Dipartimento di
Economia, Statistica e Diritto
Università di Pavia

Serie Statistica

n. 9/2011

Luisa Cutillo, Annamaria Carissimo, Silvia Figini

***Network Aggregation*: a method for rank aggregation via network evidence**

QUADERNI DEL DIPARTIMENTO DI ECONOMIA, STATISTICA E DIRITTO
UNIVERSITÀ DI PAVIA

REDAZIONE

Claudia Banchieri
Dipartimento di Economia, Statistica e Diritto
Università degli Studi di Pavia
Corso Strada Nuova 65
27100 PAVIA
tel. 0039-0382-984406
fax 0039-0382-984402
E-MAIL banchier@unipv.it
I Quaderni sono disponibili sul sito: http://www-5.unipv.it/webdesed/quaderni.php

Network Aggregation: a method for rank aggregation via network evidence

Luisa Cutillo[*]
University of Naples *Parthenope*,
Annamaria Carissimo
Telethon Institute of Genetics and Medicine
and
Silvia Figini
University of Pavia

July 18, 2011

1 Abstract

We address the problem of rank aggregation, the task of combining different rank orderings on the same set of units (preference lists) so as to achieve a single final ordering. We propose *Network Aggregation*, a novel approach for rank aggregation inspired to graph theory. The rationale of this choice relies on the observation that, after a custom preprocessing, we are able to view our set of lists as a network. Only the lists populating the same community in the network would then be aggregated. In order to highlight the strength of our proposal, we show an application both on simulated and on real data set. Experimental results on simulated data show that *Network Aggregation* can significantly outperform existing related methods. Moreover, the empirical evidences achieved on real financial data reveal that *Network Aggregation* is also able to select the most relevant variables in data mining predictive models, providing a clear superiority in terms of predictive

[*] luisa.cutillo@uniparthenope.it

power of the models build.

Keywords: Rank Aggregation, Graph Theory, Network Community, Risk Integration, Supervised Predictive Models

2 Introduction

In the latest years, rank aggregation methods have emerged as an important approach able to combine the ranking information from different statistical units. In diverse interest areas, the rank aggregation process is usually devoted to the merging of different preference lists on the same set of units. Relevant applications are collected in marketing and advertisement research, applied psychology, internet search engines and more recently in omics scale biological studies. In the literature, this problem was first addressed by Arrow (Arrow K. J. , 1950), Kemeny (Kemeney J. G. , 1959) and later, in terms of application to the World Wide Web data, by (Dwork C. et al., 2001).

On the basis of our experience rank aggregation techniques revel to be very informative also in the filed of economic applications, especially in risk analysis and risk integration. In particular, given a set of statistical units (i.e. a set of enterprises) potentially at risk of failure, it would be highly interesting to order them using a collection of variables available. In this perspective, we think that rank aggregation methods lend naturally to the field of economic applications. In the present paper we propose a novel methodology for rank aggregation and we show an application on a real financial data set.

Despite its clear and intuitive target, effective rank aggregation becomes difficult in real-world situations in which the set of collected rankings can be noisy, incomplete, or even disjoint. The biggest challenges of the aggregating process remain today the choice of an appropriate measure of dissimilarity between lists, and a reasonable top k length for a particular list (DeConde et al. (2006), Pihur and Datta (2008)). In order to overcome the weaknesses of classical rank aggregation techniques, we propose a methodological approach that directly take into account that a unique true underling ranking might not exist. Moreover we point out that only the lists that share the highest amount of information should be aggregated and moreover only consensus sets of lists should be considered for the aggregation process.

Our heuristic rank aggregation method is inspired to the graph theory. The rationale of this choice relies on the observation that, after a custom preprocessing, we are able to view our set of lists as a network. The preprocessing step that we will

describe later on, basically consists in choosing an appropriate measure of dissimilarity between lists and in performing an hypergeometric hypothesis test on each computed distance. The constructed network would then be partitioned via a standard communities extraction method (Pons and Latapy (2006)) . Only the set of lists populating the same community in the network would then be aggregated. For sake of brevity, we introduce the acronym *NetAggreg* to refer to our *Network Aggregation* method .
The paper is organized as follows: Section 2 presents the general framework and underlines the main elements of novelty of our proposal, Section 3 reports the computational algorithm implemented, Section 4 describes the application on a real financial data set and Section 5 ends the paper and underlines further ideas of research.

3 General framework

Before to describe our proposal, we introduce the general framework of the rank aggregation (RA) problem for a discrete set of statistical units.
RA methods can be broadly classified as distributional based, stochastic optimization and heuristic algorithms (Lin S. , 2010). The first category is populated by Thurstone method and it's extensions. These methods reveal to be appropriate for aggregating many short ranked lists. Optimization algorithms are based on an optimization criteria and are usually dependent on the distance measure. in fact, given a distance measure, they aim to find the aggregate list as the candidate list that minimize its distance from all the input lists. An instance of this category is the Kemeny optimal aggregation which optimizes the average Kendall's distances (Dwork et al., 2001). Unfortunately it is well known that computing the Kemeny optimal aggregate is NP-hard even when the number of ranked lists to be aggregated is small; this is due to the combinatorial nature of the problem. These difficulties can be circumvented by stochastic search algorithms as described in (Lin and Ding 2009).
A novel alternative to direct optimization is given by the heuristic algorithms that are capable to provide approximate solutions to the RA problem without optimizing any criterion. Despite the effective applicative results shown in (Dwork et al., 2001; DeConde et al., 2006), the heuristic nature of these category of algorithms prevent from the exploration of well defined properties.
From a different point of view we can classify the RA method according to the average length of the set of lists under study. The problem of aggregating many

short lists is addressed by the distributional based and stochastic optimization algorithm, while the problem of aggregating a few long lists is tackled mainly by heuristic algorithms. In this paper we focus on the problem of aggregating many long lists. None of the RA method mentioned so far can handle this problem. The main limitation of the heuristic algorithms is the unfairness of the result for non homogeneous set of long lists. In the case of many long lists it is reasonable they are non homogeneous and thus the the aggregate list reveals to be random. On the other hand, the limitation of the distributional based and stochastic optimization algorithm is that, despite effective for very short lists, they reveal to be unfeasible for long lists.

In the present paper we propose an innovative heuristic RA strategy that is particularly suited for the problem of aggregating many long lists.

3.1 Preliminaries

In this section we introduce some necessary concepts and notations. Let U be a set of n objects each with a unique identifier $i \in \{1,\dots,n\}$ and consider a subset $S \subseteq U$. A ranking function on S is a permutation r on the set S. For each object $i \in U$, $r(i) \in \{1,\dots,|S|\}$ shows the ranking of item i. We will say that L is a ranked list of elements of U with ranking function r if :

$$r : a \in L \longrightarrow r(a) \in \{1 \dots |L|\} \tag{1}$$

We will use the notation r^L to refer to r, in order to explicit the linkage between the ordered list L and it's ranking function. Note that the best ranking is 1, rankings are always positive, and higher rank correspond to lower preference in the list.

For sake of simplicity, we will make an abuse of notation and use S=L.

A *full list* is a list that contains a ranking for every item $i \in U$, that is $S = U$, in this case its ranking function is a *complete ranking* on U.

A *partial list* is a list that contains rankings for only a proper subset of items $S \subset U$. A partial list will be also referred to as a *Top-k* when $|S| = |L| = k$.

Note that in this case we assume that all other items $i \notin S$ are supposed to be ranked below every item in S according to a customized ranking value. Given a set of complete or incomplete lists, we need to provide an approximate solution to their RA problem.

In order to clarify the overall procedure described in section 4, we will briefly recall Borda-inspired methods and optimization method.

Borda-inspired algorithms is a family of RA methods intuitive and easy to understand that basically reproduce a voting strategy. Jean-Charles de Borda in 1781, originally proposed to aggregate ranks by sorting the ranks arithmetic average for full ranked lists. Many other variations of Borda method have been proposed and used, and are applicable to top-k lists.

Suppose we have N ordered complete lists $L_i, i \in \{1, \dots, N\}$, the Borda score associated to a generic element $u \in U$ for the list L_i is $B_{L_i}(u) = r^{L_i}(u)$ apart from a scaling factor. Borda's score may in fact take into account other additional information then the rankings when available. Once defined a Borda's aggregating function $B(u) = f(B_{L_1}(u), \dots, B_{L_N}(u))$, the aggregated ranked list is obtained by sorting all the aggregated Borda's scores. An example of mostly suggested aggregating function is the *p-norm*:

$$f(B_{L_1}(u), \dots, B_{L_N}(u)) = \sum_{i=1}^{N} B_{L_i}(u)^p / N \tag{2}$$

We observe that the original method proposed by Borda in 1781 is a particular case of the p-norm, with $p = 1$. The extension of this method to the Top-k case is straightforward. In fact, If $k_i = |L_i|$ and $u \in S = \bigcup_{i=1}^{N} L_i$, but $u \notin L_j$, it will be $B_{L_j}(u) = k_j + 1$. Of course this score can be modified accordingly to other information when available. The method then proceed just as for as the full ranked lists case.

On the other hand, optimization methods is a family of algorithms that address the RA problem in terms of an optimization rule. The most common optimization strategies are based on a measure of disagreement between the input top-k lists and the unknown aggregate rankings. One formulation that follows the generalized Kemeny criterion is the weighted sum of distances between the aggregate rankings and the input lists. Thus, whether a particular aggregate list is better than another depends on the distance measure chosen.

Given two lists L_i and L_j, the Kendall tau distance between them, denoted by $K(L_i, L_j)$, is the number of couples of elements $(u, v) \in SxS$, where $S = L_i \bigcup L_j$, such that either $r^{L_i}(u) < r^{L_i}(v)$ but $r^{L_j}(u) > r^{L_j}(v)$, or $r^{L_i}(u) > r^{L_i}(v)$ but $r^{L_j}(u) < r^{L_j}(v)$.

It is easy to see that $K(L_i, L_j)$ measures the number of pairwise disagreements between the two lists. The footrule distance between L_i and L_j, denoted by $F(L_i, L_j)$, is defined to be $F(L_i, L_j) = \sum_{u \in S}(|r^{L_i}(u) - r^{L_j}(u)|$. This distance express a sort of total absolute deviation of the two lists on single elements but does not take into

account the relative orderings of couple of elements.
Consider now N ordered complete lists $L_i, i \in \{1, \dots, N\}$. A Kendall optimal aggregation of the given set of lists is any aggregate lists L that minimizes $\sum_{i=1}^{N} K(L, L_i)$; on the other hand, a Footrule optimal aggregation is any list L that minimizes $\sum_{i=1}^{N} F(L, L_i)$. As previously noticed, computing a Kendall optimal aggregation is NPhard, while computing a Footrule optimal aggregation can be done in polynomial time via minimum cost perfect matching (Dwork C. et al. (2001)).
Nevertheless, in the majority of the cases, it is of higher interest to provide an aggregate list that accounts for the most frequent pairwise agreements in the set of input lists.

4 Our proposal

On the basis of the remarks reported in Section 3, in this paper we focus on the problem of aggregating many long lists. In the followings we describe our contribution that results in a novel algorithm able to tackle a non homogeneous set of lists composed by a large set of statistical units.
NetAggreg overall procedure can be broadly summarized in four steps.
The first step considers the distance matrix allocation. In order to aggregate a given set of lists, it is required to define a degree of similarity. To reach this objective, we have to introduce a similarity-dissimilarity measure between couple of lists. If we interpret each list as a point in an l-dimensional space, this measure reveals to be a distance. There are several standard methods to define a distance measure between two lists. We choose the Kendall's tau metric defined as follows. Let L_i indicate a generic ordered list and let:

$$r^{L_i} : a \in L_i \longrightarrow r^{L_i}(a) \in \{1 \dots |L_i|\} \tag{3}$$

be the ranking function of the list L_i. Let $L_i(h)$ be the top h sublist of L_i. Suppose we have N ordered lists $L_i\{i = 1, \dots, N\}$ whose lengths, $k_i = |L_i|\{i = 1, \dots, L\}$, are not necessary the same. Our method iteratively works on pairs of full or partial ranked lists. We create a distance matrix according to the a modified version of the Kendall's tau distance (Pihur and Datta , 2008):

$$K_{i.j} = K(L_i, L_j) = \sum_{t,u \in L_i \cup L_j} K_{tu}^{p} \tag{4}$$

where $K_{tu}^{p} : L_i x L_j - > \{0, 1, p\}$ is a piece-wice function of the relative orderings (3) defined as follows:

$$K_{tu}^{p} = \begin{cases} 0 & \text{if } r^{L_i}(t) < r^{L_i}(u), r^{L_j}(t) < r^{L_j}(u) \text{ or } r^{L_i}(t) > r^{L_i}(u), r^{L_j}(t) > r^{L_j}(u) \\ 1 & \text{if } r^{L_i}(t) > r^{L_i}(u), r^{L_j}(t) < r^{L_j}(u) \text{ or } r^{L_i}(t) < r^{L_i}(u), r^{L_j}(t) > r^{L_j}(u) \\ \text{p} & \text{if } r^{L_i}(t) = r^{L_i}(u) = k_i + 1 \text{ or } r^{L_j}(t) = r^{L_j}(u) = k_j + 1. \end{cases} \tag{5}$$

Our choice of $p = \frac{|L_i \cup L_j| - |L_i \cap L_j|}{|L_i| \cup |L_j|}$ accounts for the relative mismatches of the two lists. The choice of this measure relies on the fact that the information we want to extract from the overall set of lists is which are the set of lists that share the same relative ordering of couple of individuals.

The second step consists in translating the distance matrix into the adjacency matrix of an undirected graph. Let $K_{i,j}$ be the generic element of the distance matrix obtained so far. $K_{i,j}$ shows us how dissimilar list i and list j are, but we want to be more strict on the concept of dissimilarity using a statistical test of significance. In this perspective we reduce the distance matrix K in a 0-1 adjacency matrix D of an undirected graph where each vertex is a list. This is achieved via an hypothesis test on the match value of each couple of lists as explained in the following.

Given a couple of lists of length l we test the null hypothesis H_0 that the two lists are dissimilar versus the alternative H_1 that the two lists are similar. Under H_0 the measured number of matches has an hypergeometric distribution:

$$X \sim Hyper(n,M,N) \tag{6}$$

of parameters $n = \binom{l}{2}$, $N = 2 * \binom{l}{2}$ and $M = \binom{l}{2}$.

The rationale of this is that if the two lists are dissimilar the possible matches and mismatches are undistinguishable and equiprobable. In particular let $f = n/N$ be the sampling fraction and let $p = M/N$ denote the proportion of matches in the population. Normal approximations to Hypergeometric distribution are classical in the standard cases where f and p are bounded away from 0 and 1 (Feller, 1971). Thus under H_0 we approximate the hypergeometric distribution with the Normal distribution with mean $\mu = np$ and variance $\sigma^2 = Nf(1-f)p(1-p)$.

For each $K_{i,j}$ the corresponding $D_{i,j}$ would be set either to one, if the null hypothesis is rejected, or to zero otherwise.

The third step is devoted to the extraction of similar lists communities from the network constructed in the second step. The adjacency matrix built so far would in fact be used to individuate the set of similar lists and to eventually isolate outliers. This is carried out through a community extraction algorithm as we assume our list network consists of modules which are densely connected themselves but sparsely connected to other modules. In this light we performed the community structure detection via a standard algorithm based on random walks (Pons and Latapy , 2006).

The goal of the fourth and last step is to provide a consensus aggregate list for each of the extracted communities according to the third step. The aggregation is performed via standard literature aggregation methods for partial lists. We choose the Borda's method (voting strategy) that, as said in the preprocessing section, has a very low computational cost and reveals to be efficient on homogeneous set of lists.

Our strategy enables to isolate outliers in our set of lists and tells which are the community of lists sharing the same information. For each community this information is provided by the list resulted from the aggregation step summarizing and representing the over all community. *NetAggreg* also provides a set of indicators that would suggest which communities are more representative of the underling observed units. Suppose we detected nC communities C_h , and assume that each community has size $s_h = |C_h|$ and aggregate list AL_h, with $h \in \{1, \ldots, nC\}$. Our indicators are defined as follows:

- $\delta_h = \frac{\frac{\sum_{i,j \in C_h} K_{i,j}}{\binom{s_h}{2}}}{\binom{l}{2}}$ and $\sigma_h = \frac{1}{\binom{l}{2}} \sqrt{\frac{\sum_{i,j \in C_h} (K_{i,j} - \binom{l}{2} \delta_h)^2}{\binom{s_h}{2}}}$
the δ_h gives the average percentage of mismatches within the same community and the σ_h is its standard deviation.

- $\delta A_h = \frac{\frac{\sum_{j \in C_h} K(AL_h, L_j)}{s_h}}{\binom{l}{2}}$ and $\sigma A_h = \frac{1}{\binom{l}{2}} \sqrt{\frac{\sum_{j \in C_h} (K(AL_h, L_j) - \binom{l}{2} \delta_h)^2}{s_h}}$
similarly the δA_h gives the average percentage of mismatches between the aggregate list AL_h and the lists in community C_h, while σA_h provides its standard deviation.

- $\delta_{h,k} = \frac{\frac{\sum_{i \in C_h} \sum_{j \in C_k} K_{i,j}}{\binom{s_h}{2}\binom{s_k}{2}}}{\binom{l}{2}}$ and $\sigma_{h,k} = \frac{1}{\binom{l}{2}} \sqrt{\frac{\sum_{i \in C_h} \sum_{j \in C_k} (K_{i,j} - \binom{l}{2} \delta_{h,k})^2}{\binom{s_h}{2}\binom{s_k}{2}}}$
On the other hand $\delta_{h,k}$ provides the average distance between each couple of identified communities$(h,k) \in \{1, \ldots, nC\} X \{1, \ldots, nC\}$ and $\sigma_{h,k}$ expresses its standard deviation.

Notice that, the most representative communities will be the one with the smallest δ_h and the smallest σ_h. Moreover, in the best scenario, the most representative communities (say h and k), would also reveal to be well separated in the sense that $\delta_{h,k} > \max\{\delta_h, \delta_k\}$ and $\sigma_{h,k}$ is small. Finally the goodness of the aggregate list will be dependent on the value of the parameters δA_h and σA_h.

5 Simulation Analysis

In this section we show the performances of *NetAggreg* on simulated data sets. We considered a scenario of s communities of lists, with $s \in \{2,6,10\}$. In order to control the ability of the method to recover the truth, we generated s underling true rankings (*generating lists*), that is a generating list for each community. We allowed the dissimilarity between them to be $\beta\%$ in terms of mismatches, with $\beta \geq 20$. Each community was then populated by lists with $\alpha\%$ of disagreement from the relative generating one, with $\alpha \in \{0.05,\dots,10\}$. The number of lists per community was set to $ns = nL/s$, where $nL \in \{60,120,240\}$ is the total number of lists in each simulation run. The desired distances were reached composing two possible source of mismatches, *inversion* and *block exchange*, as defined in the following.

Definition 1 *Given a ranking function $r(.)$ on a list L, we define inversion r^p the ranking of L obtained by the permutation that expresses the reverse ordering of L with respect to $r(.)$:*

$$r^p : a \in L \longrightarrow |L| - r(a) + 1 \in \{1 \dots |L|\}. \tag{7}$$

Observe that the ranked list resulting from the application of the inverse ranking $r^p(.)$ on the lists L reaches the maximum number of mismatches with L, that is $\binom{|L|}{2}$.

Definition 2 *Given a ranking function $r(.)$ on a list L, suppose it is possible to divide integrally the ranked list L in ncut consecutive sublists (or blocks) $L_{cut}(i)$, with $i \in \{1,\dots,ncut\}$. Assume that each block consists of m consecutive elements. We define block exchange of jump $|j-i|$ the exchange of the rankings of all the elements of block $L_{cut}(i)$ with the rankings of the corresponding elements of block $L_{cut}(j)$, for $i, j \in \{1,\dots,ncut\}$. That is we define the new ranking $r_{i,j}$ as follows:*

$$\begin{aligned} r_{i,j}(L_{cut}(i)[h]) &= r((L_{cut}(j)[h]) \; \forall h \in \{1,\dots,m\} \\ r_{i,j}(L_{cut}(j)[h]) &= r((L_{cut}(i)[h]) \; \forall h \in \{1,\dots,m\} \\ r_{i,j}((L_{cut}(k)[h]) &= r((L_{cut}(k)[h]) \; \forall k \in \{1,\dots,ncut\}/\{i,j\} \end{aligned}$$

When $|L|$ is not integrally divisible by $ncut$, this definition can be trivially extended if the residual elements are included in the blocks external to $L_{cut}(i)$ and $L_{cut}(j)$. Note that the application of a *block exchange* of jump $|j-i|$ on a list L, produces a number of mismatches with respect to the original list equal to $(2*|j-i|+1)*|L_{cut}(i)|^2$.

The results from our simulations are summarised in terms of *sensitivity* and *specificity*. As main result we get that *NetAggreg* is robust with respect to the variability within the same group, the number of groups and the average cardinality of each group. On the other hand it is strongly influenced by the percentage of mismatches between groups. In fact, for any of the tested values of the parameters α, s and nS and only when $\beta \geq 50$, our method perfectly meet the true original communities with $specificity = sensitivity = 1$. Moreover the aggregate list of each extracted community and the corresponding generating list show a matche percentage greater than 80%. The other way round, when $\beta < 50$, our algorithm fails to detect the underlying simulated community structure as all the lists are assigned to the same community. In this case we still get $sensitivity = 1$ but the *specificity* turns to be zero. This is due to the true nature of the simulated data set that is composed of similar lists in terms of mismatches. In this case our method is not well suited as it reduces to a classical rank aggregation procedure based on the Borda method and thus we suggest to use a more specific custom strategy.

6 Application to real data

The main objective of this section is to apply our proposal on a real real data set composed of 1000 SMEs (Small and Medium Enterprises) and a set of financial ratios. We have decided to use financial data, because to our knowledge, there are not contributions of rank aggregation techniques in the field of credit risk analysis. Furthermore, since the data at hand are analysed also in Figini and Giudici (2011), we have a clear comparison of the superiority of our approach with respect to previous results.

In order to introduce the application on real financial data, we recall that credit is the loan that can only be granted by authorized fiancancial institutions or banks to the customer who apply for credit.

After a credit application is taken by a creditor, an assessment process is performed in order to decide whether to approve or reject grating credit to the applicant depending on the registered customer information expressed by quantitative and qualitative statistical variables. This process is known as credit scoring in finance literature, which is a classification method aiming to distinguish the desired customers who will fully repay from defaulters.

There have been several supervised methods applied to credit scoring of customers in literature such as discriminate analysis, linear regression, logistic regression, non parametric smoothing methods (i.e. Generalized Additive Models), genetic algorithm, neural networks, graphical models and others (see e.g. for a review Hand and Zhou (2009), Hand et al. (2010)).

Among this models we underline that in supervised classification, the aim is to construct a rule for assigning a score which represents a risk for each statistical units on the basis of a set of covariates available.

In order to predict the probability of defalut, $PD_i = P(Y_i = 1)$ in a traditional way a supervised model for credit risk estimation considers for observation $i = 1, \ldots, N$, Y_i as the objective binary variable which takes value 0 if the customer is good and 1 otherwise and a set of p covariates $L_{i1}, \ldots, L_{ip}$.

More precisely, a credit scoring model summarises all the information available measured on the variables in a single list which reports for each statistical units the corresponding probability of default. This means that starting from a multivariate problem, we derive only one variable which can be used to provide an ordering of risk among the statistical units at hand.

On the basis of our methodological proposal, described in Section 2 and 3, we think that *NetAggreg* can improve the results achieved in supervised models because it take into account the information on each list, providing a better ordering and comparison in the data collected.

We show that using *NetAggreg*, we are able to select groups of variables (lists) which provide similar order in terms of risk for the statistical units. This means that our approach leads also to select groups of features highly related to default.

Furthermore, we highligth that *NetAggreg* is more rubust with respect to data mining with missing data, corrupted data, inconsistent data and outliers.

In our analysis, we use a real data set provided by a credit rating agency. For the i considered statistical units, $i = 1, \ldots, N$, our information consists of a binary response variable Y_i and a set of explanatory variables $L_1, \ldots, L_p$. In particular, the data set is composed of companies with negative solvency (default) if $Y_i = 1$ and companies with positive solvency (not default) if $Y_i = 0$. The number of observations is equal to 1000. We have considered this set of financial ratios:

- *Supplier target*: a temporal measure of financial sustainability expressed in days that considers all short and medium term debts as well as other payables;

- *Outside capital structure*: the capability of the firm to receive other forms of financing beyond banks' loans;
- *Cash Ratio:* the cash a company can generate in relation to its size.
- *Capital tied up*: the turnover of short term debts with respect to sales;
- *Equity ratio*: a measure of a company's financial leverage calculated by dividing a particular measure of equity by firm's total assets;
- *Cash flow to effective debt*: the cash a company can generate in relation to its size and debts;
- *Cost income ratio*: an efficiency measure similar to the operating margin that is useful to measure how costs are changing compared to income.
- *Trade payable ratio*: how often the firm payables turn over during the year; a high ratio means a relatively short time between purchase of goods and services and payment for them, otherwise a low ratio may be a sign that the company has chronic cash shortages;
- *Liabilities ratio*: a measure of a company's financial leverage calculated by dividing a gross measure of long-term debt by firm's assets; it indicates what proportion of debt the company is using to finance its assets.
- *Result ratio*: how profitable a company is relative to its total assets; it gives an idea as to how efficient management is at using its assets to generate earnings.
- *Liquidity ratio*: this ratio measures the extent to which a firm can quickly liquidate assets and cover short-term liabilities, and therefore is of interest to short-term creditors.

An intensive descriptive analysis of this data can be found in Figini and Giudici (2011). The a priori probability of default is equal to 12.5%. On the basis of the data at hand, in order to predict the probability of default for each SME we have compared a classical logistic regression model with a classification tree. The logistic regression selects as significant variables: equity ratio and result ratio, while classification tree reports result ratio, equity ratio, capital tied up, supplier target days and result ratio as relevant variables.

In order to select the best model we have done a cross validation exercise using 70% of observations as trainig data and 30% of observations as validation data. We have emploied different measures of performances (on the validation set) based on the confusion matrix (see e.g. Hand et al. (2010)) and assessment indicators derivable througth the lift and the response chart (see e.g. Giudici and Figini (2009)).

On the basis of the lift, we puts the observations in the validation set into increasing or decreasing order on the basis of thier score, which is the probability of the response event (default), as estimated on the basis of the training set. It subdivides these scores into deciles then calculates the observed probability of default for each of the decile classes in the validation set.

A model is good if the observed success probabilities follow the same order as the estimated probabilities.

The cumulative captured response (CCR) gives for each decile the percentage of predicted events. If the model were perfect this percentage would be 100% for the first deciles and equal to zero for the other deciles.

We have also considered for each model, the area under the receiver operating characteristic (AUC) chart which is a graphical display that gives the measure of the predictive accuracy of a model. The out of sample performance of logistic regression and tree models computed using all the variables available are shown in Table 1. Considering the lift and the cumulative captured response, we choose as best model the classification tree which captures using only the first three deciles the 68.70% of the event of interest. On the basis of the validation set, we remark that the AUC are equal to 0.78 for the logistic regression and 0.85 for the tree model; furthermore, the percentage of correct classifications is equal to 80.5% for the logistic regression and 86% for the classification tree.

However, looking at the nature and the meaning of the financial ratios selected by the logistic regression, we think that result ratio and equity ratio can provide only an idea on how is efficient the management to use its assets to generate earnings and equity ratio. On the other hand, classification tree selects as relevant to predict default a set of features very heterogeneous and different with respect to business practice and expert opinions.

This lead us to investigate a different approach to select the relevant features to do predictive models starting from a set of covariates which can generate equal ranking in terms of default forecasting. We hope also that the variables selected have a clear interpretation in terms of business knowledge and expert opinion and provide also an improvement in terms of predictive performances.

To this purpose we applied *NetAggreg* to our set of financial ratios. As shown in

Model	**decile**	**LIFT**	**CCR**
Tree	1	2.77	27.71
Tree	2	2.54	53.07
Tree	3	1.56	68.70
Tree	4	0.57	74.36
Tree	5	0.46	78.93
Tree	6	0.46	83.49
Tree	7	0.46	88.05
Tree	8	0.46	92.62
Tree	9	0.46	97.18
Tree	10	0.28	100.00
Log Reg	1	0.09	0.86
Log Reg	2	0.69	7.76
Log Reg	3	0.34	11.21
Log Reg	4	0.60	17.24
Log Reg	5	0.86	25.86
Log Reg	6	1.38	39.66
Log Reg	7	2.07	60.34
Log Reg	8	0.95	69.83
Log Reg	9	1.55	85.34
Log Reg	10	1.47	100.00

Table 1: Predictive models on the whole data set

Figure 1 and more explicitely in Table 2, two different groups of variables and two outliers were identified. Tables 3 and 4 suggest that the best separated variables are the ones populating communities C_1 and C_2, in the sense specified in section 4. Moreover, Table 4 also suggests that the aggregate lists AL_1 and AL_2 are well choosen candidate aggregates of the corresponding communities as $\delta A_h < \delta_h$ and σA_h is small ($h \in \{1,2\}$).

Expert opinions and business experts confirm that the group of variables derived using *NetAggreg* are coherent with business practice (see e.g. Altman et al. (2005)), especially for C_1 and C_2.

C_1	C_2	C_3	C_4
Supplier target days Outside capital strucure Capital tied up	Liquidity ratio Cash ratio Equity ratio Cash flow to effective debt Liabilities ratio Result ratio	Cost income ratio	Trade payable ratio

Table 2: *NetAggreg* communities extraction result on the proposed set of financial ratios.

C_h	C_k	$\delta_{h,k}$	$\sigma_{h,k}$
C_1	C_2	0.605087116	0.04946876
C_1	C_3	0.500418	0.105093445
C_1	C_4	0.394367666	0.141355664
C_2	C_3	0.500584162	0.060114893
C_2	C_4	0.560789719	0.093720458
C_3	C_4	0.662595709	0

Table 3: Average pairwise distance between the four *NetAggreg* extracted communities and relative standard deviation.

C_h	δA_h	σA_h	δ_h	σ_h
C_1	0.201099917	0.003032157	0.341511596	0.006955172
C_2	0.234515315	0.041611965	0.33071787	0.11191681

Table 4: C1 and C2 average distance from the corresponding aggregate lists (δA_h), their average within distance (δ_h) and the corresponding standard deviations (σA_h and σ_h).

In order to assess if the groups are also relevant in terms of predictive ability, we have applied logistic regression and classification tree on C_1 and C_2. We have tested the models in terms of out of sample performance using the same proportions specified before.

On the basis of the variables in C_1, both predictive models perform better with respect to the models build on the whole data set. Table 5 reports the results in

Model	**decile**	**LIFT**	**CCR**
Tree	1	4.32	43.21
Tree	2	1.25	55.70
Tree	3	0.84	64.06
Tree	4	0.84	72.42
Tree	5	0.84	80.78
Tree	6	0.81	88.91
Tree	7	0.29	91.85
Tree	8	0.29	94.79
Tree	9	0.29	97.73
Tree	10	0.23	100.00
Log Reg	1	3.00	30.00
Log Reg	2	2.17	51.72
Log Reg	3	0.95	61.21
Log Reg	4	0.69	68.10
Log Reg	5	0.86	76.72
Log Reg	6	0.52	81.90
Log Reg	7	0.86	90.52
Log Reg	8	0.34	93.97
Log Reg	9	0.34	97.41
Log Reg	10	0.26	100.00

Table 5: Predictive models on C_1

terms of lift and cumulated captured response. As we can observe from Table 5 tree model is the best one and using the first three deciles it captures the 64.06% of the events of interest. The AUC values are equals to 0.85 for the logistic regression and 0.89 for the tree model and the percentage of correct classifications is equal to 83.5% for the logistic regression and 90% for the classification tree. Finally, we have considered C_2 as variables to predict default. Both variables are statistically significant for the logistic regression. The logistic regression and the tree models give in terms of out of sample performance interesting results. From Table 6 we underline that tree model is the best one and using the first three deciles it captures the 71.85% of the events of interest. The AUC values are equals to 0.80 for the logistic regression and 0.87 for the tree model and the percentage of correct classifications is equal to 82.5% for the logistic regression and 88% for the classification tree.

Model	decile	LIFT	CCR
Tree	1	3.17	31.70
Tree	2	2.08	52.47
Tree	3	1.94	71.85
Tree	4	0.94	81.29
Tree	5	0.72	88.51
Tree	6	0.37	92.23
Tree	7	0.37	95.94
Tree	8	0.37	99.66
Tree	9	0.03	100.00
Tree	10	0.00	100.00
Reg	1	3.10	31.03
Reg	2	1.98	50.86
Reg	3	1.72	68.10
Reg	4	1.21	80.17
Reg	5	0.60	86.21
Reg	6	0.60	92.24
Reg	7	0.34	95.69
Reg	8	0.34	99.14
Reg	9	0.09	100.00
Reg	10	0.00	100.00

Table 6: Predictive models on C_2

Our real application shows that using *NetAggreg* we are able to select sub sets of variables highly related to default estimation. Furthermore, we underline that the models built on the communities selected, perform better in terms of out of sample measures with respect to the results achieved on the whole data set.

7 Conclusions

In this paper we propose *NetAggreg*, a novel methodology for rank aggregation. We describe our proposal in a theoretical framework and we also provide an effective algorithm for rank aggregation. The implementation of *NetAggreg* is written in the statistical programming language *R* and is available on demand. On the basis of an extensive simulation activity, we prove that our approach outperforms related methods proposed in the literature. Finally, testing on real financial data shows that *NetAggreg* is a powerful approach able to improve predictive performances in credit risk analysis.
Our method is easy to implement, does not have any computational overhead and is able to isolate outliers. Future work would focus on measuring the efficacy of *NetAggreg* as a variable selection method when the variables can be interpreted as orderings.

References

Altman E. I., Brady B., Resti, A. and Sironi A. (2006). The Link Between Default and Recovery Rates: Theory, Empirical Evidence and Implications *Journal of Business*, 2005, vol. 78, no. 6

Arrow K. J. (1950) . A difficulty in the concept of social welfare. *Journal of Political Economy*, 58:328–346.

Borda J.C. (1781). Memoire sur les elections au scrutin.*Histoire de l'Academie Royale des Sciences*.

DeConde R., Hawley S., Falcon S., Clegg N., Knudsen B., Etzioni R. (2006). Combining results of microarray experiments: a rank aggregation approach. *Stat Appl Genet Mol Biol*, 5:Article 15.

Dwork C. , Kumar R. , Naor M. , Sivakumar D. (2001). Rank aggregation methods for the Web. *Proceedings of the 10th international conference on World Wide Web*

Feller W. (1971). An introduction to Probability Theory and its Applications. *vol. I Wiley, New York, NY.*

Figini S. and Giudici P. (2011). Statistical merging of rating models, *Journal of the Operational Research Society* , 62: 1067–1074

Giudici P. and Figini S. (2009). Applied data mining, Wiley London

Hand D.J. and Zhou F. (2009). Evaluating models for classifying customers in retail banking collections. *Journal of the Operational Research Society*, 61, 1540–1547.

Hand D., Tasoulis D.K., Anagnostopoulos C. and Adams N.M. (2010). Temporally-adaptive linear classification for handling population drift in credit scoring, Lechevallier, Y. And Saporta. (eds) *COMPSTAT2010, Proceedings of the 19th International Conference on Computational Statistics*, Springer, 167-176.

Kemeny J. G.(1959). Mathematics without numbers. Daedalus, 88(4):577–591.

Shili Lin (2010). Rank aggregation methods *Wiley Interdisciplinary Reviews: Computational Statistics* Volume 2, Issue 5, pages 555–570, September/October 2010

Pihur V. and Datta S. (2008). Finding cancer genes through meta-analysis of microarray experiments: Rank aggregation via the cross entropy algorithm. *Genomics*, (92):400–403.

Pons P. and Latapy M. (2006). Computing Communities in Large Networks Using Random Walks. *Journal of Graph Algorithms and Applications (JGAA)* , Vol. 10, no. 2, pp. 191–218

Shili Lin (2010). Rank aggregation methods *Wiley Interdisciplinary Reviews: Computational Statistics* Volume 2, Issue 5, pages 555–570, September/October 2010

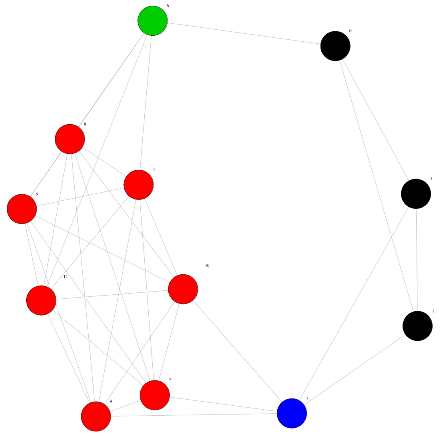

Figure 1: *RankAggreg* network along with the relative exctracted communities C1(black dots), C2 (red dots) and the two outliers variables C4 (blue dot) and C3 (green dot).

QUADERNI DEL DIPARTIMENTO DI ECONOMIA, STATISTICA E DIRITTO

n. 1/2011 Paolo Giudici, Emanuela Raffinetti, *A Gini concentration quality measure for ordinal variables,* Serie Statistica.

n. 2/2011 Silvia Figini, Lijun Gao, Paolo Giudici, *Bayesian efficient capital at risk estimation,* Serie Statistica.

n. 3/2011 Luigi Bernardi, *Tendenze dei prelievi tributari ed effetti fiscali della crisi finanziaria nell'Unione europea e in Svizzera,* Serie Economia.

n. 4/2011 Silvio Beretta, Renata Targetti Lenti, *India in the Outsourcing/Offshoring Process: A Western Perspective,* Serie Economia.

n. 5/2011 Paolo Giudici, Eleonora Lorenzini, *SMEs, e-commerce and territorial development: the experience of a "web district",* Serie Statistica.

n. 6/2011 Paola Cerchiello, *On the distribution of functionals of discrete ordinal variables,* Serie Statistica.

n. 7/2011 Silvia Figini, Lijun Gao, Paolo Giudici, *Model averaged credit risk models,* Serie Statistica.

n. 8/2011 Andrea Zatti, *Tassazione ambientale e federalismo fiscale: riflessioni e prospettive, con particolare riferimento al caso italiano,* Serie Economia.

n. 9/2011 Luisa Cutillo, Annamaria Carissimo, Silvia Figini, Network Aggregation: *A Method for rank aggregation via network evidence,* Serie Statistica.

COLLANE PRECEDENTI

QUADERNI DEL DIPARTIMENTO DI ECONOMIA PUBBLICA E TERRITORIALE

n. 1/2010 Silvio Beretta, *Variabili finanziarie ed economia globale in tempo di crisi*

n. 2/2010 Silvio Beretta, Renata Targetti Lenti, *L'India nel processo di integrazione internazionale. Dal primo al secondo* unbundling *e la posizione dell'Italia*

n. 3/2010 Margarita Olivera, *Challenges to Regional Integration in Latin America*

n. 4/2010 Italo Magnani, *Un economista liberale guarda alla economia dell'ambiente: impressioni e riflessioni*

n. 5/2010 Italo Magnani, *A cinquant'anni dalla scomparsa di Benvenuto Griziotti: Riflessioni*

n. 6/2010 Luca Mantovan**,** *Class-bias in Technology Adoption: Stagnation and Transformation of Subsistence Agriculture in the Ethiopian Northern Highlands*

n. 7/2010 Marco Missaglia, Giovanni Valensisi, *A trade-focused, post-Keynesian CGE model for Palestine*

n. 8/2010 Giovanni Valensisi, Marco Missaglia, *Reappraising the World Bank CGE model on Palestine: macroeconomic and financial issues*

n. 1/2009 Giorgio Panella, Andrea Zatti, Fiorenza Carraro, *Market Based Instruments for Energy Sustainability*

n. 1/2008 Italo Magnani, *Il pubblico e il privato nella economia della città*

n. 2/2008 Italo Magnani, *Note a margine di una recente opera sull'indirizzo sociologico della scienza delle finanze italiana*

n. 3/2008 Italo Magnani, *La riforma sociale nella formazione di Nitti economista*

n. 4/2008 Marisa Bottiroli Civardi, Renata Targetti Lenti and Rosaria Vega Pansini, *Multiplier Decomposition, Poverty and Inequality in Income Distribution in a SAM Framework: The Vietnamese Case*

n. 5/2008 Luca Mantovan, *A Study on Rural Subsistence in the Ethiopian Northern Highlands*

[per i Quaderni precedenti si rinvia a http://www-5.unipv.it/webdesed/ept/quaderni.php]

QUADERNI DI RICERCA DEL DIPARTIMENTO DI STATISTICA ED ECONOMIA APPLICATE "L. LENTI"

Carla Ge Rondi, *L'après mariage en Italie au début du XXIe siècle* (2005, n. 27)

Carla Ge Rondi, *Casalinga: popolazione attiva senza retribuzione* (2005, n. 25)

Bruno Scarpa, David Dunson, *Bayesian Methods for Searching for Optimal Rules for Timing Intercourse to Achieve Pregnancy* (2005, n. 24)

Bruno Scarpa, *Lo stress in azienda. Modelli di analisi di un'indagine per l'identificazione delle cause di stress* (2004, n. 23)

Bruno Scarpa, *La Customer Satisfaction per un'azienda di servizi informatici. Impostazione e analisi di un'indagine via web* (2004, n. 22)

[per i Quaderni precedenti si rinvia a http://www-5.unipv.it/webdesed/lenti/quaderni.php]

OSSERVATORIO DEI CONTRATTI DELLA P.A.

[Si rinvia a http://www.contratti-appalti.it/]