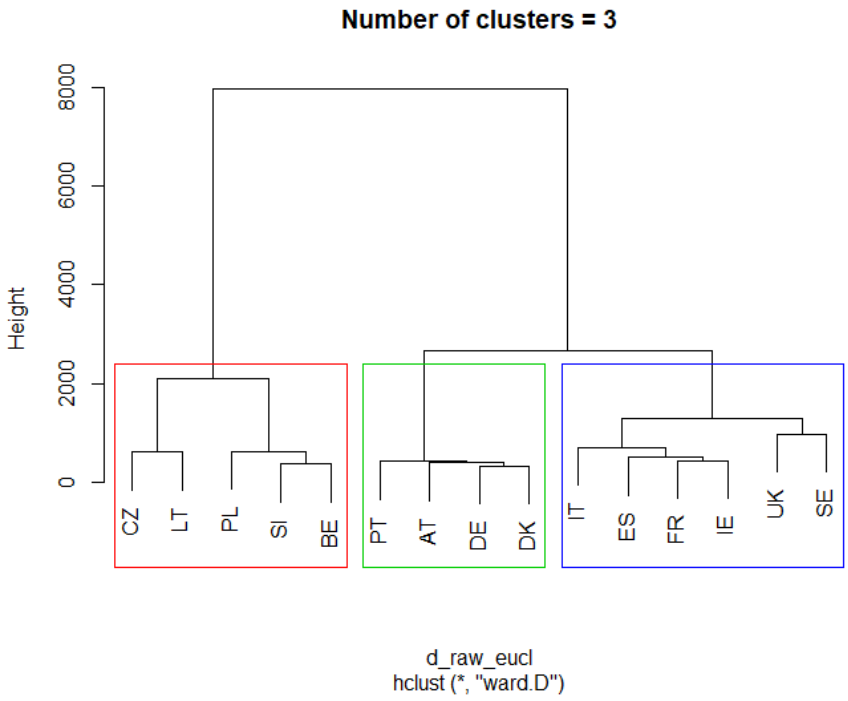
Figure A6: Correlation heatmap (Kendall) of the raw prices

Table A4: Dissimilarity matrix based on the euclidean distance, raw dataset

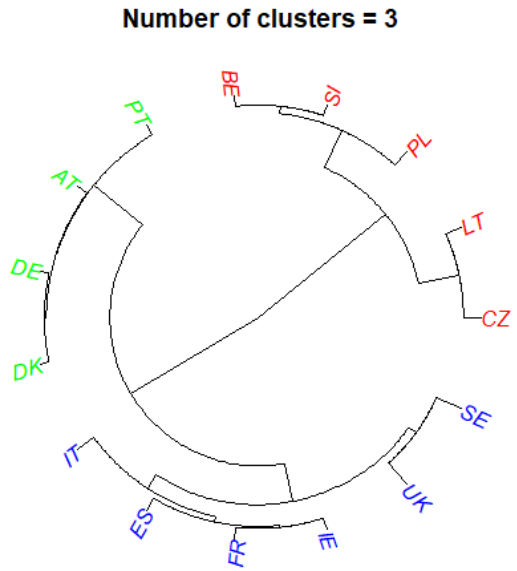
	DE	FR	UK	IT	IE	PL	ES	AT	PT	SI	SE	BE	CZ	DK	LT
DE	0.00	704.79	1313.10	991.14	698.07	1317.92	538.14	336.81	352.69	802.35	1343.78	960.45	1933.67	324.34	2358.58
FR	704.79	0.00	764.08	628.27	435.67	1981.61	484.48	1005.60	800.39	1463.15	1022.51	1571.83	2582.25	832.25	3004.77
UK	1313.10	764.08	0.00	813.77	823.40	2576.38	1023.02	1601.61	1381.55	2053.59	979.27	2193.36	3190.23	1410.85	3609.56
IT	991.14	628.27	813.77	0.00	637.88	2250.96	607.47	1270.30	935.81	1719.42	928.84	1907.97	2857.19	1059.72	3267.27
IE	698.07	435.67	823.40	637.88	0.00	1927.30	495.20	974.81	752.25	1407.55	954.31	1563.24	2522.59	731.87	2937.48
PL	1317.92	1981.61	2576.38	2250.96	1927.30	0.00	1779.72	1016.73	1372.28	600.07	2470.24	532.60	708.48	1309.01	1103.39
ES	538.14	484.48	1023.02	607.47	495.20	1779.72	0.00	818.46	543.98	1258.30	1119.56	1420.32	2383.57	643.81	2801.84
AT	336.81	1005.60	1601.61	1270.30	974.81	1016.73	818.46	0.00	440.08	498.77	1559.96	691.56	1638.60	417.38	2060.78
PT	352.69	800.39	1381.55	935.81	752.25	1372.28	543.98	440.08	0.00	836.75	1320.02	1055.32	1980.13	409.33	2395.82
SI	802.35	1463.15	2053.59	1719.42	1407.55	600.07	1258.30	498.77	836.75	0.00	1976.51	370.36	1189.75	788.40	1623.70
SE	1343.78	1022.51	979.27	928.84	954.31	2470.24	1119.56	1559.96	1320.02	1976.51	0.00	2176.24	3085.82	1351.18	3468.01
BE	960.45	1571.83	2193.36	1907.97	1563.24	532.60	1420.32	691.56	1055.32	370.36	2176.24	0.00	1048.12	972.01	1493.57
CZ	1933.67	2582.25	3190.23	2857.19	2522.59	708.48	2383.57	1638.60	1980.13	1189.75	3085.82	1048.12	0.00	1881.22	614.46
DK	324.34	832.25	1410.85	1059.72	731.87	1309.01	643.81	417.38	409.33	788.40	1351.18	972.01	1881.22	0.00	2318.10
LT	2358.58	3004.77	3609.56	3267.27	2937.48	1103.39	2801.84	2060.78	2395.82	1623.70	3468.01	1493.57	614.46	2318.10	0.00

Table A5: Dissimilarity matrix based on the DTW distance, raw dataset

	DE	FR	UK	IT	IE	PL	ES	AT	PT	SI	SE	BE	CZ	DK	LT
DE	0.00	9189.36	16897.69	14920.23	6849.16	20040.34	5993.34	3091.01	3773.10	9959.41	19230.34	13087.61	44656.19	2506.30	38427.91
FR	9189.36	0.00	4604.72	8013.48	3597.17	40995.86	4294.02	13905.92	8716.70	23487.78	6711.00	36285.50	67303.10	11140.66	62087.51
UK	16897.69	4604.72	0.00	8238.60	5605.27	50645.44	9188.76	22911.47	18886.19	35238.20	6671.81	42423.91	77772.48	17816.58	70629.43
IT	14920.23	8013.48	8238.60	0.00	6723.91	62837.97	6748.51	25566.67	15249.25	40564.08	8711.55	41235.67	81671.54	13046.43	85347.14
IE	6849.16	3597.17	5605.27	6723.91	0.00	38415.16	3969.96	10246.58	8398.55	22128.75	7075.91	27181.81	66300.51	7371.15	57964.49
PL	20040.34	40995.86	50645.44	62837.97	38415.16	0.00	38148.72	13529.39	29197.42	4776.50	48132.31	4162.01	7806.46	22092.25	9932.07
ES	5993.34	4294.02	9188.76	6748.51	3969.96	38148.72	0.00	9015.72	5880.66	20871.24	10042.57	27327.81	65249.78	5714.83	58970.46
AT	3091.01	13905.92	22911.47	25566.67	10246.58	13529.39	9015.72	0.00	4961.95	4401.69	22775.48	9825.01	36568.70	3753.25	31980.20
PT	3773.10	8716.70	18886.19	15249.25	8398.55	29197.42	5880.66	4961.95	0.00	12199.96	17209.08	20195.10	55181.25	4081.28	50074.92
SI	9959.41	23487.78	35238.20	40564.08	22128.75	4776.50	20871.24	4401.69	12199.96	0.00	32463.51	3515.26	21830.86	8858.51	21371.92
SE	19230.34	6711.00	6671.81	8711.55	7075.91	48132.31	10042.57	22775.48	17209.08	32463.51	0.00	43247.20	73432.09	19634.89	70029.48
BE	13087.61	36285.50	42423.91	41235.67	27181.81	4162.01	27327.81	9825.01	20195.10	3515.26	43247.20	0.00	15418.19	11075.26	15013.00
CZ	44656.19	67303.10	77772.48	81671.54	66300.51	7806.46	65249.78	36568.70	55181.25	21830.86	73432.09	15418.19	0.00	43871.56	7480.18
DK	2506.30	11140.66	17816.58	13046.43	7371.15	22092.25	5714.83	3753.25	4081.28	8858.51	19634.89	11075.26	43871.56	0.00	41940.40
LT	38427.91	62087.51	70629.43	85347.14	57964.49	9932.07	58970.46	31980.20	50074.92	21371.92	70029.48	15013.00	7480.18	41940.40	0.00

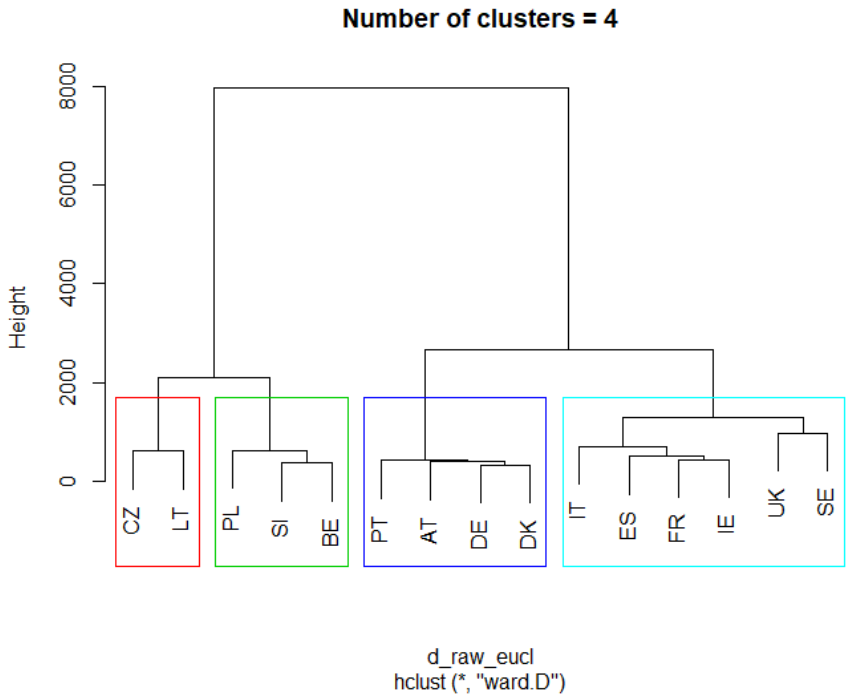


(a) Customization

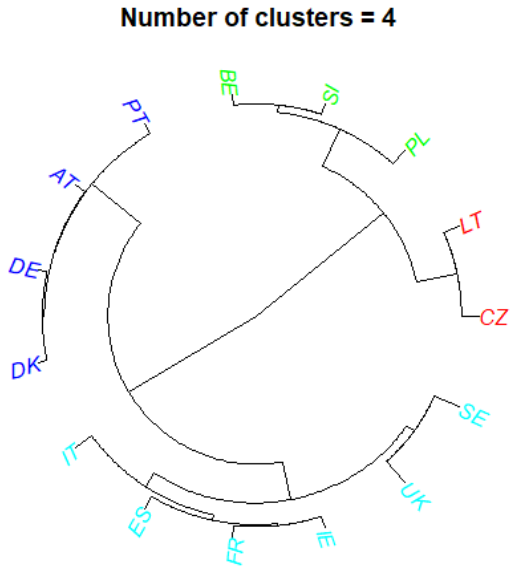


(b) Customization

Figure A7: Final hierarchical clustering for $k = 3$ based on euclidean distance and ward linkage method, raw data



(a) Customization



(b) Customization

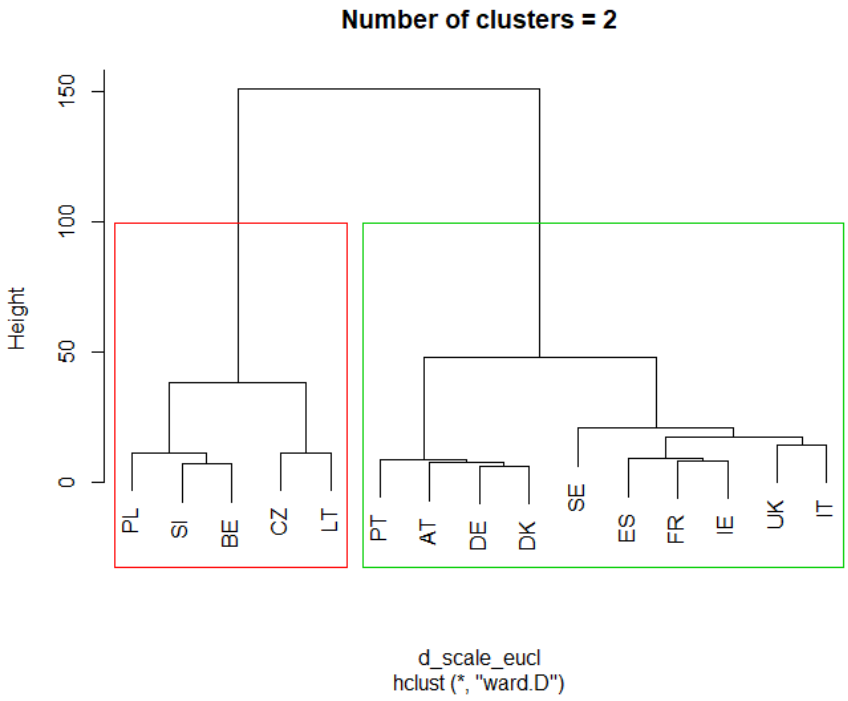
Figure A8: Final hierarchical clustering for $k = 4$ based on euclidean distance and ward linkage method, raw data

Table A6: Dissimilarity matrix based on the euclidean distance, scaled dataset

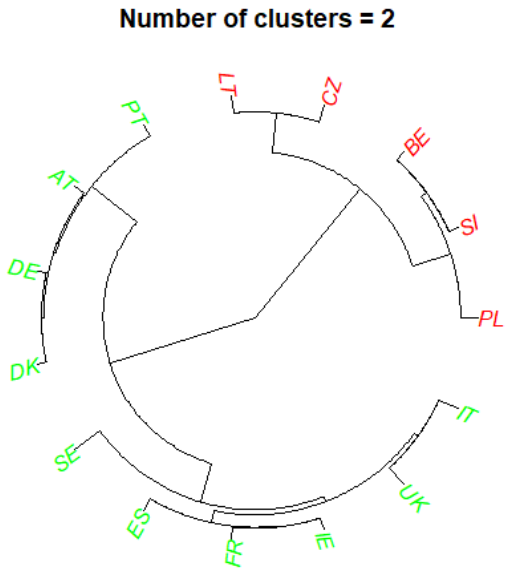
	DE	FR	UK	IT	IE	PL	ES	AT	PT	SI	SE	BE	CZ	DK	LT
DE	0.00	12.84	23.02	19.62	12.71	24.75	10.41	6.35	7.29	15.15	23.78	18.75	36.33	6.25	43.78
FR	12.84	0.00	13.29	12.79	8.38	36.71	8.85	18.35	14.15	27.03	17.84	29.77	48.08	15.41	55.47
UK	23.02	13.29	0.00	14.49	14.34	46.73	17.57	28.50	24.05	37.08	17.07	40.28	58.31	24.97	65.61
IT	19.62	12.79	14.49	0.00	12.76	43.22	12.22	24.78	17.82	33.23	16.49	37.46	54.54	20.70	61.73
IE	12.71	8.38	14.34	12.76	0.00	35.75	9.01	17.87	13.38	26.06	16.77	29.66	46.95	13.42	54.22
PL	24.75	36.71	46.73	43.22	35.75	0.00	33.70	19.14	26.66	11.39	45.10	9.52	13.52	24.78	20.27
ES	10.41	8.85	17.57	12.22	9.01	33.70	0.00	15.58	10.00	23.91	19.35	27.63	45.02	12.19	52.37
AT	6.35	18.35	28.50	24.78	17.87	19.14	15.58	0.00	9.07	9.52	27.89	13.63	30.82	8.11	38.18
PT	7.29	14.15	24.05	17.82	13.38	26.66	10.00	9.07	0.00	16.58	22.99	21.24	38.00	8.32	45.27
SI	15.15	27.03	37.08	33.23	26.06	11.39	23.91	9.52	16.58	0.00	35.94	7.16	22.28	14.98	29.97
SE	23.78	17.84	17.07	16.49	16.77	45.10	19.35	27.89	22.99	35.94	0.00	40.14	56.73	24.17	63.30
BE	18.75	29.77	40.28	37.46	29.66	9.52	27.63	13.63	21.24	7.16	40.14	0.00	19.08	19.03	26.98
CZ	36.33	48.08	58.31	54.54	46.95	13.52	45.02	30.82	38.00	22.28	56.73	19.08	0.00	35.37	11.60
DK	6.25	15.41	24.97	20.70	13.42	24.78	12.19	8.11	8.32	14.98	24.17	19.03	35.37	0.00	43.14
LT	43.78	55.47	65.61	61.73	54.22	20.27	52.37	38.18	45.27	29.97	63.30	26.98	11.60	43.14	0.00

Table A7: Dissimilarity matrix based on the DTW distance, scaled dataset

	DE	FR	UK	IT	IE	PL	ES	AT	PT	SI	SE	BE	CZ	DK	LT
DE	0.00	184.39	382.42	345.79	143.65	681.08	132.54	56.59	73.52	246.71	371.36	385.67	1181.74	51.56	1197.82
FR	184.39	0.00	79.61	145.95	63.63	1114.60	68.77	356.76	121.94	666.19	131.14	838.02	1626.54	158.07	1659.98
UK	382.42	79.61	0.00	144.27	116.39	1281.04	129.77	542.81	220.33	850.09	112.92	1028.71	1841.29	305.12	1805.42
IT	345.79	145.95	144.27	0.00	133.18	1343.40	129.71	488.58	167.55	788.74	120.99	1006.73	1699.99	216.21	1912.50
IE	143.65	63.63	116.39	133.18	0.00	1028.40	61.93	254.49	106.27	543.56	152.85	735.99	1498.34	119.25	1571.80
PL	681.08	1114.60	1281.04	1343.40	1028.40	0.00	963.56	457.60	751.81	162.79	1140.40	87.99	181.91	534.08	239.23
ES	132.54	68.77	129.77	129.71	61.93	963.56	0.00	206.16	70.12	466.63	180.42	649.70	1382.85	90.40	1508.98
AT	56.59	356.76	542.81	488.58	254.49	457.60	206.16	0.00	84.84	99.87	416.89	238.86	983.97	71.41	987.77
PT	73.52	121.94	220.33	167.55	106.27	751.81	70.12	84.84	0.00	250.39	211.85	455.23	1169.49	58.18	1326.75
SI	246.71	666.19	850.09	788.74	543.56	162.79	466.63	99.87	250.39	0.00	661.22	76.59	538.92	178.53	645.91
SE	371.36	131.14	112.92	120.99	152.85	1140.40	180.42	416.89	211.85	661.22	0.00	920.31	1645.30	300.47	1702.18
BE	385.67	838.02	1028.71	1006.73	735.99	87.99	649.70	238.86	455.23	76.59	920.31	0.00	290.64	340.59	499.61
CZ	1181.74	1626.54	1841.29	1699.99	1498.34	181.91	1382.85	983.97	1169.49	538.92	1645.30	290.64	0.00	959.39	142.46
DK	51.56	158.07	305.12	216.21	119.25	534.08	90.40	71.41	58.18	178.53	300.47	340.59	959.39	0.00	1128.53
LT	1197.82	1659.98	1805.42	1912.50	1571.80	239.23	1508.98	987.77	1326.75	645.91	1702.18	499.61	142.46	1128.53	0.00

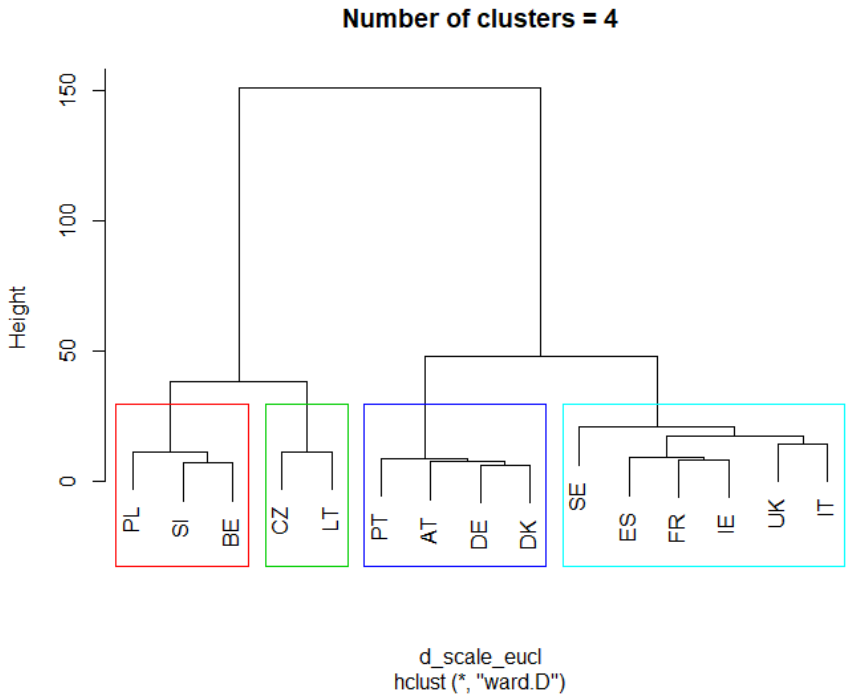


(a) Customization

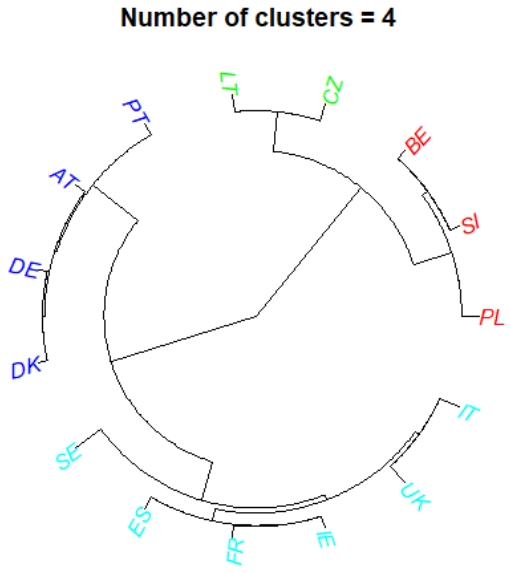


(b) Customization

Figure A9: Final hierarchical clustering for $k = 2$ based on euclidean distance and ward linkage method, scaled data



(a) Customization



(b) Customization

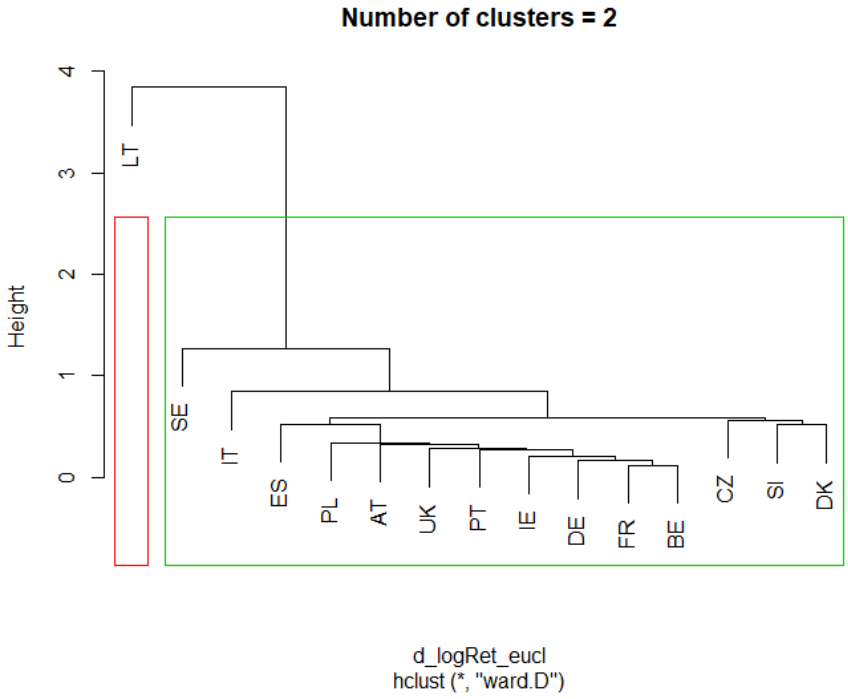
Figure A10: Final hierarchical clustering for $k = 4$ based on euclidean distance and ward linkage method, scaled data

Table A8: Dissimilarity matrix based on the euclidean distance, log-returns

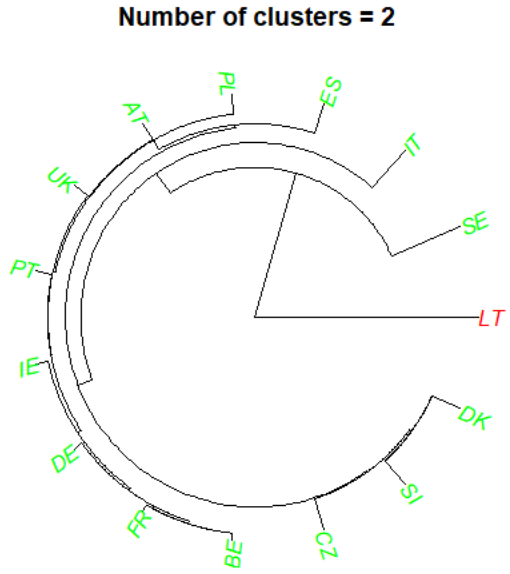
	DE	FR	UK	IT	IE	PL	ES	AT	PT	SI	SE	BE	CZ	DK	LT
DE	0.00	0.17	0.24	0.58	0.20	0.27	0.39	0.26	0.24	0.43	0.84	0.14	0.41	0.37	2.24
FR	0.17	0.00	0.24	0.58	0.19	0.26	0.37	0.25	0.23	0.41	0.82	0.12	0.40	0.38	2.26
UK	0.24	0.24	0.00	0.61	0.24	0.32	0.44	0.32	0.30	0.47	0.84	0.22	0.45	0.43	2.30
IT	0.58	0.58	0.61	0.00	0.59	0.63	0.69	0.63	0.62	0.69	1.01	0.57	0.66	0.64	2.27
IE	0.20	0.19	0.24	0.59	0.00	0.28	0.40	0.29	0.26	0.43	0.82	0.16	0.42	0.39	2.26
PL	0.27	0.26	0.32	0.63	0.28	0.00	0.42	0.35	0.32	0.46	0.86	0.25	0.45	0.43	2.26
ES	0.39	0.37	0.44	0.69	0.40	0.42	0.00	0.42	0.42	0.52	0.87	0.37	0.53	0.53	2.29
AT	0.26	0.25	0.32	0.63	0.29	0.35	0.42	0.00	0.32	0.46	0.84	0.24	0.46	0.43	2.25
PT	0.24	0.23	0.30	0.62	0.26	0.32	0.42	0.32	0.00	0.44	0.84	0.21	0.43	0.40	2.28
SI	0.43	0.41	0.47	0.69	0.43	0.46	0.52	0.46	0.44	0.00	0.90	0.40	0.58	0.52	2.27
SE	0.84	0.82	0.84	1.01	0.82	0.86	0.87	0.84	0.84	0.90	0.00	0.82	0.93	0.88	2.43
BE	0.14	0.12	0.22	0.57	0.16	0.25	0.37	0.24	0.21	0.40	0.82	0.00	0.39	0.36	2.25
CZ	0.41	0.40	0.45	0.66	0.42	0.45	0.53	0.46	0.43	0.58	0.93	0.39	0.00	0.53	2.28
DK	0.37	0.38	0.43	0.64	0.39	0.43	0.53	0.43	0.40	0.52	0.88	0.36	0.53	0.00	2.23
LT	2.24	2.26	2.30	2.27	2.26	2.26	2.29	2.25	2.28	2.27	2.43	2.25	2.28	2.23	0.00

Table A9: Dissimilarity matrix based on the DTW distance, log-returns

	DE	FR	UK	IT	IE	PL	ES	AT	PT	SI	SE	BE	CZ	DK	LT
DE	0.00	1.94	2.99	8.58	2.26	3.65	5.67	3.90	2.83	6.14	12.92	1.64	6.40	5.48	37.52
FR	1.94	0.00	2.98	8.62	2.20	3.50	5.56	3.78	2.82	6.13	12.98	1.63	6.35	5.39	37.62
UK	2.99	2.98	0.00	8.47	2.98	3.88	5.70	4.21	3.46	6.33	13.09	3.22	6.39	5.56	38.10
IT	8.58	8.62	8.47	0.00	8.74	8.13	8.04	8.14	8.51	8.54	13.86	8.81	8.17	8.24	38.99
IE	2.26	2.20	2.98	8.74	0.00	3.74	5.65	4.01	2.94	6.28	12.96	1.95	6.42	5.48	37.88
PL	3.65	3.50	3.88	8.13	3.74	0.00	5.52	4.22	3.90	6.26	13.00	3.72	6.27	5.54	38.05
ES	5.67	5.56	5.70	8.04	5.65	5.52	0.00	5.64	5.69	6.82	13.14	5.75	7.06	6.39	38.40
AT	3.90	3.78	4.21	8.14	4.01	4.22	5.64	0.00	4.11	5.97	12.89	3.97	6.06	5.42	38.26
PT	2.83	2.82	3.46	8.51	2.94	3.90	5.69	4.11	0.00	6.11	12.62	2.59	6.36	5.41	37.49
SI	6.14	6.13	6.33	8.54	6.28	6.26	6.82	5.97	6.11	0.00	12.82	6.28	7.10	6.81	38.08
SE	12.92	12.98	13.09	13.86	12.96	13.00	13.14	12.89	12.62	12.82	0.00	12.92	12.69	12.63	38.19
BE	1.64	1.63	3.22	8.81	1.95	3.72	5.75	3.97	2.59	6.28	12.92	0.00	6.46	5.52	37.19
CZ	6.40	6.35	6.39	8.17	6.42	6.27	7.06	6.06	6.36	7.10	12.69	6.46	0.00	6.44	38.56
DK	5.48	5.39	5.56	8.24	5.48	5.54	6.39	5.42	5.41	6.81	12.63	5.52	6.44	0.00	38.44
LT	37.52	37.62	38.10	38.99	37.88	38.05	38.40	38.26	37.49	38.08	38.19	37.19	38.56	38.44	0.00

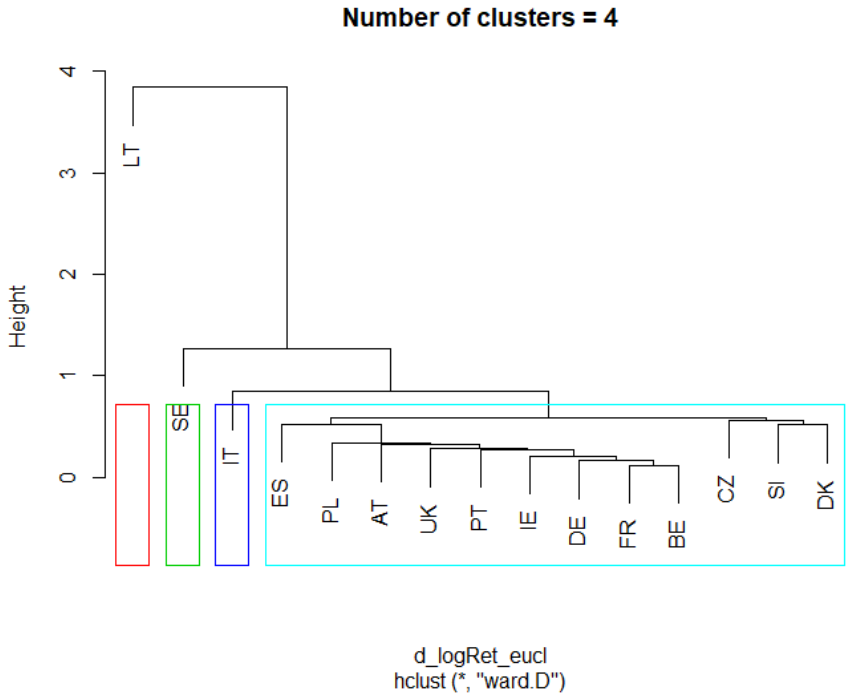


(a) Customization

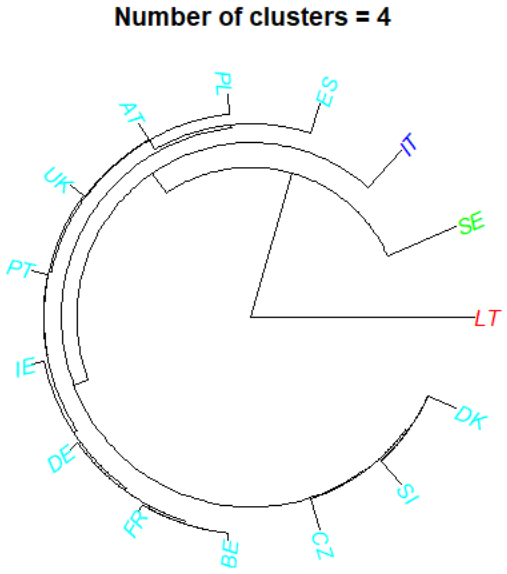


(b) Customization

Figure A11: Final hierarchical clustering for $k = 2$ based on euclidean distance and ward linkage method, log-returns



(a) Customization



(b) Customization

Figure A12: Final hierarchical clustering for $k = 4$ based on euclidean distance and ward linkage method, log-returns

B

R coding

B.1 Data downloading and cleaning

```
############ Packages used in R (libraries) ############

library(readxl)
library(tidyverse)
library(lubridate)
library(ggthemes)
library(ggrepel)
library(writexl)
library(stringr)
library(googlesheets)
library(tsibble)
library(psych)
library(plotly)
library(hrbrthemes)
library(dplyr)
library(readr)
library(tidyr)
library(lubridate)
library(DT)
library(purrr)
library(broom)
library(ggplot2)
library(ggthemes)
library(ggrepel)
library(xtable)
library(qqplotr)
library(hrbrthemes)
library(viridis)
library(ggExtra)
library(TSdist)
library(forcats)
library(TSclust)
library(cluster)
```

```r
library(factoextra)
library(dendextend)
library(dtw)
library(ape)
```

```r
############## Incerting the Data  ##############

url<- "https://ec.europa.eu/agriculture/sites/agriculture/files/
       market-observatory/meat/
       beef/doc/beef-historic-weekly-prices-2011onwards_en.xlsx"

download.file(url, "../data/beef-2011onwards.xlsx")

DF <- read_excel("../data/beef-2011onwards.xlsx", sheet = "ER3",
      skip = 4)
```

```r
############## Cleaning the Data ##############

DF <- DF[-c(453:475),]

names(DF)[1:3] <- c("Year", "Week", "Date")

zero_2_NA  <- function(x) { x <- ifelse(x == 0, NA, x) }

CNTRs <- c("DE", "FR", "UK", "IT", "IE", "PL", "ES", "AT", "PT",
           "SI", "SE", "BE", "CZ", "DK", "LT")

DF <- select(DF, c("Date", CNTRs))

log_return <- function(x) {
  x <- as.numeric(x)
  y <- c(NA, diff(log(x)))
  return (y)
}

beef_pri_spr <- DF %>%
                mutate(Date = ymd(Date)) %>%
                mutate_if(is.numeric, zero_2_NA) %>%
                filter(is.na(Date) == FALSE)


beef_pri_gat <- beef_pri_spr %>%
                gather(country, price, -Date) %>%
                mutate(price = as.numeric(price))


beef_ret_spr <- beef_pri_spr %>%
                mutate_if(is.numeric, log_return)


```

```r

beef_ret_gat <- beef_ret_spr %>%
                gather(country, ret, -Date)
```

B.2 Descriptive statistics

```r
############## Line graphs(raw) ##############

ggplot(beef_pri_gat, aes(x = Date, y = price)) +
       geom_line() +
       theme_bw() +
       xlab("Date") +
       ylab("Price") +
       facet_wrap(~country ,ncol=3, scales = "free_y")
```

```r
############## Boxplots(raw) ##############

beef_pri_gat %>%
             mutate(country = reorder(country, price)) %>%
             ggplot(aes(x = country, y = price)) +
             labs(x = "Countries", y = "Price") +
             geom_boxplot(fill = "#69b3a2" )+
             theme_bw() +
             stat_summary(fun.y=mean, geom="point", shape=18, size=3,
             color="white") +
             coord_flip()
```

```r
############## Histograms(raw) ##############

beef_pri_gat %>%
             ggplot(aes(x = price)) +
             theme_bw() +
             geom_histogram() +
             xlab("Price") +
             stat_bin( fill="#69b3a2", color="black", alpha=0.9) +
             ylab("Frequency") +
             facet_wrap(~country, ncol=3, scales = "free")
```

```r
############## Density plots(raw) ##############

beef_pri_gat %>%
```

```
                ggplot(aes(x = price)) +
                theme_bw() +
                xlab("Price") +
                ylab("Density") +
                geom_density(fill="#69b3a2", color="#e9ecef", alpha=0.9) +
                facet_wrap(~country, ncol=3, scales = "free")
```

```
############## Heatmap(Pearson) ##############


corm <- round(cor(beef_pri_spr[ , sort(c("UK", "SI", "SE", "PT", "PL",
            "LT", "IT", "IE", "FR", "ES", "DK", "DE", "CZ", "BE", "AT"))
    ],
          method = "pearson", use = "pairwise.complete.obs"),2)
corm[lower.tri(corm)] <- NA
corm


corm <- melt(corm)
corm$Var1 <- as.character(corm$Var1)
corm$Var2 <- as.character(corm$Var2)
corm <- na.omit(corm)
head(corm, 10)
corm


corheatmap<-ggplot(corm, aes(x = Var2, y = Var1)) +
                    geom_raster(data = corm, aes(fill = value),
                    color = "white") +
                    scale_fill_gradient2(low = "blue", high = "red",
                    mid = "white",
                    midpoint = 0, limit = c(-1, 1),
                    name = "Correlation\n(Pearson)") +
                    theme(axis.text.x = element_text(angle = 45,
                    size = 11,vjust = 1, hjust = 1),
                    axis.title.x = element_blank(),
                    axis.title.y = element_blank(),
                    panel.background = element_blank(),
                    legend.justification = c(1, 0),
                    legend.position = c(0.6, 0.7),
                    legend.direction = "horizontal") +
                    guides(fill = guide_colorbar(barwidth = 7,
                    barheight = 1,
                    title.position = "top", title.hjust = 0.5)) +
              coord_equal()


corheatmap +
          geom_text(aes(Var2, Var1, label = value), color = "black",
          size = 4) +
          theme(
              axis.title.x = element_blank(),
              axis.title.y = element_blank(),
```

```
            panel.grid.major = element_blank(),
            panel.border = element_blank(),
            panel.background = element_blank(),
            axis.ticks = element_blank(),
            legend.justification = c(1, 0),
            legend.position = c(0.6, 0.7),
            legend.direction = "horizontal") +
          guides(fill = guide_colorbar(barwidth = 10, barheight = 1.,
                 title.position = "top", title.hjust = 0.5))
```

```
############## Line graphs(logRet) ##############

ggplot(beef_ret_gat, aes(x = Date, y = ret)) +
       geom_line() +
       theme_bw() +
       xlab("Date") +
       ylab("LogReturn(Price)") +
       facet_wrap(~country ,ncol=3,  scales = "free_y")
```

```
############## Histograms(logRet) ##############

beef_pri_gat %>%
             ggplot(aes(x = price)) +
             theme_bw() +
             geom_histogram() +
             xlab("Price") +
             stat_bin( fill="#69b3a2", color="black", alpha=0.9) +
             ylab("Frequency") +
             facet_wrap(~country, ncol=3, scales = "free")
```

```
############# Density plots(logRet) #############

beef_pri_gat %>%
              ggplot(aes(x = price)) +
              theme_bw() +
              xlab("Price") +
              ylab("Density") +
              geom_density(fill="#69b3a2", color="#e9ecef", alpha=0.9) +
              facet_wrap(~country, ncol=3, scales = "free")
```

```
############# Boxplots(logRet) #############

beef_ret_gat %>%
```

```
                mutate(country = reorder(country, ret)) %>%
                ggplot(aes(x = country, y = ret)) +
                labs(x = "Countries", y = "LogRet(price)") +
                theme_bw() +
                geom_boxplot(fill = "#69b3a2") +
                stat_summary(fun.y=mean, geom="point", shape=18,
                size=3, color="white") +
                coord_flip()
```

```
############## Descriptive statistics tables ##############

desraw<-describe(beef_pri_spr)
print(desraw,digits=3)

#########

deslog<-describe(beef_ret_spr)
print(deslog,digits=3)
```

B.3 Empirical analysis (raw data)

```
############## Creating the euclidean matrix ##############

beef_pri_spr_2<-na.omit(beef_pri_spr)
table_raw <- t(beef_pri_spr_2[-1])
d_raw_eucl<-dist(table_raw, method = "euclidean", upper=TRUE)
d_raw_eucl
```

```
############## Euclidean matrix vizualization ##############

fviz_dist(d_raw_eucl, lab_size = 8, gradient = list(low = "#00AFBB",
          mid = "white", high = "#FC4E07"))
```

```
############## Dendrograms (Euclidean) ##############

hc1 <- hclust(d_raw_eucl, method="ward.D")
hc1 = as.dendrogram(hc1)
plot(hc1, ylab="Height", main="Ward linkage")


hc2 <- hclust(d_raw_eucl, method="complete")
```

```r
hc2 = as.dendrogram(hc2)
plot(hc2, ylab="Height", main="Complete linkage")


hc3 <- hclust(d_raw_eucl, method="average")
hc3 = as.dendrogram(hc3)
plot(hc3, ylab="Height", main="Average linkage")


hc4 <- hclust(d_raw_eucl, method="single")
hc4 = as.dendrogram(hc4)
plot(hc4, ylab="Height", main="Single linkage")
```

```r
############## Agglomerative coefficient (Euclidean) ##############

m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")
ac <- function(x) {
  agnes(d_raw_eucl, method = x)$ac
}

map_dbl(m, ac)
```

```r
############### Creating the DTW matrix ###############

d_raw_dtw<-dist(table_raw, method = "dtw", upper=TRUE)
d_raw_dtw
```

```r
############### Euclidean matrix vizualization ###############

fviz_dist(d_raw_dtw, lab_size = 8, gradient = list(low = "#00AFBB",
          mid = "white", high = "#FC4E07"))
```

```r
############### Dendrograms (DTW) ###############

hc5 <- hclust(d_raw_dtw, method="ward.D")
hc5 = as.dendrogram(hc5)
plot(hc5, ylab="Height", main="Ward linkage")


hc6 <- hclust(d_raw_dtw, method="complete")
hc6 = as.dendrogram(hc6)
plot(hc6, ylab="Height", main="Complete linkage")
```

```
11
12
13 hc7 <- hclust(d_raw_dtw, method="average")
14 hc7 = as.dendrogram(hc7)
15 plot(hc7, ylab="Height", main="Average linkage")
16
17
18 hc8 <- hclust(d_raw_dtw, method="single")
19 hc8 = as.dendrogram(hc8)
20 plot(hc8, ylab="Height", main="Single linkage")
```

```
1  ############## Agglomerative coefficient (DTW) ##############
2
3 m <- c( "average", "single", "complete", "ward")
4 names(m) <- c( "average", "single", "complete", "ward")
5 ac <- function(x) {
6   agnes(d_raw_dtw, method = x)$ac
7 }
8
9 map_dbl(m, ac)
```

```
1  ############## Determing k methods ##############
2
3 fviz_nbclust(table_raw, hcut, method = "wss") +
4              geom_vline(xintercept = 4, linetype = 2) +
5              labs(subtitle = "Elbow method") +
6              theme_minimal()
7
8
9 fviz_nbclust(table_raw, hcut, method = "silhouette") +
10             labs(subtitle = "Silhouette method") +
11             theme_minimal()
12
13
14 set.seed(123)
15 fviz_nbclust(table_raw, hcut, nstart = 25,  method = "gap_stat",
16             nboot = 50) +
17             labs(subtitle = "Gap statistic method") +
18             theme_minimal()
```

```
1  ############## Comparing the trees ##############
2
3 dend_list <- dendlist(hc1, hc5)
4 tanglegram(hc1,hc5, main = paste("entanglement =",
5            round(entanglement(dend_list), 2)))
```

```
############## Final trees (DTW, ward: k=3,4) ##############

###### k=3
hc5 <- hclust(d_raw_dtw, method="ward.D")
plot(hc5, ylab="Height", main="Number of clusters = 3")
rect.hclust(hc5, k = 3, border = 2:4)


mypal = c("green", "blue", "red")
clus1 = cutree(hc5, 3)
plot(as.phylo(hc5), type = "fan", tip.color = mypal[clus1],
     main="Number of clusters = 3", use.edge.length = TRUE)

###### k=4
hc5 <- hclust(d_raw_dtw, method="ward.D")
plot(hc5, ylab="Height", main="Number of clusters = 4")
rect.hclust(hc5, k = 4, border = 2:5)


mypal = c("blue", "cyan1", "red", "green")
clus2 = cutree(hc5, 4)
plot(as.phylo(hc5), type = "fan", tip.color = mypal[clus2],
     main="Number of clusters = 4", use.edge.length = TRUE)
```

```
############## Grouped line graphs k=3,4 ##############

beef_pri_gat_2 <- beef_pri_spr_2 %>%
                  gather(country, price, -Date) %>%
                  mutate(price = as.numeric(price))


p1<-beef_pri_gat_2 %>%
    filter(country %in% c('UK',"SE", "IT","ES", "FR", "IE")) %>%
    ggplot() +
    aes(x = Date, y = price, colour = country) +
    geom_line(size = 0.8) +
    theme_bw() +
    ylab("Price") +
    theme(legend.title=element_blank()) +
    theme(text = element_text(size=15))
p1


p2<-beef_pri_gat_2 %>%
    filter(country %in% c("PT", "AT", "DE", "DK")) %>%
    ggplot() +
    aes(x = Date, y = price, colour = country) +
    geom_line(size = 0.8) +
    ylab("Price") +
    theme(legend.title=element_blank()) +
    theme(text = element_text(size=15))
p2
```

```r

p3<-beef_pri_gat_2 %>%
     filter(country %in% c('PL',"SI", "BE")) %>%
     ggplot() +
     aes(x = Date, y = price, colour = country) +
     geom_line(size = 0.8) +
     theme_bw() +
     ylab("Price") +
     theme(legend.title=element_blank()) +
     theme(text = element_text(size=15))
p3


p4<-beef_pri_gat_2 %>%
     filter(country %in% c('CZ',"LT")) %>%
     ggplot() +
     aes(x = Date, y = price, colour = country) +
     geom_line(size = 0.8) +
     theme_bw() +
     ylab("Price") +
     theme(legend.title=element_blank()) +
     theme(text = element_text(size=15))
p4


p5<-beef_pri_gat_2 %>%
     filter(country %in% c('PL',"SI", "BE","CZ", "LT")) %>%
     ggplot() +
     aes(x = Date, y = price, colour = country) +
     geom_line(size = 0.8) +
     theme_bw() +
     ylab("Price") +
     theme(legend.title=element_blank()) +
     theme(text = element_text(size=15))
p5

gridExtra::grid.arrange(p1, p2 ,p5, nrow = 3)
gridExtra::grid.arrange(p1, p2, p3 ,p4, nrow = 4)
```

B.4 Empirical analysis (scaled data)

```r
 ############## Creating the euclidean matrix ##############

table_scale <- scale(table_raw)
d_scale_eucl<-dist(table_scale, method = "euclidean", upper=TRUE)
d_scale_eucl
```

```
############## Euclidean matrix vizualization ##############

fviz_dist(d_scale_eucl, lab_size = 8, gradient = list(low = "#00AFBB",
          mid = "white", high = "#FC4E07"))
```

```
############## Dendrograms (Euclidean) ##############

hc9 <- hclust(d_scale_eucl, method="ward.D")
hc9 = as.dendrogram(hc9)
plot(hc9, ylab="Height", main="Ward linkage")


hc10 <- hclust(d_scale_eucl, method="complete")
hc10 = as.dendrogram(hc10)
plot(hc10,  ylab="Height", main="Complete linkage")


hc11 <- hclust(d_scale_eucl, method="average")
hc11 = as.dendrogram(hc11)
plot(hc11, ylab="Height", main="Average linkage")


hc12 <- hclust(d_scale_eucl, method="single")
hc12 = as.dendrogram(hc12)
plot(hc12, ylab="Height", main="Single linkage")
```

```
############## Agglomerative coefficient (Euclidean) ##############

m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")
ac <- function(x) {
  agnes(d_scale_eucl, method = x)$ac
}

map_dbl(m, ac)
```

```
############## Creating the DTW matrix ##############

d_scale_dtw<-dist(table_scale, method = "dtw", upper=TRUE)
d_scale_dtw
```

```
############## Euclidean matrix vizualization ##############
```

```
2
3 fviz_dist(d_scale_dtw, lab_size = 8, gradient = list(low = "#00AFBB",
4           mid = "white", high = "#FC4E07"))
```

```
1  ############ Dendrograms (DTW) ############
2
3 hc13 <- hclust(d_scale_dtw, method="ward.D")
4 hc13 = as.dendrogram(hc13)
5 plot(hc13, ylab="Height", main="Ward linkage")
6
7
8 hc14 <- hclust(d_scale_dtw, method="complete")
9 hc14 = as.dendrogram(hc14)
10 plot(hc14, ylab="Height", main="Complete linkage")
11
12
13 hc15 <- hclust(d_scale_dtw, method="average")
14 hc15 = as.dendrogram(hc15)
15 plot(hc15, ylab="Height", main="Average linkage")
16
17
18 hc16 <- hclust(d_scale_dtw, method="single")
19 hc16 = as.dendrogram(hc16)
20 plot(hc16, ylab="Height", main="Single linkage")
```

```
1  ############ Agglomerative coefficient (DTW) ############
2
3 m <- c( "average", "single", "complete", "ward")
4 names(m) <- c( "average", "single", "complete", "ward")
5 ac <- function(x) {
6   agnes(d_scale_dtw, method = x)$ac
7 }
8
9 map_dbl(m, ac)
```

```
1  ############ Determing k methods ############
2
3 fviz_nbclust(table_scale, hcut, method = "wss") +
4              geom_vline(xintercept = 4, linetype = 2) +
5              labs(subtitle = "Elbow method") +
6              theme_minimal()
7
8
9
10 fviz_nbclust(table_scale, hcut, method = "silhouette") +
11             labs(subtitle = "Silhouette method") +
```

```
                theme_minimal()


set.seed(123)
fviz_nbclust(table_scale, hcut, nstart = 25,  method = "gap_stat",
                nboot = 50) +
                labs(subtitle = "Gap statistic method") +
                theme_minimal()
```

```
 ############## Comparing the trees ##############

dend_list <- dendlist(hc9, hc13)
tanglegram(hc9,hc13, main = paste("entanglement =",
            round(entanglement(dend_list), 2)))

dend_list <- dendlist(hc5, hc13)
tanglegram(hc5,hc13, main = paste("entanglement =",
            round(entanglement(dend_list), 2)))
```

```
 ############## Final trees (DTW, ward: k=2,4) ##############

###### k=2
hc13 <- hclust(d_scale_dtw, method="ward.D")
plot(hc13, ylab="Height", main="Number of clusters = 2")
rect.hclust(hc13, k = 2, border = 2:3)


mypal = c("green", "red")
clus3 = cutree(hc13, 2)
plot(as.phylo(hc13), type = "fan", tip.color = mypal[clus3],
     main="Number of clusters = 2", use.edge.length = TRUE)

###### k=4
hc13 <- hclust(d_scale_dtw, method="ward.D")
plot(hc13, ylab="Height", main="Number of clusters = 4")
rect.hclust(hc13, k = 4, border = 2:5)


mypal = c("blue", "cyan1", "red", "green")
clus4 = cutree(hc13, 4)
plot(as.phylo(hc13), type = "fan", tip.color = mypal[clus4],
      main="Number of clusters = 4", use.edge.length = TRUE)
```

B.5 Empirical analysis (log-returns)

```r
############## Creating the euclidean matrix ##############

beef_ret_spr_2<-na.omit(beef_ret_spr)
table_logRet <- t(beef_ret_spr_2[-1])
d_logRet_eucl<-dist(table_logRet, method = "euclidean", upper=TRUE)
d_logRet_eucl
```

```r
############## Euclidean matrix vizualization ##############

fviz_dist(d_logRet_eucl, lab_size = 8, gradient = list(low = "#00AFBB",
          mid = "white", high = "#FC4E07"))
```

```r
############## Dendrograms (Euclidean) ##############

hc17 <- hclust(d_logRet_eucl, method="ward.D")
hc17 = as.dendrogram(hc17)
plot(hc17, ylab="Height", main="Ward linkage")


hc18 <- hclust(d_logRet_eucl, method="complete")
hc18 = as.dendrogram(hc18)
plot(hc18, ylab="Height", main="Complete linkage")


hc19 <- hclust(d_logRet_eucl, method="average")
hc19 = as.dendrogram(hc19)
plot(hc19, ylab="Height", main="Average linkage")


hc20 <- hclust(d_logRet_eucl, method="single")
hc20 = as.dendrogram(hc20)
plot(hc20,  ylab="Height", main="Single linkage")
```

```r
############## Agglomerative coefficient (Euclidean) ##############

m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")
ac <- function(x) {
  agnes(d_logRet_eucl, method = x)$ac
}

map_dbl(m, ac)
```

```r
############## Creating the DTW matrix ##############

d_logRet_dtw<-dist(table_logRet, method = "dtw", upper=TRUE)
d_logRet_dtw
```

```r
############## Euclidean matrix vizualization ##############

fviz_dist(d_logRet_dtw, lab_size = 8, gradient = list(low = "#00AFBB",
          mid = "white", high = "#FC4E07"))
```

```r
############## Dendrograms (DTW) ##############

hc21 <- hclust(d_logRet_dtw, method="ward.D")
hc21 = as.dendrogram(hc21)
plot(hc21, ylab="Height", main="Ward linkage")


hc22 <- hclust(d_logRet_dtw, method="complete")
hc22 = as.dendrogram(hc22)
plot(hc22, ylab="Height", main="Complete linkage")


hc23 <- hclust(d_logRet_dtw, method="average")
hc23 = as.dendrogram(hc23)
plot(hc23, ylab="Height", main="Average linkage")


hc24 <- hclust(d_logRet_dtw, method="single")
hc24 = as.dendrogram(hc24)
plot(hc24, ylab="Height", main="Single linkage")
```

```r
############## Agglomerative coefficient (DTW) ##############

m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")
ac <- function(x) {
  agnes(d_logRet_dtw, method = x)$ac
}

map_dbl(m, ac)
```

```r
############## Determing k methods ##############
```

```
2
3  fviz_nbclust(table_logRet, hcut, method = "wss") +
4    geom_vline(xintercept = 4, linetype = 2)+
5    labs(subtitle = "Elbow method")+
6    theme_minimal()
7
8
9  fviz_nbclust(table_logRet, hcut, method = "silhouette")+
10   labs(subtitle = "Silhouette method")+
11   theme_minimal()
12
13
14 set.seed(123)
15 fviz_nbclust(table_logRet, hcut, nstart = 25,  method = "gap_stat",
       nboot = 50)+
16   labs(subtitle = "Gap statistic method")+
17   theme_minimal()
```

```
1  ############## Comparing the trees ##############
2
3  dend_list <- dendlist(hc17, hc21)
4  tanglegram(hc17,hc21, main = paste("entanglement =",
5             round(entanglement(dend_list), 2)))
```

```
1  ############## Final trees (DTW, ward: k=2,4) ##############
2
3  ###### k=2
4  hc21 <- hclust(d_logRet_dtw, method="ward.D")
5  plot(hc21, ylab="Height", main="Number of clusters = 2")
6  rect.hclust(hc21, k=2, border = 2:3)
7
8
9  mypal = c("green", "red")
10 clus5 = cutree(hc21, 2)
11 plot(as.phylo(hc21), type = "fan", tip.color = mypal[clus5],
12      main="Number of clusters = 2", use.edge.length = TRUE)
13
14 ###### k=4
15 hc21 <- hclust(d_logRet_dtw, method="ward.D")
16 plot(hc21, ylab="Height", main="Number of clusters = 4")
17 rect.hclust(hc21, k = 4, border = 2:5)
18
19
20 mypal = c("blue", "cyan1", "green", "red")
21 clus6 = cutree(hc21, 4)
22 plot(as.phylo(hc21), type = "fan", tip.color = mypal[clus6],
23      main="Number of clusters = 4", use.edge.length = TRUE)
```

B.6 Appendix figures

```
############ Q-Q plots ############

ggplot(beef_pri_gat, aes(sample = price)) +
       geom_qq(distribution = qnorm,  col = "steelblue") +
       geom_qq_line(line.p = c(0.25, 0.75), col = "black") +
       ylab("Price") +
       facet_wrap(~ country, nrow = 3) + ylab("Price")



ggplot(beef_ret_gat, aes(sample = ret)) +
       geom_qq(distribution = qnorm,  col = "steelblue") +
       geom_qq_line(line.p = c(0.25, 0.75), col = "black") +
       ylab("LogRet(Price)") +
       facet_wrap(~ country, nrow = 3) + ylab("Logret(Price)")
```

```
 ############## Heatmap(Kendall) ##############


corm <- round(cor(beef_pri_spr[ , sort(c("UK", "SI", "SE", "PT", "PL",
            "LT", "IT", "IE", "FR", "ES", "DK", "DE", "CZ", "BE", "AT"))
    ],
          method = "kendall", use = "pairwise.complete.obs"),2)
corm[lower.tri(corm)] <- NA
corm


corm <- melt(corm)
corm$Var1 <- as.character(corm$Var1)
corm$Var2 <- as.character(corm$Var2)
corm <- na.omit(corm)
head(corm, 10)
corm


corheatmap<-ggplot(corm, aes(x = Var2, y = Var1)) +
                   geom_raster(data = corm, aes(fill = value),
                   color = "white") +
                   scale_fill_gradient2(low = "blue", high = "red",
                   mid = "white",
                   midpoint = 0, limit = c(-1, 1),
                   name = "Correlation\n(Kendall)") +
                   theme(axis.text.x = element_text(angle = 45,
                   size = 11,vjust = 1, hjust = 1),
                   axis.title.x = element_blank(),
                   axis.title.y = element_blank(),
                   panel.background = element_blank(),
```

```
                    legend.justification = c(1, 0),
                    legend.position = c(0.6, 0.7),
                    legend.direction = "horizontal") +
                    guides(fill = guide_colorbar(barwidth = 7,
                    barheight = 1,
                    title.position = "top", title.hjust = 0.5)) +
              coord_equal()


corheatmap +
          geom_text(aes(Var2, Var1, label = value), color = "black",
          size = 4) +
          theme(
              axis.title.x = element_blank(),
              axis.title.y = element_blank(),
              panel.grid.major = element_blank(),
              panel.border = element_blank(),
              panel.background = element_blank(),
              axis.ticks = element_blank(),
              legend.justification = c(1, 0),
              legend.position = c(0.6, 0.7),
              legend.direction = "horizontal") +
          guides(fill = guide_colorbar(barwidth = 10, barheight = 1.,
                  title.position = "top", title.hjust = 0.5))


 ############### Heatmap(Spearman) ###############

corm <- round(cor(beef_pri_spr[ , sort(c("UK", "SI", "SE", "PT", "PL",
            "LT", "IT", "IE", "FR", "ES", "DK", "DE", "CZ", "BE", "AT"))
    ],
          method = "spearman", use = "pairwise.complete.obs"),2)
corm[lower.tri(corm)] <- NA
corm


corm <- melt(corm)
corm$Var1 <- as.character(corm$Var1)
corm$Var2 <- as.character(corm$Var2)
corm <- na.omit(corm)
head(corm, 10)
corm


corheatmap<-ggplot(corm, aes(x = Var2, y = Var1)) +
                    geom_raster(data = corm, aes(fill = value),
                    color = "white") +
                    scale_fill_gradient2(low = "blue", high = "red",
                    mid = "white",
                    midpoint = 0, limit = c(-1, 1),
                    name = "Correlation\n(Spearman)") +
                    theme(axis.text.x = element_text(angle = 45,
                    size = 11,vjust = 1, hjust = 1),
                    axis.title.x = element_blank(),
                    axis.title.y = element_blank(),
                    panel.background = element_blank(),
```

```
                    legend.justification = c(1, 0),
                    legend.position = c(0.6, 0.7),
                    legend.direction = "horizontal") +
                    guides(fill = guide_colorbar(barwidth = 7,
                    barheight = 1,
                    title.position = "top", title.hjust = 0.5)) +
          coord_equal()


corheatmap +
         geom_text(aes(Var2, Var1, label = value), color = "black",
         size = 4) +
         theme(
            axis.title.x = element_blank(),
            axis.title.y = element_blank(),
            panel.grid.major = element_blank(),
            panel.border = element_blank(),
            panel.background = element_blank(),
            axis.ticks = element_blank(),
            legend.justification = c(1, 0),
            legend.position = c(0.6, 0.7),
            legend.direction = "horizontal") +
         guides(fill = guide_colorbar(barwidth = 10, barheight = 1.,
                title.position = "top", title.hjust = 0.5))
```

```
##### Final trees (Raw data, euclidean, ward: k=3,4) #####

##### k=3
hc1 <- hclust(d_raw_eucl, method="ward.D")
plot(hc1, ylab="Height", main="Number of clusters = 3")
rect.hclust(hc1, k = 3, border = 2:4)



mypal = c("green", "blue", "red")
clus7 = cutree(hc1, 3)
plot(as.phylo(hc1), type = "fan", tip.color = mypal[clus7],
     main="Number of clusters = 3", use.edge.length = TRUE)

##### k=4
hc1 <- hclust(d_raw_eucl, method="ward.D")
plot(hc1, ylab="Height", main="Number of clusters = 4")
rect.hclust(hc1, k = 4, border = 2:5)



mypal = c("blue", "cyan1", "green", "red")
clus8 = cutree(hc1, 4)
plot(as.phylo(hc1), type = "fan", tip.color = mypal[clus8],
     main="Number of clusters = 4", use.edge.length = TRUE)
```

```
##### Final trees (Scaled data, euclidean, ward: k=2,4) #####

##### k=2
hc9 <- hclust(d_scale_eucl, method="ward.D")
plot(hc9, ylab="Height", main="Number of clusters = 2")
rect.hclust(hc9, k = 2, border = 2:3)


mypal = c("green", "red")
clus9 = cutree(hc9, 2)
plot(as.phylo(hc9), type = "fan", tip.color = mypal[clus9],
     main="Number of clusters = 2", use.edge.length = TRUE)


##### k=4
hc9 <- hclust(d_scale_eucl, method="ward.D")
plot(hc9, ylab="Height", main="Number of clusters = 4")
rect.hclust(hc9, k = 4, border = 2:5)



mypal = c("blue", "cyan1", "red", "green")
clus10 = cutree(hc9, 4)
plot(as.phylo(hc9), type = "fan", tip.color = mypal[clus10],
     main="Number of clusters = 4", use.edge.length = TRUE)
```

```
##### Final trees (Log-returns, euclidean, ward: k=2,4) #####

##### k=2
hc17 <- hclust(d_logRet_eucl, method="ward.D")
plot(hc17, ylab="Height", main="Number of clusters = 2")
rect.hclust(hc17, k = 2, border = 2:3)



mypal = c("green", "red")
clus11 = cutree(hc17, 2)
plot(as.phylo(hc1), type = "fan", tip.color = mypal[clus11],
     main="Number of clusters = 2", use.edge.length = TRUE)


##### k=4
hc17 <- hclust(d_logRet_eucl, method="ward.D")
plot(hc17, ylab="Height", main="Number of clusters = 4")
rect.hclust(hc17, k = 4, border = 2:5)

mypal = c("cyan1", "blue", "green", "red")
clus12 = cutree(hc17, 4)
plot(as.phylo(hc17), type = "fan", tip.color = mypal[clus12],
     main="Number of clusters = 4", use.edge.length = TRUE)
```