



Estimating discretionary accruals using a grouping genetic algorithm

Henrik Höglund*

Hanken School of Economics, Handelsplanaden 2, 65101 Vasa, Finland

ARTICLE INFO

Keywords:

Earnings management
Discretionary accruals
Genetic algorithms

ABSTRACT

A number of different models have been suggested for detecting earnings management but the linear regression-based model presented by Jones (1991) is the most frequently used. The underlying assumption with the Jones model is that earnings are managed through accounting accruals. Typically, the companies for which earnings management is studied are grouped based on their industries. It is thus assumed that the accrual generating process for companies within a specific industry is similar. However, some studies have recently shown that this assumption does not necessarily hold. An alternative approach which returns a grouping which is, if not optimal, at least very close to optimal is the use of genetic algorithms. The purpose of this study is to assess the performance of the cross-sectional Jones accrual model when the data set firms are grouped using a grouping genetic algorithm. The results provide strong evidence that the grouping genetic algorithm method outperforms the various alternative grouping methods.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

The occurrence of earnings management has been a widely studied subject for the past 30 years. One of the major challenges when examining possible earnings management is that the magnitude of it is difficult to assess. A number of different models have been suggested for detecting earnings management but the linear regression-based model presented by Jones (1991) is the most frequently used. The underlying assumption with the Jones model is that earnings are managed through accounting accruals. Typically, the companies for which earnings management is studied are grouped based on their industries. It is thus assumed that the accrual generating process for companies within a specific industry is similar. Recently, however, some studies have shown that this assumption does not necessarily hold. Dopuch, Mashruwala, Seethamraju, and Zach (2012), for example, showed that a violation of the homogeneous accrual generating process within an industry causes measurement errors. In another study Ecker, Francis, Olsson, and Schipper (2011) showed that the performance of the Jones model is improved when lagged total assets are used as a grouping variable instead of the industry membership. Even though alternative methods have been used for grouping companies when using the Jones-model, none of them has clearly outperformed the grouping-based on industry membership.

An exhaustive search for the best possible grouping is in most cases impossible considering the large number of possible combinations even with moderate size data sets. An alternative approach

that returns a grouping which is, if not optimal, at least very close to optimal is the use of genetic algorithms. Genetic algorithms have proven efficient in solving difficult problems such as the travelling salesman and the equal piles problems.

Genetic algorithms have been used in a number of accounting applications. Back, Laitinen, and Sere (1996) used a genetic algorithm to determine the optimal predictors for a neural network-based bankruptcy prediction model. A similar study was carried out by Shin and Lee (2002) when they used a genetic algorithm to generate bankruptcy prediction rules. Hoogs, Kiehl, Lacombe, and Senturk (2007) presented a genetic algorithm approach for detecting financial statement fraud. Their model successfully classified 63% of the companies that had been accused by the Securities and Exchange Commission (SEC) for improperly recognizing revenue.

The purpose of this study is to assess the performance of the cross-sectional Jones accrual model when the data set firms are grouped using a grouping genetic algorithm. The performance of the grouping genetic algorithm approach is compared with the performance of a number of other grouping techniques.

The remainder of this study is organized as follows. The basic operating principle of the linear regression-based accrual models is covered in Section 2. In Section 3 an overview of both classic and grouping genetic algorithms is given. The research design is presented in Section 4 and the results from the empirical study are presented in Section 5. Section 6 concludes the study.

2. Discretionary accrual estimation models

Several approaches for measuring the extent of earnings management have been proposed but the method that has gained the

* Tel.: +358 040 3521768.

E-mail address: henrik.hoglund@hanken.fi

widest acceptance among researchers is the linear regression-based model presented by Jones (1991). The general assumption with the Jones-model is that earnings are managed through accounting accruals. The accruals are defined as the difference between net earnings before extraordinary items and cash flows from operations. The purpose of the Jones-model is to split the total accruals of a company into non-discretionary (expected) and discretionary (unexpected) accruals. The non-discretionary accruals are accruals that the company management has no or little control over, whereas the discretionary accruals are a proxy for earnings management. In the model the reciprocal of total assets, change in revenues (ΔREV) and gross property, plant and equipment (PPE) are regressed on total accruals (TACC). The variable for change in revenues controls for current accruals such as receivables, payables and inventory whereas the variable for property, plant and equipment mainly controls for depreciation and amortization accruals. In the linear regression-based Jones-model the regression error term (ε) equals the discretionary accruals

$$\frac{TACC_t}{TA_{t-1}} = \beta_0 \frac{1}{TA_{t-1}} + \beta_1 \frac{\Delta REV_t}{TA_{t-1}} + \beta_2 \frac{PPE_t}{TA_{t-1}} + \varepsilon$$

In the original Jones-model the regression was run separately for each company using a time-series of at least 10 observations. The time-series approach has, however, several drawbacks. First, the requirement of at least 10 years of financial statement data might lead to both a survivorship bias and a selection bias (Jeter & Shivakumar, 1999). Second, the assumption that the accrual generating process of a company is stable over longer periods of time does not necessarily hold (Dopuch et al., 2012). Due to these drawbacks the time-series approach has largely been replaced by the cross-sectional approach first suggested by Dechow (1994). With the cross-sectional approach it is assumed that companies within a specific industry have a similar accrual generating process. Usually the industry is defined at a two-digit level SIC (e.g. Bartov, Gul, & Tsui, 2000; Jeter & Shivakumar, 1999). A number of recent studies have questioned whether the assumption of similar accounting generating processes within the industries is valid. Dopuch et al. (2012) showed that the assumption of a homogenous accruals generating process does not apply for several industries. Furthermore, they also showed that the violation of this assumption caused some measurement error of discretionary accruals. They do not, however, suggest any alternative method of grouping the companies when using the Jones-model. Ecker et al. (2011) used a number of different variables for grouping companies when using the Jones-model and their findings showed that the best result was achieved when lagged total assets were used.

3. Genetic algorithms

3.1. Operating principle

Genetic algorithms are an optimization technique based on models of natural selection and evolution. The fundamental principles of the genetic algorithm were first presented by Holland (1975). The starting point when using genetic algorithms is a population consisting of a certain number of chromosomes (individuals), where the chromosomes represent valid solutions to the problem. Once the population size has been determined, the initial population is usually randomly generated. The size of the initial population depends on the complexity of the problem. A genetic algorithm with a smaller population is faster but at the same time the risk of premature convergence increases (Koljonen, Mannila, & Wanne, 2007). Once the initial population has been generated, the fitness of each chromosome is evaluated. Based on the fitness values, parent chromosomes are selected from the initial

population to form new chromosomes to the next generation through a breeding process. The most commonly used method for selecting the parent chromosomes is the roulette wheel selection (Butun, Erfidan, & Urgun, 2006) in which a proportion of the wheel is assigned to each chromosome based on their fitness values. The larger the proportion of the wheel, the higher the probability of getting selected. A central part of the breeding process is the cross-over function. The parent chromosomes are combined using a cross-over function to form new chromosomes. A commonly used cross-over technique is the single-point cross-over where a single cross-over point is selected in the chromosome. The part beyond the cross-over point is then swapped between the two parent chromosomes. Once the next generation has been formed, the new chromosomes are subjected to random mutation. The purpose of the mutation is to prevent the premature convergence of the genetic algorithm. After the mutation process has been completed, the fitness of the chromosomes in the new generation is assessed and the selection and cross-over procedures start all over. The evolution process is stopped when a satisfactory solution has been reached or when some other predetermined condition has been met. To prevent from losing the best chromosomes during the cross-over and mutation operations, elitism can be employed. Elitism means that the best or a few of the best chromosomes are directly copied from the previous to the next generation. Elitism can improve the performance of the genetic algorithm but there is also a risk that it leads to premature convergence.

3.2. Grouping genetic algorithms

Classic genetic algorithms do generally not perform well on grouping problems. The main problems are a high redundancy among the population chromosomes and a context insensitivity of the cross-over function (Falkenauer, 1996). To deal with these shortcomings of the classic genetic algorithm, Falkenauer (1992) suggested a modified genetic algorithm suited for grouping problems. The general difference between the two types of genetic algorithms is that in classic genetic algorithms the focus is on individual items whereas in grouping genetic algorithms the focus is on groups of items.

In a grouping genetic algorithm the chromosome is divided into a number of groups which in turn contain a number of individual items. The values of the individual items as such are not important. Instead, it is the group membership of the items that is of value. In the grouping genetic algorithm suggested by Falkenauer, the chromosome length is variable. The grouping genetic algorithm can, however, also be used with fixed length chromosomes when required by the nature of the problem (e.g. the equal piles problem, Falkenauer, 1996).

The cross-over function for the grouping genetic algorithm is illustrated in Fig. 1. First, two cross-point are selected randomly in both chromosomes involved in the cross-over function. To generate the first new chromosome, the genes between the cross-points in the second old chromosome are injected after the first cross-point in the first old chromosome. At this point some items occur in two groups. The groups coming from the first old chromosome and that have items also occurring in the groups coming from the second old chromosome are deleted. After removing these groups, some items might not be present in the remaining groups. These items are allocated to new groups using various methods. The missing items can, for example, form a new group or they can be distributed over a certain number of new groups. Once the first new chromosome has been created, a second new chromosome is formed using the same two old chromosomes but in a reverse order.

As with the cross-over function, the mutation operator for grouping genetic algorithms works on groups rather than items.

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات