Pennacchioli et al. *EPJ Data Science* 2014, **3**:33
http://www.epjdatascience.com/content/3/1/33

EPJ Data Science
a SpringerOpen Journal

REGULAR ARTICLE Open Access

The retail market as a complex system

Diego Pennacchioli[1,2], Michele Coscia[3*], Salvatore Rinzivillo[2], Fosca Giannotti[2] and Dino Pedreschi[2,4]

*Correspondence:
michele_coscia@hks.harvard.edu
[3]CID, Harvard University, 79 JFK St, Cambridge, USA
Full list of author information is available at the end of the article

Abstract

Aim of this paper is to introduce the complex system perspective into retail market analysis. Currently, to understand the retail market means to search for local patterns at the micro level, involving the segmentation, separation and profiling of diverse groups of consumers. In other contexts, however, markets are modelled as complex systems. Such strategy is able to uncover emerging regularities and patterns that make markets more predictable, e.g. enabling to predict how much a country's GDP will grow. Rather than isolate actors in homogeneous groups, this strategy requires to consider the system as a whole, as the emerging pattern can be detected only as a result of the interaction between its self-organizing parts. This assumption holds also in the retail market: each customer can be seen as an independent unit maximizing its own utility function. As a consequence, the global behaviour of the retail market naturally emerges, enabling a novel description of its properties, complementary to the local pattern approach. Such task demands for a data-driven empirical framework. In this paper, we analyse a unique transaction database, recording the micro-purchases of a million customers observed for several years in the stores of a national supermarket chain. We show the emergence of the fundamental pattern of this complex system, connecting the products' volumes of sales with the customers' volumes of purchases. This pattern has a number of applications. We provide three of them. By enabling us to evaluate the sophistication of needs that a customer has and a product satisfies, this pattern has been applied to the task of uncovering the hierarchy of needs of the customers, providing a hint about what is the next product a customer could be interested in buying and predicting in which shop she is likely to go to buy it.

Keywords: marketing; complex systems; nestedness

1 Introduction

The retail market has been one among the most successful application scenarios for data mining research. Supermarkets generate a large amount of data each day, by recording which customers are buying which products, where and when. Traditional statistics tools have been abandoned, as unsuitable tools for dealing with such data richness, in favour of association rule mining [1], data clustering [2], OLAP techniques for business intelligence [3, 4] and other approaches. The common strategy shared by these tools is to segment, separate and profile diverse groups of consumers. Their typical result is to find unexpected pairwise relationships between products, or group together some customers given their purchase behaviour or personal data. We call this class of results 'local patterns', as they typically involve specific groups of customers/products, and they proved their usefulness in many real world scenarios [5–7].

There are alternative approaches to the analysis of other types of markets. In [8, 9] the global export market at the country level is modelled as a complex system. Rather than focusing on local patterns, the authors looked for a global pattern emerging from the self-organization of competing actors. Under such perspective, many fluctuating and unpredictable local behaviours can be interpreted as adjustments happening at a higher level. The world export market, then, ceases to be unpredictable and a global pattern emerges. Exploiting this new knowledge of the market as a complex system, authors are able to define the new concept of 'Economic Complexity' and prove that this measure is a very accurate predictor of a country's future growth, outperforming any other traditional socio-economical indicator. This approach is very successful and it has been replicated elsewhere [10].

In this paper, we introduce the idea of analysing the retail market as a complex system. Our approach is based on the observation that the retail market is composed by independent units, the customers, which act accordingly to their internal logic, the maximization of their utility function. By putting together these interacting units, the system of retail market starts showing properties of its own, as a result of the self organization of the customers. This approach has the potential to overcome some severe limitations of the classical retail market data mining. For instance, the output of association rule mining is usually composed by thousands rules, each describing a single particle of the customer behaviour, and selecting the most representative ones is usually a problem [11]. Moreover, usually many products are not present in the result set, as they are not frequently purchased, causing this description to be incomplete. On the downside, we forfeit the high granularity and precision achievable with data mining techniques.

By looking at the retail market as a complex system, we are able to define the Purchase Function, which is a description of the mechanics of this complex system at the global level. The Purchase Function enables us to enhance our knowledge about the system as a whole, describing both customers and products, and we prove its usefulness in three different analyses. First, we provide one empirical observation of Maslow's hierarchy of needs [12]. Using the Purchase Function we discover that highly ranked customers, with more sophisticated needs, tend to buy niche products, i.e., low-ranked products; on the other hand, low-ranked, low purchase volume customers tend buy only high-ranked product, very popular products that everyone buys.

Second, we propose a simple marketing application useful for targeted advertising. Given that the Purchase Function classifies the likelihood of a customer-product connection, a target marketing campaign may spot with a higher accuracy the smallest customer set that is likely to start buying a given product.

Finally, our third application is focused on the predictability of customer movements on the territory. We aim at predicting in which shop a customer will go to buy a given product. We show how the typical low-level information about the product (its price or its usual purchase amounts) have some explanatory power. However, our customer/product sophistication measure, derived from the Purchase Function, has a much greater explanatory power.

Our applications are founded on a data-driven empirical proof. We analyse a unique transaction database, collected by a retail supermarket chain in Italy, which recorded the micro-purchases of a million customers. Each customer is recognizable as the system

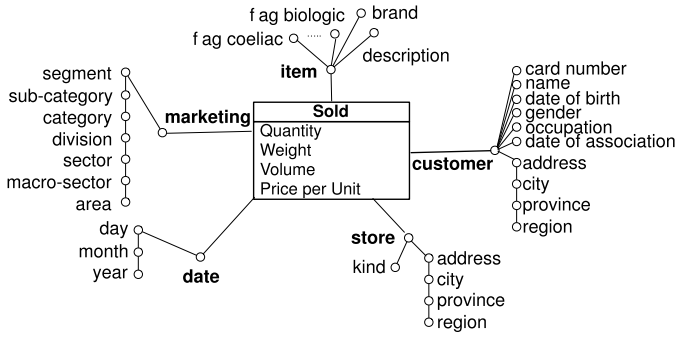


Figure 1 Data model. The data model of the Data Warehouse.

records her purchases using the identification code of her membership card. We are then able to track the purchases of each customers over a four year period, from 2007 to 2011.

In our analysis, we build the adjacency matrix of the bipartite network connecting a customer to the products she buys. This matrix has a triangular shape, consistent with the observations of the global export market [8, 9]. We prove that this shape is not expected, by implementing a simple null model of customer behaviour and observing that the fundamental properties of the observed matrix structure are not present in the null model matrix. Therefore, the observed system is indeed the product of a complex interaction not reducible to simple assumptions. We then proceed to define the Purchase Function, which divides the adjacency matrix in expressed and not expressed connections. This is the global pattern of the system and we can exploit it in our application scenarios.

2 The data

Our analysis is based on real world data about customer behaviour. The dataset we used is the retail market data of one of the largest Italian retail distribution companies. The conceptual data model of the data warehouse is depicted in Figure 1. The whole dataset contains retail market data in a time window that goes from January 1st, 2007 to December 31st, 2011. The active and recognizable customers are 1,066,020. A customer is active if she has purchased something during the data time window, while she is recognizable if the purchase has been made using a membership card. The customers of this supermarket with a card are very engaged in the shop itself: the supermarket is in fact a cooperative and whoever has the card is considered a member. This makes the data more valuable as the customers with a card have very high incentives to buy whatever they can in this supermarket, making it the primary (and sometimes only) source of the products they buy. In fact, a study by Bocconi showed that COOP is able to score among the highest in the metrics of customer fidelity.[a] The 138 stores of the company cover the whole west coast of Italy, selling 345,208 different items.

An important dimension of the data warehouse is Marketing, representing the classification of products: it is organized as a tree and it represents a hierarchy built on the product typologies, designed by marketing experts of the company. The top level of this hierarchy is called 'Area' that split the products into two fundamental categories: 'Food' and 'No Food'. The bottom level of the hierarchy, the one that contains the leaves of the tree, is called 'Segment' and it contains 7,003 different values. Hence, for each item contained in the dataset, there is an entry assigning it to the right path of the hierarchy tree.

Table 1 Distribution of the number of products per category

Food	2,026 (77.6%)	Fresh	1,005 (70.9%)	Regular	493 (78.8%)
				Very fresh	512 (63.2%)
		Various	1,021 (84.1%)	Chemicals	333 (83.4%)
				Grocery	688 (84.5%)
No food	2,791 (37.8%)	-	2,791 (37.8%)	House	565 (54.9%)
				Multimedia	368 (33.5%)
				Personal	746 (32.0%)
				Seasonal & DIY	1,112 (34.4%)

In parenthesis, we report the percentage of the products that are sold in all three types of shops in our dataset.

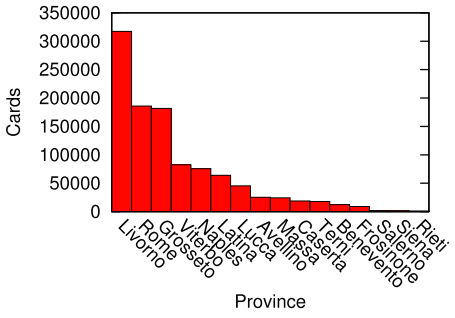


Figure 2 Card distribution. Distribution of the number of cards per province in our dataset.

The 'Store Kind' column refers to the shop classification: in increasing order of size we have 'Gestin', 'Super' and 'Iper'. 'Gestin' are usually low area shops, occupying the ground floor of a building, usually in the city center and in smaller towns and villages. 'Super' are larger, usually occupying their own building and built into larger cities just outside the city center. 'Iper' are usually an Italian equivalent of US malls.

In Table 1 we report how many segments are allocated in the top three levels of the marketing classification, proving that the supermarket is indeed selling a complete variety of products, not just grocery and fresh food. We also report the percentage of products that are sold at all three types of shops. While it is expected that the share of non-food products sold in smaller shops is lower, gestin shops still sell a significant quantity of them.[b] For example, the absolute number of DIY products sold in the smallest shops is practically equivalent to the absolute number of fresh food products.

Given that the dataset contains more than one million customers and almost 350k items, to build a matrix 'customers $\times$ items' would generate a $\sim$370 billion cells matrix, that is redundant for our purposes. Hence we need to reduce both dimensions (customers and items). There are two main criteria to select the customers: on the basis of their purchase behaviour (e.g. excluding from the analysis all the people that did not purchased at least a total number x items) or geographically (e.g. considering just the customers of an area). We decided to apply the latter filter, since we do not want to exclude any customer behaviour apriori. We select a subset of shops in the dataset belonging to the same areas of Italy. The number of cards per area is presented in Figure 2. Note that there might be some double-counting due to lost cards. However, this should not influence the analysis because customer behavior is constant regardless if she lost her card or not and, being our observations cumulative and normalized as explained in later sections, this double counting is bounded to have limited effect only in the fitted parameters, not in the overall phenomenon.

We generated different views of the dataset for different purposes. Our main dataset is Livorno2007-2009, that is including all the purchases of the customers located in the city of Livorno during the period from 2007 to 2009. We use only this view for the applications of the framework's output. We also generated the dataset Lazio2007-2009 (same period, different geographical location, the union of the cities of Rome, Viterbo, Latina, Rieti and Frosinone) and Livorno2010-2011 (different period, same geographical location). The two views are generated to prove that the fundamental properties of the adjacency matrix needed for our framework are not bounded to a particular place or time. The following steps of data preparation are applied equally to the different datasets extracted.

The second issue, as introduced above, regards the cardinality of products. There is a conceptual problem in using the level of detail of 'item': the granularity is too fine, making the analysis impractical as it would consider a very low detail level. The distinction between different packages of the same product, e.g. different sizes of bottles containing the same liquid, is not of interest in our study. A natural way to solve this problem is to use the marketing hierarchy of the products, substituting the item with its marketing Segment value. In this way, we reduce the cardinality of the dimension of the product by 98% (from 345,208 to 7,003), aggregating logically equivalent products.

The last step in data selection is to exclude from the analysis all segments that are either too frequent (e.g. the shopping bag) or meaningless for the purchasing analysis (e.g. discount vouchers, errors, segments never sold, etc.). After this last filter, and consequently the discharge of the customers that bought exclusively products classified under the removed segments, we got the adjacency matrix, the input to our framework. Livorno2007-2009 matrix has 317,269 customers and 4,817 segments, with 182,821,943 purchases; Livorno2010-2011 has 326,010 customers and 4,807 segments, with 183,679,550 purchases; and Lazio2007-2009 has 278,154 customers and 4,641 segments, with 135,517,300 purchases.

3 Methods

Analysing customers' purchase behaviour is one of major success stories of data mining research. The pioneer work on association rules [1] is still one of most cited papers in computer science. However, we believe that data mining is able to take into account only a part of the whole picture, not accounting for a great amount of valuable data. Firstly, it excludes customers, that are used only for counting the support of products. Secondly, as many natural phenomena, purchasing behaviour is characterized by long tail distributions. In Figures 3 and 4 we depict the cumulative frequency for products and customers

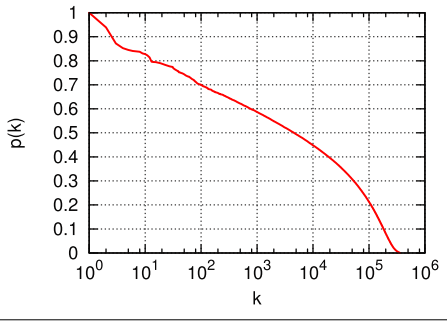


Figure 3 Cumulative product frequency distribution. The cumulative frequency distribution ($p(k)$) of products in a transactional dataset. By frequency (k) we mean the number of times a products has been put in a shopping cart. It is a concept equivalent to the support, used in the association rule mining literature.

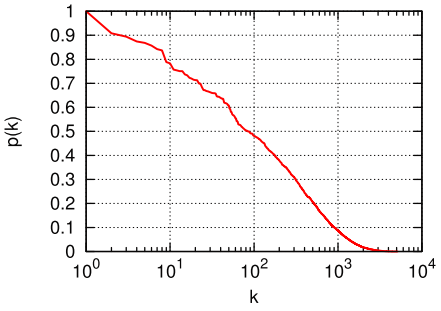
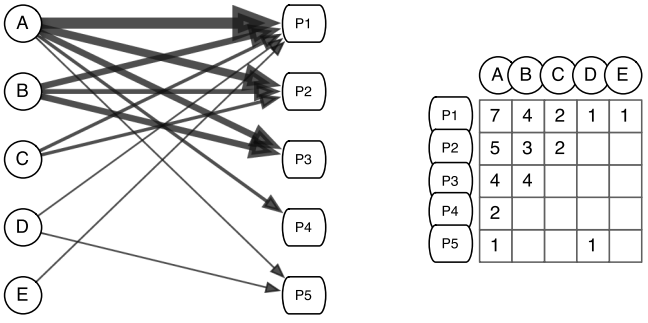


Figure 4 Cumulative customer frequency distribution. The cumulative frequency distribution ($p(k)$) of customers in a transactional dataset. k is the total number of products a customer has put in her shopping cart.



	A	B	C	D	E
P1	7	4	2	1	1
P2	5	3	2		
P3	4	4			
P4	2				
P5	1			1	

Figure 5 The bipartite purchase graph. A representation of the bipartite purchase graph, connecting customers to the product they buy. On the right, the purchase graph is represented as an adjacency matrix.

in a transactional dataset. The plots describe the probability (y axis) of a product (customer) being bought by (buying) at least a given number of customers (products). The distributions are skewed with a long tail, spanning several orders of magnitude, with 20% products bought only by at less than 10 customers, and 20% customers buying less than 10 products. As a consequence, a large amount of products is not considered in association rule mining, as these products fail to meet the frequency threshold. Moreover, the connections between the most popular products are not randomly distributed into the dataset, as they tend to be connected to the same set of customers, the ones buying everything. So, in association rule mining we only consider the products that are being bought by the same set of big buyers, ignoring all the other customer classes. A methodology able to include customers and less popular products into a global picture can be useful as a complementary part of association rule mining.

This is what we aim to do by looking at the entire transactional dataset as a complex system. Our proposal can be summarized by representing the purchases of customers and products as a weighted bipartite graph $G = (C, P, E)$, connecting a customer $c_i \in C$ to a product $p_j \in P$ she bought. The weight w on the edge $(c_i, p_j, w) \in E$ is the number of times customer c_i bought product p_j. A depiction of our model is provided in Figure 5. Our methodology aims at returning two different descriptions of the complex system of retail: the global and the local descriptors of the bipartite structure. At the global level, we generate the Purchase Function (f_*) connecting the volume of sales of products with the set of customers buying them and the volume of purchases of customers with the set of products they are buying. At the local level we perform an evaluation of how much a prod-

uct, and a customer's need, is basic or sophisticated. We call it Product (and Customer) Sophistication.

The methodology is implemented in a three-step process: (i) pre-process, where the data about customer purchases is transformed in a format suitable for our analysis; (ii) analysis, where we calculate the Purchase Function and the Product/Customer Sophistication; (iii) validation, where through a null model we evaluate the significance of the descriptors. We now proceed describing these three steps, starting from the pre-process.

3.1 Pre-process

The first step of our methodology is to pre process the connections between customers and products. This operation is carried on the adjacency matrix of the bipartite network customer-product. We are interested in showing that the best sold products are bought by all customers, while products with a low market share are bought exclusively by customers who buy everything. To highlight this pattern, we sort the matrix with the following criterion: fixing the top-left corner of the matrix M as the origin, we sort the customers on the basis of the sum of the items purchased in descending order (the top buying customer at the first row and so on), and the products with the same criteria from left to right (the best seller product at the first column and so on). In this way, at the cell $(0,0)$ we find the quantity of best seller product purchased by the top buying customer.

The final step of data preparation is to binarize the matrix, by identifying which purchases are significant and which are not. We cannot simply binarize the matrix considering the purchase presence/absence of a customer for a product. A matrix with a 1 if the customer c_j purchased the product p_i and 0 otherwise will result in a certain amount of noise: it takes only a single purchase to connect a customer to a product, even if generally the customer buys large amounts of everything else and the product is generally purchased in larger amount by every other customer.

We evaluate the meaningfulness of a purchase quantity, for each product p_i for each customer c_j, by calculating its Revealed Comparative Advantage (RCA), following [8]. Given a product p_i and a customer c_j, the RCA of the couple is defined as follows:

$$\mathrm{RCA}(p_i, c_j) = \frac{X(p_i, c_j)}{X(p_*, c_j)} \left(\frac{X(p_i, c_*)}{X(p_*, c_*)} \right)^{-1},$$

where $X(p_i, c_j)$ is the number of p_i bought by c_j, $X(p_*, c_j)$ is the total number of products bought by c_j, $X(p_i, c_*)$ is the total number of times p_i has been sold and $X(p_*, c_*)$ is the total number of products sold.

RCA takes values from 0 (when $X(p_i, c_j) = 0$, i.e. customer c_j never bought a single instance of product p_i) to $+\infty$. When $\mathrm{RCA}(p_i, c_j) = 1$, it means that $X(p_i, c_j)$ is exactly the expected value under the assumption of statistical independence, i.e. the connection between customer c_j and product p_i has the expected weight. If $\mathrm{RCA}(p_i, c_j) < 1$ it means that the customer c_j purchased the product p_i less than expected, and vice versa. Therefore, the value of 1 for the RCA indicator is a reasonable threshold to discern the meaningfulness of the quantity purchased: if it is strictly higher, then the purchases are meaningful and the corresponding cell in the binary matrix is 1; otherwise the purchases are not meaningful, even if some purchases are actually made, and the corresponding cell in the binary matrix

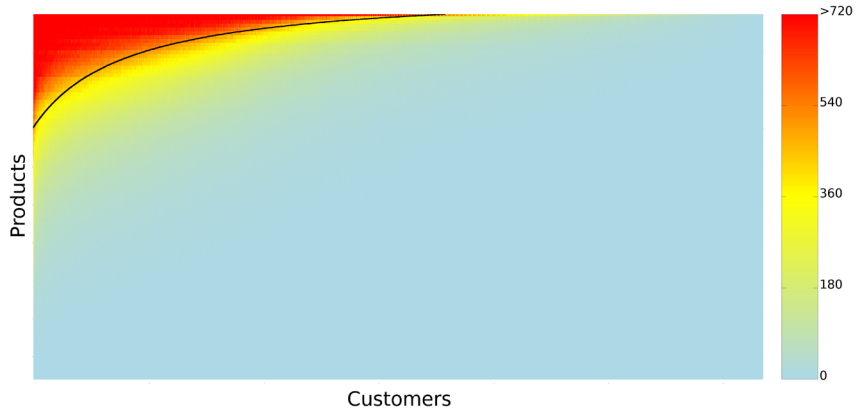


Figure 6 The M_{cp} purchase matrix. For layout purposes, the matrix has been transposed, thus we have customers as columns and products as rows. In black we report the isocline.

is 0. The M_{cp} matrix is built accordingly to this rule:

$$M_{cp} = \begin{cases} 1 & \text{if } RCA(p_i, c_j) > 1; \\ 0 & \text{otherwise.} \end{cases}$$

This is the final output of the preprocess phase, hence from now on it will be referred as the purchase matrix M_{cp}, and $M_{cp}(c_j, p_i)$ is the entry of M_{cp} of row j and column i.

The M_{cp} purchase matrix for the Livorno2007-2009 dataset, result of the pre-process phase, is depicted in Figure 6. In Figure 6, the columns of the matrix are the 317,269 customers and the rows are the 4,817 products. We depicted a compressed view of the matrix, where each data dot represent a 50×50 square of the original matrix and the gradient represents how many 1's are present in that section of the matrix, for space constraints.

We can observe in Figure 6 the phenomenon we expect given our assumptions: only a small amount of popular products are bought by everyone, but smaller sets of customers purchase the rest of the products (going from the right to the left columns). The same set of big buyers are always part of these smaller and smaller sets.

Livorno2010-2011 and Lazio2007-2009 matrices are depicted in Figures 7 and 8, left and right respectively (the legend for both figures is the same as Figure 6 legend). From Figures 7 and 8 we can see that the triangularity of the M_{cp} matrix is constant, regardless the geographical and/or temporal selection of the data.

3.2 Analysis

In this phase, the aim is to obtain the global and local descriptors of the complex system of retail. For the global level, we define the function f_* connecting the volume of sales of products with the set of customers buying them and the volume of purchases of customers with the set of products they are buying. At the local level we perform an evaluation of how much a product, and a customer's need, is basic or sophisticated. We start with the global level and then we proceed describing the local level.

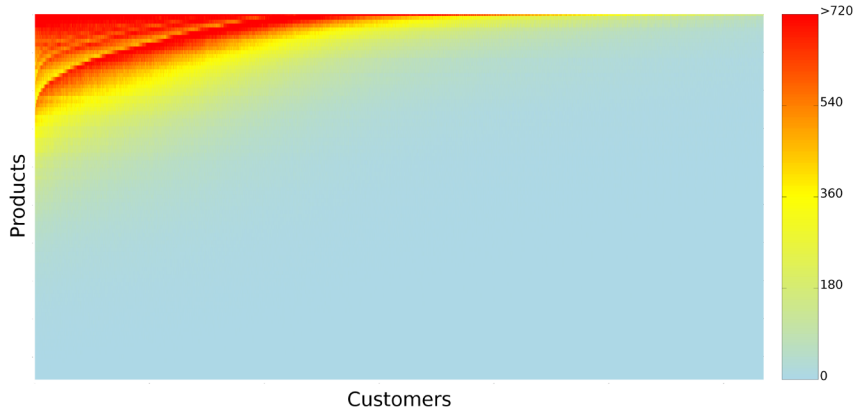


Figure 7 Instance of Livorno purchase matrix. The M_{cp} matrix for Livorno2010-2011.

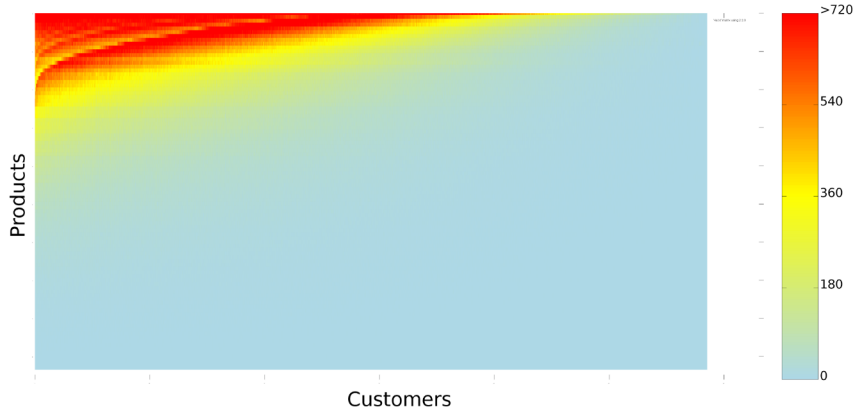


Figure 8 Instance of Lazio purchase matrix. The M_{cp} matrix for Lazio2007-2009.

3.2.1 Global descriptor

Customer behaviour is not random: as we have seen there are many studies dealing with the problem of finding correlations between products frequently bought together [1]. However, here we strengthen this assumption as follows: these correlations are actually organized following a general function that regulates retail purchases. In other words, we are not dealing with a set of correlations limiting their effects on two or three products. There exists a general pattern, meaning that it is possible to define the set of products bought by a customer as a function of the amount of products she buys. We call this the Purchase Function.

The Purchase Function states that the assortment of products bought by any given customer c_j is determined by c_j's volume of purchase, and the population of customers that buy any given product p_i is determined by p_i's volume of sales. More precisely, we indicate it as a f_* function, that relates the *rank* of products with the *rank* of customers, where the rank i of a product p_i (or j for customer c_j) stands for the fact that p_i is the ith highest sold product (or c_j is the jth customer with the largest volume of purchases). In practice, looking at the $M_{c,p}$ matrix in Figure 6, the f_* function is the equation of the line dividing the area with the high density of ones (colored in red) from the rest.

For any customer c_j we denote assortment(c_j) = $\{p_1, \ldots, p_{f_*(j)}\}$ and, for any product p_i, customer_base(p_i) = $\{c_1, \ldots, c_{f_*^{-1}(i)}\}$. The Purchase Function is assumed to be a decreasing monotonic function, i.e., $i_1 < i_2$ implies that $f_*(i_1) > f_*(i_2)$, which in turn implies that assortment(c_2) $\subseteq$ assortment(c_1). In other words, if c_1 is a customer purchasing more in terms of product quantities than c_2, then it is very likely that c_1 buys the same set of products c_2 buys, plus something more.

Matrices with triangular structures have been already studied in ecology literature. In ecosystems, simpler organisms are ubiquitous and more complex organisms appear *iff* simpler organisms are already present [13]. In these works, authors define nestedness as a measure to understand how much triangular is the structure of the matrix representing the connections between species and ecosystems. The nestedness is calculated by identifying the border dividing the matrix in two areas containing respectively most ones and most zeroes, that is exactly the role of the Purchase Function. In this literature, this is known as *isocline*.

In literature there are several algorithms tackling the problem of computing the isocline of a matrix [14]. The general approach is usually made in two steps: a reordering of the rows and columns of the matrix, such that the ones tend to be clustered in the upper-left corner of the matrix; and an estimate of the isocline function on the reordered matrix.

In our framework we are implementing an alternative way to calculate the Purchase Function (isocline). We have chosen to do so for two reasons. First, all algorithms explicitly reorder the matrix. We do not want to reorder our matrix, since the order we defined in the pre-process is a fundamental prerequisite for the purchase function, as it has been defined above to connect the ranks of customers and products calculated on their volumes of purchases and sales, respectively. These ranks are obtained by the matrix ordering during the preprocessing stage and thus this order cannot be modified. Secondly, the state-of-the-art algorithms are designed to deal with ecology data, with a number of cells in the order of 10^4 or 10^5. Since our cells are $\sim 10^9$, we need to define a new procedure, enabling the application of our framework to large datasets. We described the specifications of this methodology in a previously published technical report [15].

We need an evaluation measure to understand if a proposed isocline is good or not. We use the following formulation:

$$N(M_{\mathrm{cp}}, f_*) = \frac{1}{2}\left(\frac{f_l(M_{\mathrm{cp}}, 1)}{f_l(M_{\mathrm{cp}}, *)} + \frac{f_r(M_{\mathrm{cp}}, 0)}{f_r(M_{\mathrm{cp}}, *)}\right),$$

where $f_l(M_{\mathrm{cp}}, *)$ counts the number of cells at the left of the isocline in M_{cp} where we expect to find the ones (and $f_l(M_{\mathrm{cp}}, 1)$ counts the ones) and where $f_r(M_{\mathrm{cp}}, *)$ counts the number of cells at the right of the isocline in M_{cp} where we expect to find the zeroes (and $f_r(M_{\mathrm{cp}}, 0)$ counts the zeroes). In practice, we take the average of the one-density at the left and zero-density at the right of the isocline. We used this measure because simply counting unexpected presences and absences of ones at the right/left of the isocline is not a fair measure, being our matrix very sparse.

We now need to find the isocline. To find it, we estimate where the isocline should pass to maximize the division of ones at the left and zeroes at the right. We consider our matrix as a Cartesian space. For each discrete x axis value (customer) we get an estimate of where the isocline should pass (y axis). We do so by summing the ones of the corresponding matrix row ($k_{c,0} = \sum_p M_{\mathrm{cp}}(c, p)$). Then, for each discrete y axis value (product) we get

Table 2 The $N(M_{cp}, f_*)$ for the different f_* shapes tested

f_*	$N(M_{cp}, f_*)$
$ax + b$	0.616106811666
$ax^2 + bx + c$	0.628533747603
$a\log(x) + b$	0.623911138356
ax^b	0.572996769269
$\frac{a}{x}$	0.609588181022
$-\frac{ax+b}{cx+d}$	0.632547410976

Table 3 The $N(M_{cp}, f_*)$ for the different views of the dataset

M_{cp}	$N(M_{cp}, f_*)$
Livorno2007-2009	0.632547410976
Lazio2007-2009	0.622983174602
Livorno2010-2011	0.615276275848
Null model average	0.5892564877

an estimate of where the isocline should pass (x axis). We do so by summing the ones of the corresponding matrix column ($k_{0,p} = \sum_c M_{cp}(c,p)$). We average these two values and we obtain a pair of coordinates. This procedure is linear in the number of customers and products and therefore it can scale with very big matrices. We fit these coordinates using a non-linear least squares optimization with the Levenberg-Marquardt algorithm [16] to obtain the best function able to represent the isocline and, therefore, the Purchase Function.

To fit a function with the non-linear least squares optimization, the shape of the function is needed. Our framework tries several different shapes, storing the corresponding $N(M_{cp},f_*)$ value and then choosing the best performing one. In our case, we obtained a simple hyperbola in all the three cases in exam. The $N(M_{cp},f_*)$ results of different f_* formulations for the Livorno2007-2009 matrix are reported in Table 2. The evaluation via the $N(M_{cp},f_*)$ function of the goodness of the division operated by the isocline for all datasets and the null model is provided in Table 3. The null model average reported in the last row of Table 3 has been calculated averaging 30 iterations. As we can see, the average value for null model is lower. Its standard deviation is in the order of 10^{-5}. Therefore, we can conclude that the difference is also significant.

For the Livorno2007-2009 dataset, the value of the f_* parameters has been estimated as: $\alpha = 11{,}318.559$, $\beta = 94.2526$, $\gamma = 0.2834$, $\delta = -16{,}866{,}558$. The corresponding isocline has been plotted in black in Figure 6. We do not report the values of the parameters for the other datasets as we are not using them in the rest of the paper.

3.2.2 Local descriptor

As for the local descriptor, we quantify the sophistication level of the products sold and of the needs of the customers buying products. The basic intuition is that more sophisticated products are by definition less needed, as they are expression of a more complex need. One may be tempted to answer to this question by trivially returning the products in descending order of their popularity: the more a product is sold, the more basic it is. However, this is not considering an important aspect of the problem: to be sold to a large set of costumers is a condition to be considered 'basic', but it does not fully describe the term. Another condition is that the set of customers buying the product should include

the set of costumers with the lowest level of sophistication of their needs. The conjunction of the two properties is closer to define a product as 'basic'.

This conjunction is not trivial and it is made possible by the triangular structure of the adjacency matrix. Consider Figure 6: the columns in the right part of the matrix are those customers buying only few products. Those products are more or less bought by everyone. In a world where our theory does not hold, instead of buying the products at the top row of the matrix they would buy random products.

For this reason, we need to evaluate at the same time the level of sophistication of a product and of the needs of a customer using the data in the purchase matrix, and recursively correct the one with the other. We adapt the procedure of [17], adjusting it for our big data.

We calculated the sums of the purchase matrix for each customer ($k_{c,0} = \sum_p M_{\mathrm{cp}}(c,p)$) and product ($k_{0,p} = \sum_c M_{\mathrm{cp}}(c,p)$) to estimate the isocline for the Purchase Function f_*. To generate a more accurate measure of the sophistication of a product we need to correct the sums recursively: this requires us to calculate the average level of sophistication of the customers' needs by looking at the average sophistication of the products that they buy, and then use it to update the average sophistication of these products, and so forth. This can be expressed as follows: $k_{N,p} = \frac{1}{k_{0,p}} \sum_c M_{\mathrm{cp}} k_{c,N-1}$. We then insert $k_{c,N-1}$ into $k_{N,p}$ obtaining:

$$k_{N,p} = \frac{1}{k_{0,p}} \sum_c M_{\mathrm{cp}} \frac{1}{k_{c,0}} \sum_{p'} M_{\mathrm{cp}'} k_{N-2,p'},$$

$$k_{N,p} = \sum_{p'} k_{N-2,p'} \sum_c \frac{M_{\mathrm{cp}} M_{\mathrm{cp}'}}{k_{0,p} k_{c,0}}$$

and rewrite this as:

$$k_{N,p} = \sum_{p'} \widetilde{M}_{pp'} k_{N-2,p'},$$

where:

$$\widetilde{M}_{pp'} = \sum_c \frac{M_{\mathrm{cp}} M_{\mathrm{cp}'}}{k_{0,p} k_{c,0}}.$$

We note in the last formulation $k_{N,p}$ is satisfied when $k_{N,p} = k_{N-2,p}$ and this is equal to a certain constant a. This is the eigenvector which is associated with the largest eigenvalue (that is equal to one).[c] Since this eigenvector is a vector composed by the same constant, it is not informative. We look, instead, for the eigenvector associated with the second largest eigenvalue. This is the eigenvector associated with the variance in the system and thus it is the correct estimate of product sophistication.

However, this formulation is very sensitive to noise, i.e. products that are bought only by a very narrow set of customers. To calculate the eigenvector on the entire set of products generates a small amount of products whose sophistication level is seven orders of magnitude larger than the rest of the products. This variance provokes the other sophistication estimates to be flattened down to the same values and therefore not meaningful. However, we do not want to simply cut the least sold products, as we aim to create a full

product hierarchy, including the least sold products. To normalize this, we employ a three step strategy. First, we calculate the eigenvector on a restricted number of more popular products, purchased by at least a given threshold δ of customers. Then we use the estimate of the sophistication of these products to estimate the sophistication of the entire set of customers (that is, as defined before, the average sophistication of the restricted set of products they buy). Finally, we use the estimated sophistication of the customers to have the final sophistication of the entire set of products, again by averaging the sophistication of the customers buying them. Hence, we define the product sophistication index (*PS*) as:

$$PS = -\frac{\vec{K} - \mu(\vec{K})}{\sigma(\vec{K})},$$

where $\vec{K}$ is the eigenvector of $\widetilde{M}_{pp'}$ associated to the second largest eigenvalue, normalized as described above; $\mu(\vec{K})$ is its average and $\sigma(\vec{K})$ its standard deviation. The Customer Sophistication *CS* is calculated using the very same procedure, by estimating $k_{c,N}$ instead of $k_{N,p}$.

Notice that there are alternatives for the computation of the sophistication measure. One among the most popular is [18], where the product complexity is formulated in terms of 'country fitness'. Instead of defining product complexity as the sum of the complexities of the countries producing the product, the authors of [18] use the inverse of the sum of the inverse complexities:

$$k_{N,p} = \frac{1}{\sum_c M_{c,p} \frac{1}{k_{c,N-1}}}.$$

The aim is to maximize the impact of countries with low complexity in dragging down the complexity of the products made mostly by them. There are upsides and downsides to each measuring choices, and this case is not different. However, the measure proposed in [18] is highly correlated with our choice, as shown in [19]. Therefore, in the context of this paper, there is no reason to prefer one measure over the other, and we make the choice of using only one for clarity and readability.

As example of the Product and Customer Sophistication calculation, we report the most and least sophisticated products for the Livorno2007-2009 dataset. We do not report the Customer Sophistication for privacy concerns. In Table 4 we report a selection of the least sophisticated products, i.e. the ones with the lowest *PS* values, in the purchase matrix. The less sophisticated products should be intuitively the ones covering the most basic human needs, and this intuition is confirmed by the reported products: bread, water, fruits and milk. On the other hand, Table 5 reports the most sophisticated products, i.e. the ones with the largest *PS* values, that intuitively should be products satisfying high-level non-necessary, probably luxury, needs. In fact, what we find in Table 5 are hi-tech products (LCD televisions, DVD compilations, computer accessories), jewellery and very specific clothing. Note that these results only apply to the particular time and location studied here. Different cultures and different countries with different economic levels can only be described by collecting appropriate additional data.

3.3 Validation

The triangular structure of the matrix in Figure 6 gives an important information: a customer that purchased few products is expected to have bought just products that are best

Table 4 A selection of the more basic products according to their *PS* values

p_i	*PS*
Regular bread	–4.41
Natural still water	–4.19
Yellow nectarines (peaches)	–3.84
Semi-skimmed fresh milk	–3.81
Bananas	–3.53

Table 5 A selection of the more sophisticated products according to their *PS* values

p_i	*PS*
LCD 28″/30″ televisions	2.91
DVD music compilations	2.86
Sauna clothing	2.66
Jewelry bracelets	2.53
RAM memories	2.33

sellers. This disagrees with the expected presence of 'cherry pickers', i.e. customers that are particularly sensible and responsive to sales, especially if the sales are placed on expensive goods. Instead, looking at Figure 6, we expect customers to follow a general pattern.

Starting from this consideration, we need to validate the model, in particular we want to control that the triangular structure is meaningful. We need a null model definition with which to compare our theory. We identify three important features that our null model must hold: (1) the purchases are distributed randomly; (2) customers must preserve the total amount of their purchases; and (3) each product must preserve its sale volume on the market. The implementation of the null model is reported in the Appendix.

We depict the null matrix for Livorno2007-2009 in Figure 9, that is an accurate depiction also of the typical null matrix for Livorno2010-2011 and Lazio2007-2009. We can see that Figure 9 still presents some of the characteristic of the original M_{cp} matrix. However, in Figure 9 popular customers/products tend to have randomly distributed RCAs (therefore their columns/rows appear white in the compressed view) and, while preserving some triangularity, the null model matrix have a tendency to display more ones on the main diagonal than the original M_{cp} matrix. We can conclude that the null hypothesis, i.e. the

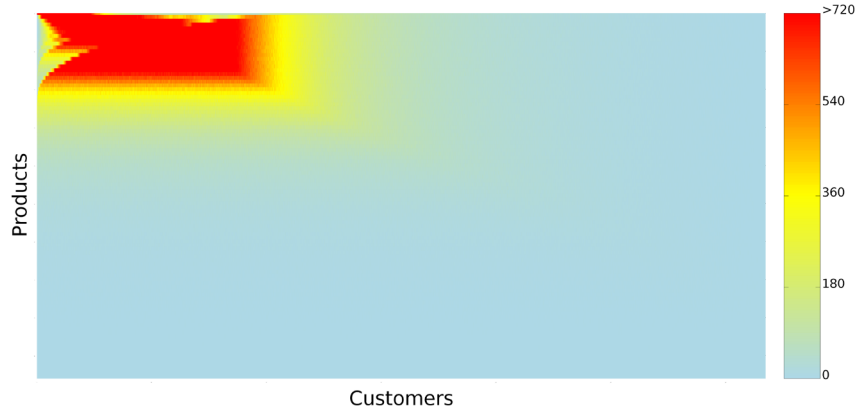


Figure 9 One instance of the null purchase matrix. To be consistent with Figure 6, also the null matrix has been transposed.

simple distribution of volume of sales and of purchases of products/customers, explain only part of the observed structure, but the original M_{cp} matrix presents some characteristics that cannot be generated randomly just by the distributions of volume of sales for the products and volume of purchases for the customers.

4 Results

In the previous section we have defined our methodology to extract the general pattern governing customer behaviour, by analysing the adjacency matrix of the bipartite structure connecting the customers to the products they are buying. In this section, we apply our methodology to real world data. We firstly describe the nature of our data. We then move on to describe the data selection policy. Then, we apply the framework, obtaining the global descriptor in the form of the Purchase Function, and the local descriptor, i.e. the sophistication levels of customers and products. Finally, we provide our three analyses: the empirical observation of Maslow's hierarchy of needs [12]; the marketing application for targeted advertising; and finally the evaluation of predictability of customer movements on the territory.

4.1 Data-driven hierarchy of needs

In this section we want to use the information provided by the Product Sophistication index to reconstruct the hierarchy of needs of the supermarket customers, and therefore provide an empirical observation of the theory of Maslow [12].

Three caveats need to be specified. First: we are not claiming that this hierarchy of needs is universal. The result we are presenting in this paper has been reached with data from one city of Italy (Livorno) and therefore it describes the hierarchy of needs of that particular city. However we showed that the triangular structure of the purchase matrix is present even in different areas of Italy (Figures 7 and 8) and therefore our framework could be applied to different world regions, helping to create a picture of different hierarchies of needs. The comparison of hierarchies of needs of different cities and the evaluation of different cultural perspectives of customers over their needs is left as future development. Further, this hierarchy is a valuable marketing tool for that particular city: products at the basis of the hierarchy are more needed, thus no marketing strategies are required for them as they will be sold anyway.

The second caveat is that we built the hierarchy of needs using the product category classification defined by the supermarket owners. To use this classification introduces the bias of a set of people, with a given culture and marketing aims. We plan to use for future developments standard product classifications.

Lastly, a collection of customers could be buying some classes of products in different shops, thus unfairly pushing up their sophistication. While this effect is considered to be small due to the high customer fidelity and the all around service provided by the supermarket, some of the products at the top of our hierarchy of needs could be over-represented.

With this caution in mind, we now build the hierarchy. To build the hierarchy we need to divide products in classes according to their PS value. Formally, we need to segment the PS values, previously sorted. We decided to perform a one-dimensional clustering using the *ck*-means algorithm. *ck*-means is an evolution of the k-means algorithm which guarantees the optimality of clustering [20]. The k-means problem is to partition data into k groups

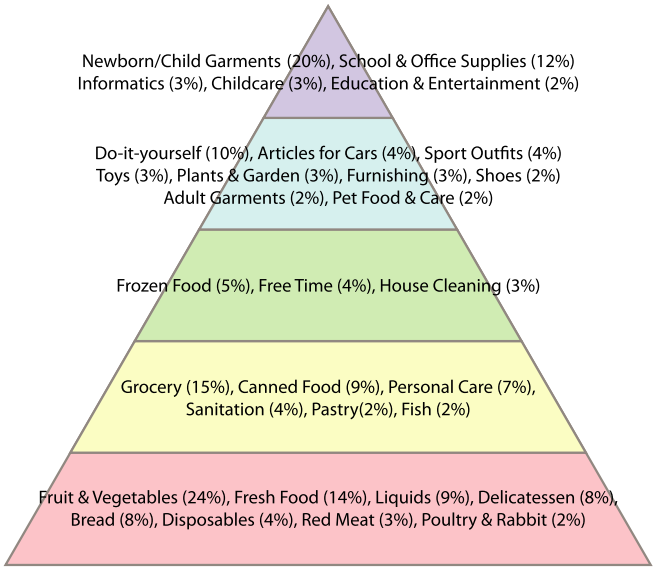


Figure 10 The pyramid of needs. Our data-driven Pyramid of Needs with the most basic products at the bottom and the most sophisticated at the top. For each product category we report its share among all purchases at that level of the hierarchy.

such that the sum of squared Euclidean distances to each group mean is minimized. *ck*-means is optimized to operate on one dimensional data, which is our application setting. In this setting *ck*-means find the optimal cluster separation, which is unique and therefore a repeatable result, properties that standard k-means does not hold.

We set $k = 5$, as we follow Maslow's hierarchy of needs classification [12] and we want to obtain roughly the following classes of products: fundamental for survival, basic needs, complementary needs, accessory needs and luxury needs. The results of the *ck*-means clustering have been depicted in Figure 10. In Figure 10 for each level of the hierarchy we report its main composition according to the product categories. The share values between the parenthesis tell, given the total amount of products purchases at that level of the hierarchy, how many of those belong to that particular category. For instance, skimmed and semi-skimmed milk may belong to two different hierarchy levels, say 0 and 1, and they have respectively been sold 4,000 and 2,000 times. Let us say that the total amount of products sold in hierarchy levels 0 and 1 are respectively 4,000,000 and 1,500,000. Then, skimmed milk contributes to level 0 as 0.1%, while semi-skimmed milk contributes to level 1 as 0.13%. We report only categories representing at least 2% of the hierarchy level. We did not report the single product segment, as they are too specific and too many: for instance apples, pears, bananas, tomatoes, potatoes and so on have been aggregate in the product category 'Fruits & Vegetables'. Of course, products in the same category may fall in different hierarchy levels: in Figure 10 we chose to put the category where it occupies the largest share of the level purchases.

Figure 10 is clear depiction of what are the priorities in the mind of the customers of Livorno. Figure 10 is telling some expected and some unexpected things. First there are the basic needs: drinking and eating, particularly fruit, vegetables, bread and meat. Then, there are more sophisticated eating products and what is needed to take care of the body hygiene. At the middle of the hierarchy we start to have product not strictly necessary

for survival: house cleaning and simple products for the free time. The two most sophisticated needs are schooling, entertainment (both for children and adults), more complex garnishment; and, climbing at the top of the pyramid, newborn childcare and unnecessary equipment. The basis of the pyramid is expected: most basic needs are food and personal hygiene. Up until now we have basic confirmation about human needs. The top of the pyramid is instead telling us something surprising. Traditionally, reproduction is considered one of the most basic needs of any living thing. However, what we see is that in our modern society to have a baby ends up being one among the most sophisticated needs, and the first one to be dropped, even before having a pet.

4.2 Data-driven marketing insights

We now describe a possible targeted marketing strategy based on the outputs of our framework. Suppose the supermarket wants to promote a product p_i and it wants to limit its target to the smallest subset with the highest probability of buying the product advertised. The Purchase Function f_* can be used in the following way: given the amount of products bought by customer c_j we use its index j to obtain the index $f_*(j) = i$ of the most sophisticated product p_i that c_j is buying. With this information, we can derive the set of products she is expected to buy, that is assortment(c_j). assortment(c_j) is defined as all the products that have an index $i' \leq i$. The same applies considering as input a product p_i, we obtain the index delimiting the set of customers buying it (for which $j' \leq f_*^{-1}(i)$).

One concern needs to be addressed before continuing: how well is the Purchase Function dividing the ones from the zeros in comparison to what we expect? How much is a customer more likely to buy a product following the Purchase Function evaluated on our real world data (P_f) over any random product (P)?

As previously reported, the Livorno2007-2009 $M_{c,p}$ matrix contains ∼37 millions ones out of ∼1.5 billions cells. This means that, given a random product p_i and a random customer c_j, the baseline probability $P(p_i, c_j)$ that customer c_j is buying product p_i in a significant amount (i.e. RCA$(c_j, p_i) > 1$) is the ratio of these two numbers, or $P(p_i, c_j) = 2.44\%$. If we consider only the portion of the matrix at the left of the calculated isocline, i.e. the area of the matrix for which f_* tells us that the customers are very likely to buy exactly that products, we count 16,748,048 ones and 60,025,000 total cells. Thus, the probability $P_f(p_i, c_j)$ for a customer c_j to buy significant amounts of a product p_i for which $i \leq -\frac{\alpha j+\delta}{\gamma j+\beta}$ (i.e. $p_i \in$ assortment(c_j)) is 27.9%. Using the Purchase Function f_*, we can narrow of two orders of magnitude the set of combinations of products and customers to analyse and still capturing almost half of the significant purchases. In other words, customers are 11.43 times more likely to buy a product p_i if i is lower than, or equal to, the index limit predicted by the Purchase Function. We refer to this ratio as $\frac{P_f(p_i,c_j)}{P(p_i,c_j)}$, i.e. the f_*-based probability of connecting customer c_j with product p_i over the baseline probability. We also calculated the same ratio, this time by counting at the right side of the isocline, where we expect to find many zeros. The number of ones is 37 millions minus 16 millions, and it is divided by the number of cells, 1.5 billions minus 60 millions. The probability of obtaining a one is 1.39%, less than one twentieth of the left side of the isocline.

Now that we have addressed the main concern about the Purchase Function, we can safely assign to product p_i a corresponding customer index $j = -\frac{\beta i+\delta}{\gamma i+\alpha}$ that is its current 'border'. All indexes $j' \leq j$ represents customers who buy product p_i (i.e. $\forall j' \leq j$, $c_{j'} \in$ customer_base(p_i)), while the indexes $j'' > j$ are customers not buying p_i. By definition, the higher the value of j'', the more unlikely is the customer buying p_i. Thus, the set

Table 6 The probabilities of buying product p_i in general ($P(p_i)$) and given that a customer already buys product p_{i-1} ($P(p_i|p_{i-1})$)

p_i	p_{i-1}	$P(p_i)$	$P(p_i\|p_{i-1})$
Dishwasher salt	Dishwasher soap	8.39%	30.41%
Asparagus	Olive	8.00%	26.12%
Peppers	Chicory	7.31%	23.73%
Canned soup	Preserved anchovies	9.96%	32.23%
Wafers	Sugar candies	11.30%	21.67%

Table 7 The comparison between the size of the target customer sets identified by the Purchase Function against random target customer sets with the same number of customers likely to buy p_i

p_i	$\|TC^*\|$	$\|TC\|$	$\frac{\|TC_r\|}{\|TC\|}$
Tomino cheese	58	137	7.51095
Raw ham	78	144	5.81250
Apricot jam	66	127	4.66142
Anchovies	83	144	4.06250

of customers the law is suggesting to target is the one immediately after index j. Since f_* is an interpolation, it is safe to define a threshold ϵ_1. Then, we define the set TC, the target customers set, as the set of all customers for which, given their index j', it holds: $j - \epsilon_1 \leq j' \leq j + \epsilon_1$ and $M_{\text{cp}}(c_j, p_i) \neq 1$ (the last condition is necessary to exclude from TC all customers who are already buying large quantities of product p_i, as it is useless to advertise p_i to them).

To evaluate how many elements of TC are likely to start buying p_i, we remark that having a 1 in the product of index $i-1$ makes the customer very likely to buy the next more sophisticated product p_i. In other words, to have purchased large amounts of the product immediately to the left in the matrix to p_i increase to probability of purchase this product. For instance, customers buying 'dishwasher soap' have 30.41% probability of buying product 'dishwasher salt' against a baseline probability of 8.39%, some instances of this are provided in Table 6. On average, the $\frac{P(p_i|p_{i-1})}{P(p_i)}$ ratio is 1.993 for the 500 most sold products, and no single product has a ratio lower than 1 (the lowest is 1.05 for Fresh Bread). Therefore, $\forall tc \in TC$ we check if $\exists x$, $M_{\text{cp}}(tc, p_x) = 1$, with $i - \epsilon_2 \leq x < i$, thus looking not only at the direct left neighbor of product p_i, but at his ϵ_2 left neighbors. If the condition holds, we have identified TC^* as the subset of TC composed by those customers who are likely to buy p_i.

The question now is: how large should be a TC_r set to obtain an equally large TC_r^* set if TC_r has been populated without knowledge about the Purchase Function, i.e. at random by picking customers who are not already buying product p_i? We address this question by looking at several different products. For each of them we identified the TC set using f_* and then we calculated 500 random TC_r sets. In Table 7 we report, for each product p_i, the following statistics: the number of customers likely to purchase p_i ($|TC^*|$ column), the total number of targeted customers ($|TC|$ column) and the average ratio between the targeted customers without and with using the purchase function f_* ($\frac{|TC_r|}{|TC|}$), by fixing $\epsilon_1 = 100$ and $\epsilon_2 = 2$. As we can see, the knowledge provided by the Purchase Function reduces the number of customers to be targeted by a marketing campaign by four or more times, with the same return of investment (as our procedure fixes $|TC^*| = |TC_r^*|$). Table 7 reports only a few products, but we tested these 500 random sets for 800 different products and

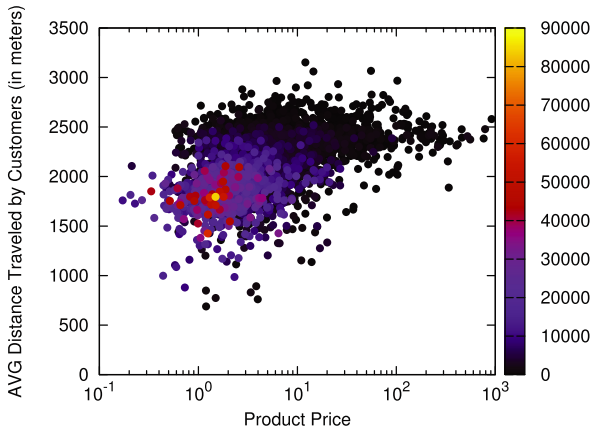


Figure 11 Price-distance relationship. Average distance traveled to get a product with a given price.

the average of the averages of the $\frac{|TC_r|}{|TC|}$ ratio is 3.55594, i.e. on average using the Purchase Function the marketing campaign can target three times less customers with the same gross return. For none of the 800 products the average of the ratio was less than 1.

4.3 Predicting customer mobility

To explain customer mobility is one of the successes of this framework. The full study of the application has been published in [21], without the framework formalization. We report here the results to prove the usefulness of this framework. Customer mobility has been shown to be rather predictable on long time scales [22]. In [22], authors show that it is possible to model the overall mobility behavior of customers. More than showing the predictability of customer movements as in [22], we focus in one of the possible causes of it.

We assume that customers modify their shopping behaviour according to their relative position to the shop they are going to. A customer may decide to buy or not buy a given product because it is close enough or too far away from the shop. We expect that customers will travel more to purchase products that are more expensive, for many possible reasons. For example, a larger money investment makes less important the amount of time spent in doing it. We check this hypothesis by plotting for each purchase the price of an item against the average distance that a customer travelled to get the product. This plot is depicted in Figure 11: the price is on the x axis (in logarithmic scale), while the distance travelled is on the y axis. The price is recorded in Euros. Each dot is a purchase and we color it accordingly to how many purchases are represented by the same price and by the same distance.

The connection of a customer to a product is created with the procedure described in the pre-process section, therefore we are only considering connections generated when the quantity of product p bought by customer c_i is significant. A customer c_i may have bought product p_j in different shops, say s_1, s_2, s_3, s_4. In this case, we weigh each distance travelled with the amount of purchases made using the following formula:

$$d(c_i, p_j) = \sum_{\forall s \in S} \frac{p_j(c_i, s) \times d(c_i, s)}{p_j(c_i, *)},$$

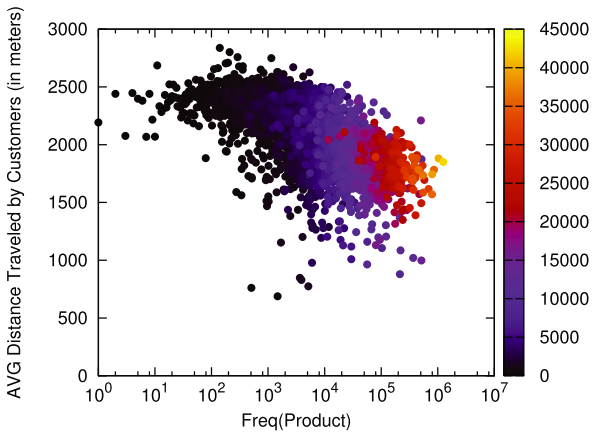


Figure 12 Quantity-distance relationship. Average distance traveled to get a product with a given popularity.

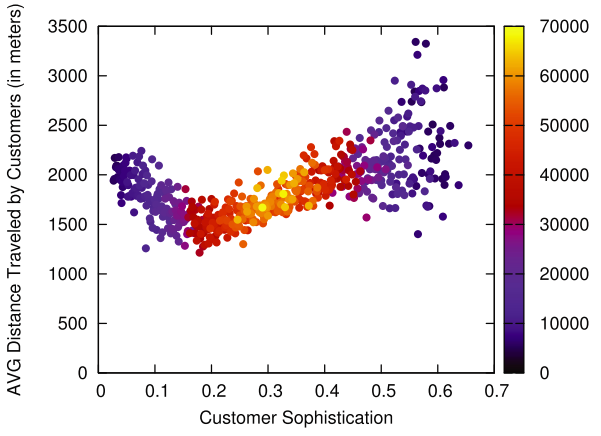


Figure 13 Product sophistication-distance relationship. Average distance traveled to get a product with a given sophistication index.

where S is the set of all shops, $d(c_i, s)$ is the distance between customer c_i and shop s, $p_j(c_i, s)$ and $p_j(c_i, *)$ are the amount of purchases of product p_j made by customer c_i in shop s and in general, respectively. This procedure has been followed for the plots depicted in Figures 11, 12, 13 and 14.

Products with the same price are bought by customers placed at different distances from the shop. Given a price, we average the distance travelled by the customers buying the products with that exact price. By averaging, we lose the ability of describing each single customer and we just describe the behaviour of the system in its entirety. We do so because the single customer is bounded by the place where she lives, thus each single customer carries a noisy information, and we can make sense of it only by looking at the global level.

From Figure 11 we can conclude that price plays a role in driving customer decisions of travelling a given distance for a product. The correlation here looks weak, but positive: customers travel more if they need to buy a more expensive product. We calculate a log-linear regression[d] using the function $f(x) = a \log x + b$. In this regression, $R^2 = 17.25\%$ ($r =$

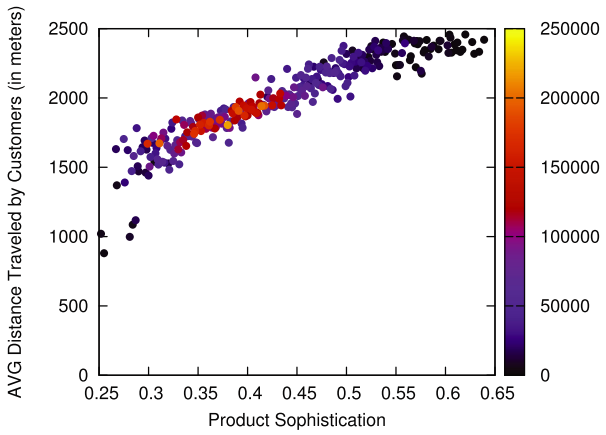


Figure 14 Customer sophistication-distance relationship. Average distance traveled by a customer with a given sophistication index.

0.4154, with p-value < 0.01), meaning that we can explain 17.25% of the variance in the distance travelled using the price.

To check if the frequency of purchase can explain the distance travelled by customers, we repeated the same analysis, using the number of purchases of a product instead of the price. We depicted the plot in Figure 12. The correlation here is negative: the more frequently a product needs to be bought, the smaller the distance a customer will travel for it. We calculate a regression with the function $f(x) = a \log x + b$ and we obtained $R^2 =$ 32.38% ($r = -0.5691$, with p-value < 0.01).

These tests confirm that the price plays a small role in predicting the distance a customer will travel for purchasing a product, by increasing it. If a product is needed more frequently then it drives (down) the distance a customer will travel to buy it, regardless of the price. However, there is a large amount of variance that remains unexplained.

We propose that our Product and Customer Sophistication indexes have, in this case, higher explanatory power. The intuition is that if a product satisfies a more sophisticated need (and the customer has those needs) then the customer is willing to travel farther to purchase the product. To test this hypothesis, we generate the same plots created for price and frequency of purchase, using our computed indexes. The plots are depicted in Figures 13 and 14.

In Figure 13, we test the relationship between the distance travelled and the Customer Sophistication: we calculate the average distance travelled by customers (y axis) to get to the shop against their sophistication value (x axis). In this case, the x axis has not a logarithmic scale, as the relationship is linear. We can see that the relationship between distance travelled and customer sophistication looks non-linear. From a value of sophistication of 0 to around 0.2 the relationship is negative, while it is clearly positive afterwards. We speculate that this effect could be driven by the fact that customers with lower sophistication could live on average further from the shops for many reasons (they prefer living outside the city, they are in poorer areas of the city, etc.). However, to test this speculation is outside the scope of this paper and we leave it as future work.

For this reason, we move on in depicting the Product Sophistication (x axis) against the average distance travelled by the customers to purchase the given product (y axis) in Figure 14. In this case, the relationship is clear: the more a product is sophisticated, the more

customers will travel to buy them. The product sophistication has a normal distribution, but less sophisticated products are more sold, given the triangular shape of the matrix. This fact explains why most of the data points are in the left part of the plot: most purchases are generated for low sophistication products. We calculated a linear regression, for which $R^2 = 85.72\%$ ($r = 0.9259$, with p-value < 0.01). This R^2 is more than twice higher than the R^2 obtained with the purchase frequency, explaining much better the variance in the distances travelled by customer.

In [21] we address possible objections such as the influence between distance and number of products bought, which may invalidate the effect of the Product Sophistication. We also show that the average Sophistication of different shop types (we recall that there are three, in decreasing order of size and Sophistication: iper, super and gestin) influence the average distance of their customers. For compactness, we point to that paper for this additional material and we conclude this section by remarking that the average sophistication of the products in a shop is influencing customers' decisions: when they need a more sophisticated product they are prone to decide to go to a larger shop with higher sophistication.

5 Discussion

In this section we firstly place this paper in the context of marketing research literature, especially in the field of data mining. We then briefly review the strong and weak points of this study. Finally, we conclude the paper, summing up contribution and future works.

5.1 Results in context with previous literature

This work is a complementary approach to the classical data mining task of the association rule mining. In data mining, association rule mining is a tool developed to find correlations between the appearances of products in shopping carts [1]. Association rule algorithms are able to uncover the most frequent and interesting rules by efficiently cutting the search space (or even without [23]). Recently, many step forwards have been proposed in association rule mining as mining multidimensional rules [24]. Our work differs from the ones presented as it is not focused on finding all the particular rules in a transactional dataset, but in exploring the general pattern characterizing it as a whole. This pattern can also be used to design better heuristics for the classical association rule mining algorithm, since it unveils novel relationships among products.

There are also works that aim to use association rule mining to obtain a general picture of the system [25]. However, also in this case our approach is different. In [25], only the associations between products are considered, leaving the customers undescribed. Then, the general picture in [25] is based on the aggregation of the local patterns, while in our work we employ a complementary approach, creating the general picture by analysing the entire set of transactions as a complex system, expressing properties at the global level that are not necessarily given by the sum of the properties at the local level. To sum up, while [25] employs a bottom-up approach, we employ a top-down approach. We employed a similar approach in previous work [26], by studying the effects of different community discovery approaches in analysing the complex network of product associations.

Other relevant literature dealing with the problem of extracting knowledge from customer behaviour can be found in business intelligence. In this field, many data mining and OLAP techniques have been developed, enriching the analytic tools [4, 27], not only for

marketing purposes but also to detect frauds [28], or for public health surveillance [29]. Data mining and customer behaviour has gone also one step forward, by exploiting sentiment analysis as a prediction tool for a product success/failure [30].

Our approach is a combination of application and evolution of some tools present in literature. First, for some specific tasks our framework makes use of the Revealed Comparative Advantage (RCA) measure. The RCA measure has been defined in international economics [31], but the very same concept has been borrowed in many fields. For example, the RCA measure is equivalent to the lift. Lift (as conviction, collective strength and many more) is one criterion used in association rule mining to evaluate the interestingness of a rule [32].

Second, we make use of concepts related to ecology literature [13] and macro economics [8, 17]. While using similar techniques (as the eigenvector factorization of the customer-product matrix to calculate the sophistication levels of both customers and products), our work differs from the ones presented on two axis: quality and quantity of data. As for the quality of the data, we deal with micro purchases instead of macro world trade or ecosystem presence/absence of animal species. As for the quantity of data, we work with matrices with a number of cells $\sim 10^9$ while related works do not scale beyond $\sim 10^5$ and therefore cannot be used in our scenario.

Our analysis of customer mobility has been designed and performed also in a data-mining oriented scenario, in previous work [21]. For that paper, we also publicly released our anonymized data, for result verification purposes.[e]

5.2 Strength and limitations of this study

To the best of our knowledge, this is the first study applying the logic of complex system theory to the retail market. This is the main strength of the paper, because it empowers researchers and market analysts with a new way of thinking this field of study. Many classical results from complexity theory can now be applied to this scenario, and the universe of testable hypotheses has been enlarged.

We backed up this claim by showing three applications. We showed that it is possible to have a data-driven large scale observation of the hierarchy of human needs. Previously, this theory could only be tested in very bounded cases. Moreover, we uncovered some aspects of the logic of customer behaviour. We did so limiting our attention to their movements on the territory. As a result of our analytic vantage point, we could discover that their mobility is more predictable than previously thought. We are able to predict part of the variance in their movements just by knowing what types of products are sold in different supermarkets of an area.

There are many limitations in the study here presented. Even if we partially controlled from time and space, by creating alternative views of our dataset from different regions in space and time, we still have a biased view of customer behaviour. In fact, our entire study is confined inside the cultural environment of Italy. This makes our empirical hierarchy of needs biased towards what are the basic and sophisticated needs for the Italian people. Moreover, we used the internal marketing classification of the supermarket under study to redact our hierarchy. This is another source of bias that can be fixed by using data from other countries, as well as an international standard product classification such as SITC[f] or HS.[g]

As a second limitation, a deeper understanding of the mechanics of the purchase matrix could be a promising future work of this paper. One could define a null model using the

Maximum Entropy Principle [33] and test whether the results shown in the paper still hold.

Thirdly, in the definition of the Purchase Function, we did not consider the number of parameters as a penalty for the functional form. It is not surprise, then, that the function with more parameters fits the data better. As future work, we will include penalties for the number of parameters and test whether the current shape of the function still provides the best results.

Also the mobility study is influenced by the technology level of Italy. Countries with better, or worse, infrastructure might show different patterns.

5.3 Conclusion

In this paper we analysed large quantities of data extracted from the retail activity of the customer subset of an Italian supermarket chain. Our aim was to build a framework able to exploit a different vantage point over retail purchase data. We highlighted some properties of retail data, namely uneven distributions of connections in the customer-product bipartite structure and the triangular structure of its adjacency matrix. These properties make association rule mining results incomplete. By looking at the retail data as a complex system, as we did in this paper, we can develop an alternative and complementary methodology to analyse purchase data.

Our thesis is that customers usually buy the same set of basic products and the more sophisticated products are only bought by customers buying everything, generating a triangular adjacency matrix for the bipartite customer-product network. Our framework is able to analyse this structure as a whole, instead of looking at the local patterns like classical rule mining, uncovering the general pattern of shopping behaviour. Building on this theory, we define a the Purchase Function that can identify the set of customers buying a specific product by looking simply at how much the product is sold (and vice versa); and a way to rank the sophistication level of both products and customer needs. We showed some possible applications of these results: a data driven empirical observation of Maslow's theory of needs; an efficient way to identify a small set of potentially very interested customers for a given product p_i; and a way to predict customer mobility on the territory.

Our work opens the way to several future developments. The first one concerns the validation of our observation of the hierarchy of needs, as it is based on a narrow geographical set of people and on a non-standard product category classification. Also, with more data we can extend our pyramid of needs to fully cover the entire spectrum of human needs. Another interesting track of research may be to investigate what is the minimum time window needed to observe the prerequisites of the Purchase Function, maybe linked with the cyclic behaviour of customers [34] and/or with the stability of customer and product ranking order in the matrix [35]. Another application scenario may be to fully exploit the purchase matrix as a complex system: to analyse products not only based on their product sophistication index, but by looking at the product-product relationship level; or to try to find the way of controlling the complex system [36].

Appendix 1: Experimental setting

The analysis presented in this paper are performed with regular user-end computers. No mainframes or parallel computing techniques have been used. The fit of the Purchase

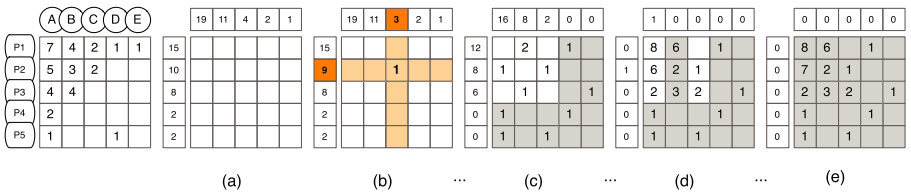


Figure 15 Null model generation. Schematic visualization of the process to generate a null model from the toy example in Figure 5. **(a)** We start from an empty matrix with a set of threshold for columns and rows; **(b)** at each step we select a pair of column and row index to increment a component; **(c)** after a sequence of assignments, the columns (rows) with no availability are blocked (grayed); **(d)** last step where only one pair of column and row is available; **(e)** the resulting null model.

Function f_*, the marketing analysis and the computing of Product and Customer Sophistication via eigenvector calculation have been performed each one in less than one hour on a Dual Core Intel i7 64 bits @ 2.8 GHz laptop, equipped with 8 GB of RAM and with a kernel Linux 3.0.0-12-generic (Ubuntu 11.10), using a combination of Octave, Numpy and Scipy Python libraries. The data preparation pipeline, and null model generation and evaluation, have been computed on a Quad Core Intel Pentium III Xeon @ 2 GHz, equipped with 8 GB of RAM and with Windows Server 2003, using Java 1.6. The most memory and time consuming operation was the null model generation: each null model required 6 GB of memory and 4 hours of computing. The conclusion is that our framework is able to scale and to analyse large data quantities.

Appendix 2: Null model

For the null model, we need to generate a random matrix where the observed sums of rows and columns are preserved. In literature there is an algorithm providing this feature [37], but it is not designed to work on very large matrices. Therefore, we extract a null model according to the algorithm explained below. A visual schematic representation of the different steps is presented in Figure 15.

We use two sets (*PLeft* and *CLeft*) to keep track of the rows and columns that are not yet full: customers that have not yet reached their amount of products bought and products that have not yet reached their diffusion among the customers. Vector R (C) keeps track in each cell of the respective residual in the row (column). The integer *NItemsLeft* contains the total number of purchases.

We start from an empty matrix, with the same dimensions as our real data matrix and with all cells initialized at 0. We iterate until we have a product left to place, i.e. as long as $NItemsLeft > 0$. At each iteration we randomly extract a position from the set of cells that are still increasable (stored in *CLeft* and *PLeft*). At this point, we just increase by 1 the value of the cell extracted, we decrease the residual of the row and the column selected (in R and P) and of the total number of purchases (*NItemsLeft*). Finally, we check if the column (row) selected has been filled and, in this case, we remove the column (row) index from the set *Pleft* (*CLeft*). After building this null adjacency matrix, we calculate the RCA for each cell, applying the pre-process step of our methodology. We obtain a null M_{cp} matrix and we can then confront it with the original one to understand if they are similar or not (and therefore if the shape of the original matrix is meaningful or not).

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
DP1 and MC performed research, prepared figures, carried out empirical analysis and wrote the manuscript. All authors designed research and reviewed the manuscript.

Author details
[1]IMT, P.za San Ponziano 6, Lucca, Italy. [2]ISTI, CNR, Via G. Moruzzi, 1, Pisa, Italy. [3]CID, Harvard University, 79 JFK St, Cambridge, USA. [4]Department of Informatics, University of Pisa, Largo B. Pontecorvo 3, Pisa, Italy.

Acknowledgements
We gratefully thank Muhammed Yildirim, César Hidalgo, Jenny Zambon and Sebastian Bustos for their support and useful discussions. We thank the supermarket company Coop and Walter Fabbri for sharing the data with us and allowing us to analyse and to publish the results. This work has been partially supported by the European Commission under the FET-Open Project n. FP7-ICT-270833, DATA SIM.

Endnotes
[a] The news of the study, in Italian, can be found at http://www.viasarfatti25.unibocconi.it/notizia.php?idArt=6527. The PI of the study can be reached at isabella.soscia@skema.edu.
[b] Also note that, for some reason, 'Chemicals' such as band aids or rat poison are classified under 'Food', although we advise not to eat these things.
[c] This happens because the matrix is subject to the Perron-Frobenius theorem. To be applicable, the theorem has two requirements: the matrix must be aperiodic and irreducible. Being symmetric, $\widetilde{M}$ satisfies the aperiodicity requirement. We also make use only of the largest giant component of $M_{c,p}$, which implies that $\widetilde{M}$ has only one component too, and thus satisfies the irreducibility requirement.
[d] This and all other regressions have been calculated with the *leastsq* function of the *SciPy* module for Python.
[e] http://www.michelecoscia.com/?page_id=379.
[f] http://unstats.un.org/unsd/cr/registry/regcst.asp?Cl=14.
[g] http://hts.usitc.gov/.

Received: 23 June 2014 Accepted: 2 December 2014 Published online: 11 December 2014

References
1. Agrawal R, Imielinski T, Swami AN (1993) Mining association rules between sets of items in large databases. In: SIGMOD international conference, Washington, D.C., pp 207-216
2. Sun Y, Aggarwal CC, Han J (2012) Relation strength-aware clustering of heterogeneous information networks with incomplete attributes. Proc VLDB Endow 5(5):394-405
3. Chaudhuri S, Narasayya VR (2011) New frontiers in business intelligence. Proc VLDB Endow 4(12):1502-1503
4. Kocakoç ID, Erdem S (2010) Business intelligence applications in retail business: OLAP, data mining & reporting services. J Inf Knowl Manag 9(2):171-181
5. Brauckhoff D, Dimitropoulos X, Wagner A, Salamatian K (2012) Anomaly extraction in backbone networks using association rules. IEEE/ACM Trans Netw 20(6):1788-1799
6. Marinica C, Guillet F (2010) Knowledge-based interactive postmining of association rules using ontologies. IEEE Trans Knowl Data Eng 22(6):784-797
7. Montella A (2011) Identifying crash contributory factors at urban roundabouts and using association rules to explore their relationships to different crash types. Accid Anal Prev 43(4):1451-1463
8. Hidalgo CA, Klinger B, Barabási AL, Hausmann R (2007) The product space conditions the development of nations. Science 317(5837):482-487. doi:10.1126/science.1144581
9. Hausmann R, Hidalgo C, Bustos S, Coscia M, Chung S, Jimenez J, Simoes A, Yildirim M (2011) The atlas of economic complexity. Boston, USA
10. Caldarelli G, Cristelli M, Gabrielli A, Pietronero L, Scala A, Tacchella A (2011) Ranking and clustering countries and their products; a network analysis. arXiv:1108.2590
11. Davis WL, IV, Schwarz P, Terzi E (2009) Finding representative association rules from large rule collections. In: SDM, pp 521-532
12. Maslow AH (1943) A theory of human motivation. Psychol Rev 50(4):370-396
13. Bascompte J, Jordano P, Melián CJ, Olesen JM (2003) The nested assembly of plant-animal mutualistic networks. Proc Natl Acad Sci USA 100(16):9383-9387. doi:10.1073/pnas.1633576100
14. Almeida-Neto M, Guimarães P, Guimarães PR, Jr., Loyola RD, Ulrich W (2008) A consistent metric for nestedness analysis in ecological systems: reconciling concept and measurement. Oikos 117:1227-1239. doi:10.1111/j.0030-1299.2008.16644.x
15. Pennacchioli D, Coscia M, Giannotti F, Pedreschi D (2013) Calculating product and customer sophistication on a large transactional dataset. Technical report cnr.isti/2013-TR-004
16. Marquardt DW (1963) An algorithm for least-squares estimation of nonlinear parameters. J Soc Ind Appl Math 11(2):431-441
17. Hidalgo CA, Hausmann R (2009) The building blocks of economic complexity. Proc Natl Acad Sci USA 106(26):10570-10575. doi:10.1073/pnas.0900943106
18. Cristelli M, Gabrielli A, Tacchella A, Caldarelli G, Pietronero L (2013) Measuring the intangibles: a metrics for the economic complexity of countries and products. PLoS ONE 8(8):e70726
19. Guidotti R (2013) Mobility ranking - human mobility analysis using ranking measures. University of Pisa

20. Wang H, Song M (2011) Ckmeans.1d.dp: optimal *k*-means clustering in one dimension by dynamic programming. R J 3(2):29-33
21. Pennacchioli D, Coscia M, Rinzivillo S, Pedreschi D, Giannotti F (2013) Explaining the product range effect in purchase data. In: 2013 IEEE international conference on big data, pp 648-656
22. Krumme C, Llorente A, Cebrián M, Pentland A, Egido EM (2013) The predictability of consumer visitation patterns. CoRR. arXiv:abs/1305.1120
23. Cohen E, Datar M, Fujiwara S, Gionis A, Indyk P, Motwani R, Ullman JD, Yang C (2000) Finding interesting associations without support pruning. In: ICDE, pp 489-500
24. Nguyen K-N, Cerf L, Plantevit M, Boulicaut J-F (2011) Multidimensional association rules in Boolean tensors. In: SDM, pp 570-581
25. Chawla S (2010) Feature selection, association rules network and theory building. J Mach Learn Res 10:14-21
26. Pennacchioli D, Coscia M, Pedreschi D (2014) Overlap versus partition: marketing classification and customer profiling in complex networks of products. In: Workshop of the international conference of data engineering (ICDE)
27. Li H (2005) Applications of data warehousing and data mining in the retail industry. In: Proceedings of ICSSSM'05: 2005 international conference on services systems and services management, vol 2
28. Gabbur P, Pankanti S, Fan Q, Trinh H (2011) A pattern discovery approach to retail fraud detection. In: KDD, pp 307-315
29. Wagner MM, Robinson JM, Tsui F-C, Espino JU, Hogan WR (2003) Design of a national retail data monitor for public health surveillance. J Am Med Inform Assoc 10(5):409-418
30. Castellanos M, Dayal U, Hsu M, Ghosh R, Dekhil M, Lu Y, Zhang L, Schreiman M (2011) LCI: a social channel analysis platform for live customer intelligence. In: SIGMOD conference, pp 1049-1058
31. Balassa B (1965) Trade liberalization and 'revealed' comparative advantage. Manch Sch 33:99-123
32. Geng L, Hamilton HJ (2006) Interestingness measures for data mining: a survey. ACM Comput Surv 38(3):9. doi:10.1145/1132960.1132963
33. Bousquet N (2010) Eliciting vague but proper maximal entropy priors in Bayesian experiments. Stat Pap 51(3):613-628
34. Shen Z-JM, Su X (2007) Customer behavior modeling in revenue management and auctions: a review and new research opportunities. Prod Oper Manag 16(6):713-728. doi:10.1111/j.1937-5956.2007.tb00291.x
35. Schich M, Lehmann S, Park J (2008) Dissecting the canon: visual subject co-popularity networks in art research. In: ECCS2008
36. Liu Y-Y, Slotine J-J, Barabási A-L (2011) Controllability of complex networks. Nature 473(7346):167-173. doi:10.1038/nature10011
37. Patefield WM (1981) An efficient method of generating random RxC tables with given row and column totals (algorithm AS 159). J R Stat Soc, Ser C, Appl Stat 30:91-97. doi:10.2307/2346669

doi:10.1140/epjds/s13688-014-0033-x
Cite this article as: Pennacchioli et al.: **The retail market as a complex system.** *EPJ Data Science* 2014 **3**:33.