A Mathematical Programming Approach for Imputation of Unknown Journal Ratings in a Combined Journal Quality List

Jinhak Kim

Mitchell College of Business, University of South Alabama, Mobile, AL 36688, USA,

e-mail: jinhakkim@southalabama.edu

Young Woong Park [1]

Ivy College of Business, Iowa State University, Ames, IA 50011, USA,

e-mail: ywpark@iastate.edu

Alvin J. Williams

Mitchell College of Business, University of South Alabama, Mobile, AL 36688, USA,

e-mail: awilliams@southalabama.edu

Abstract

The quality of faculty scholarship and productivity is one of the primary measures for faculty evaluation in most academic institutions. Due to the diversity and interdisciplinary nature of modern academic research fields, it is increasingly important to use journal quality lists, with journal ratings, that offer credible measures of the worth of faculty scholarship. Despite the existence of such metrics, journal lists, by their very nature, exclude some well-recognized journals. Consequently, academic institutions expend inordinate resources to assess the quality of unrated journals appropriately and equitably across disciplines. The current research proposes mathematical programming models as a path to determining unknown ratings of multiple journal quality lists, using only their known rating information. The objective of the models is to minimize the total number of instances where two journals are rated in opposite order by two different journal quality lists. Computational results based on journal quality list data in `https://harzing.com/` indicate that the proposed methods outperform existing imputation algorithms with most realistic test datasets in terms of accuracy, root mean square error, and mean absolute deviation.

1 Introduction

Academic institutions continuously strive to enhance their capacity to assess faculty productivity and scholarship. Credible measures of scholarship performance are at the core of the research enterprise of most institutions, and the quality of journals is one of the most crucial surrogate measures of faculty value to institutions. Despite its importance, the journal quality evaluation

[1] Corresponding author

process is debatable and somewhat contentious, in large measure due to the challenge of maintaining objectivity and impartiality across a broad range of research disciplines and cultures.

Many studies try to address this question by analyzing citation data or surveying discipline-specific experts. For example, Olson (Olson, 2005) and Hult *et al.* (Hult, Neese, & Bashaw, 1997) surveyed faculty in operations management and marketing, respectively, to rank journals in their field. There are also studies based on citation data analysis, such as the h-index (Moussa & Touzani, 2010) and PageRank (Cheang, Chu, Li, & Lim, 2014). However, while these studies can be useful, their limitations offer challenges in justifying and generalizing analytical and survey results.

Given the challenges and peculiarities of schools and disciplines, many institutions have embraced one or more of the journal quality indices published by reliable providers. For business schools in particular, the Australian Business Deans Council (ABDC), Association of Business Schools (ABS), Erasmus Research Institute of Management (EJL), European Journal of Information Systems (EJIS), and High Council for Evaluation of Research and Higher Education (HCERES) constitute a set of the most commonly used indices across a range of institutional varieties. These rating providers collectively yield journal quality lists. Each journal quality list defines its own rating categories and provides a list of journals with their ratings. For example, as rating providers, ABDC categorized 2,777 journals into four categories (A*, A, B, and C) in their 2016 interim review version and ABS used five categories (1*, 1, 2, 3, and 4) to rate 1,390 journals in the 2015 version.

While journal quality lists are used globally by many universities, there is less than full confidence in the capacity of these lists to ably reflect the increasingly exacting criteria for scholarly success. These lists are problematic, in part because they are not exhaustive and fail to satisfy the varying demands of increasingly prevalent interdisciplinary research. For example, ABDC and ABS are well-recognized journal quality lists for business-related disciplines. Among 731 selected journals listed by ABS in `https://harzing.com/`, 35 journals are not listed by ABDC. Those missing journals span many subject areas and all rating categories. Selected disciplinary journals that are not listed in ABDC but are listed in ABS include *Mathematical Programming*, *SIAM Journal on Optimization*, and *IEEE Transactions on Engineering Management*. The omitted journals are reflective of the growing culture of interdisciplinary research in the academy. Without credible means of valuing the worth of these scholarly contributions, an increasingly larger number of researchers risk being subjected to less than satisfactory means of research evaluation.

As a partial solution to this conundrum, past research has focused on developing predictive models to estimate the ratings of the unrated journals in a given discipline. Most such efforts aim to recover the underlying rating formula, using known factors, such as the acceptance rate and impact factors. In other words, the prediction for an unrated journal is dependent on the factors that are used when the model is constructed. The efficacy of such models is based on the consensus that those specific factors are inclusive to explain the target journal rating, which is generally not the case.

The current research effort posits an approach highlighting a model that predicts unknown ratings based on known ratings from multiple journal quality lists. This methodology inherently takes into account all factors considered to determine journal ratings by selected quality lists. More specifically, the initial rating data are constructed by aggregating certain journal quality lists as demonstrated in Figure 1. Each column and row represents a journal quality list and a journal, respectively. Ultimately, the goal is to fill empty cells using known entries.

ISSN	Journal	Subject area	Ejis 2007 [4,3,2,1]	EjisCl 2007 [4,3,2,1]	Wie 2008 [A+, A]	HEC 2011 [A, B+, B, C]	UQ 2011 [1,2,3,4]	ABS 2015 [4*,4,3,2,1]	Vhb 2015 [A+, A, A/B, B, B/C, C]	Abdc 2016 [A*,A,B,C]	Ess 2016 [0+, 0, 1, 2, 3]
0004-9530	Australian Journal of Psychology	Psychology					3	1		B	
0013-1881	Educational Research	PSM	3	2							
0090-0036	American Journal of Public Health	Economics		4					B	A*	
0048-7333	Research Policy *	Economics	3	3	A	A	1	4	A	A*	1
0021-9436	Journal of Business Communication	Comm	1	1			4			C	2
0144-3585	Journal of Economic Studies	Economics	2	2				2		B	
0927-538X	Pacific Basin Finance Journal	F&A	1	1		C	2	2		A	
0025-5610	Mathematical Programming	OR,MS,POM	4	4	A	A		4	A		1
1350-4851	Applied Economics Letters	Economics	1	1	A			1		B	2
1361-9209	Transportation Research Part D: Transport & Environment	OR,MS,POM	4	3				3	B	A	1

Figure 1: The header of a random portion of data provided in https://harzing.com (Harzing, 2017)

Data imputation is the task of estimating the missing entries of partially observed data. It has been studied since the 1970s in the context of statistical inferences. While the earlier applications of data imputation focused primarily on statistical model building with incomplete data, a variety of interesting uses have been recognized in the last decade. For example, Netflix, Inc. held an open competition in 2006, called the Netflix Prize Challenge (Bennett, Lanning, & etc., 2007), to predict unknown movie ratings for films based on known customer rating data, offering a one-million-dollar grand prize. They use these techniques to develop an intelligent movie recommender system.

Some estimation schemes using a single quality list have been applied to journal rating imputation problems. These approaches aim to construct predictive models based on external factors, such as Cabell's acceptance rates and impact factors. Mingers and Harzing (Mingers & Harzing, 2007) used regression-based methods to impute missing values. Tse (Tse, 2001) proposed a mathematical programming model to estimate the coefficients of a rating prediction model. Despite the simplicity of Tse's model, the factors influencing ratings and their effects may vary by quality lists, causing a fairness issue in the evaluation process. The fairness issue is in part because criteria used for faculty scholarship may not match those of a particular quality list.

A clear example of the stark differences resulting from dissimilar criteria is evident in the journal *Basic and Applied Social Psychology*. Three journal rating systems yielded very different results for this same journal. The EJIS assigned its second-best rating out of four categories, ABS gave its lowest rating for the same journal, and ABDC did not have the journal listed. Since these models consider only one quality list in model building, the prediction for a particular journal is meaningful only when there is a consensus among scholars in the journal's disciplinary area regarding the explanatory factors and their weights on rating determination. Furthermore, the prediction may not be reliable if the model is not a good fit for the data, thus, decreasing predictive accuracy.

An alternative to avoid these drawbacks is to estimate the unknown ratings based on multiple quality lists without directly using journal characteristics in the analysis. Based on the incomplete dataset constructed by aggregating multiple quality lists, the goal is to impute the unknown cells in the dataset, such as the empty cells in Figure 1. When the estimation is performed based on the sampling rating distributions, multiple imputation techniques can be applied to the problem. Multiple imputation techniques have been studied extensively over the past several decades (Rubin, 2004; Schafer, 1999; Sterne et al., 2009). Multiple imputation accounts for uncertainty by generating multiple, plausible values for each missing entry and pooling those values to obtain a single outcome. It is one of the most popular imputation tools and has been integrated within many statistical packages, including R, S-Plus, Stata, SAS, and SPSS (Van Buuren, 2012). One example of integration is the R package `mice` (multiple imputation by chained equation) (Buuren & Groothuis-Oudshoorn, 2011) that provides various multiple imputation algorithms. Although journal characteristics are not directly reflected in the model, they are inherent in the rating in-

formation in each quality list. Therefore, this approach indirectly considers all of the underlying factors that affect ratings. Despite its potential merits, multiple imputation can only be applied to data for which unknown entries are *missing at random*. According to Allison (Allison, 2000), data are identified as *missing at random* if the probability of missing data on a particular variable Y is dependent on other observed variables, but not on Y itself. This assumption is not applicable for journal ratings because each rater has criteria to determine inclusion of a journal in the quality list; hence, it is reasonable to conclude that the missing journal ratings are not random. In general, this assumption is not satisfied for most missing data. Allison (Allison, 2017) provides specific evidence satisfying this assumption. More recently, Bertsimas et al. (Bertsimas, Pawlowski, & Zhuo, 2018) used an optimization approach to impute missing data with mixed continuous and categorical variables. The authors developed non-convex optimization models with different objective functions: K-nearest neighbor, support vector machine, and optimal decision trees. They derived fast first-order methods that obtain high-quality solutions by approximating the non-convex objective functions.

Matrix completion is an alternative approach to completing missing data entries when all entries are numerical. Matrix completion has been researched extensively in the optimization and computer science communities. This approach is typically modeled as an optimization formulation for a matrix that minimizes the rank and matches the existing known entries. Specific applications and examples of optimization formulation and selective solution methods for matrix completion are highlighted in (Hastie, Mazumder, Lee, & Zadeh, 2015; Candès & Recht, 2009; Candès & Tao, 2010; Jain, Netrapalli, & Sanghavi, 2013; Recht, 2011). In general, matrix completion algorithms return fractional imputation and, in order to be used for ordered categorical data, post-processing steps such as rounding of fractional imputed values are necessary. Despite this limitation, it is still a useful approach, and we included a matrix completion algorithm in our computational experiment as a benchmark.

The current paper proposes applying mixed integer programming models to impute the missing ratings in the data. Like multiple imputation algorithms, the proposed models do not use explanatory factors directly in determining the ratings of each quality list but only use the known ratings in the list. Furthermore, the models do not require a randomness assumption on the data. Unlike traditional objectives, such as minimizing errors or maximizing likelihood, in our formulation, unknown ratings are determined by minimizing the number of instances where two journals are rated in reverse order by two different quality lists. This phenomenon is characterized as an *upset*. For example, in a partial data display in Figure 1, the *American Journal of Public Health* has a higher rating than *Research Policy* in EJIS 2007, but a lower rating in VHB 2015, the 2015 version of the quality list by the Association of University Professors of Business in German-speaking countries. Therefore, our models can be applied to a broader range of data where entries of each column are categorized by a finite ordered set and an upset can be regarded as an unusual conflict.

Although the target problem is explicitly modeled using a mixed integer program, the challenge is the computational complexity of solving a mixed integer program, as the solution time can be exponential in the dimensions of the data. Thus, to counter some of these challenges, we also consider its linear programming relaxation. Theorem 3.2 demonstrates the existence of an integral alternate solution to the linear relaxation that enables us to directly use the solution to determine missing ratings. Comparing current models with solutions provided by imputation algorithms implemented in the R package `mice` (Buuren & Groothuis-Oudshoorn, 2011), the `softImpute` (`SI`) algorithm in the R package `softImpute` (Hastie & Mazumder, 2015), and the baseline row mean (RMean) imputation show that our optimization models outperform the benchmark methods.

Furthermore, solution performance (accuracy, root mean square error (RMSE), and mean absolute deviation (MAD), proportion of upsets (%Upset)) of linear relaxation is similar to that of a mixed integer program, thus the linear relaxation model is a good alternative in practice, resolving the issue of computation time.

The impetus for our model development for journal rating imputation emanates from problems identified in the extant literature. One such problem is rank aggregation, which involves combining rankings from various sources and creating a single ranking (Dwork, Kumar, Naor, & Sivakumar, 2001). Given ordered lists of multiple judges, the goal is to create a single ordered list that reflects the evaluations of multiple judges. Studies have demonstrated the utility of mathematical programming models for rank aggregations. Raisali *et al.* (Raisali, Hassanzadeh, & Milenkovic, 2013) used integer programming for weighted rank aggregation. Pedings *et al.* (Pedings, Langville, & Yamamoto, 2012) used integer programming to minimize violations for rank aggregations, and Kou *et al.* (Kou, Chang, Zheng, & Zha, 2010) formulated quadratic programming to aggregate rankings from multiple sources. However, the rank aggregation problem has a different goal from the focus of the current method. Our goal is to impute the missing ratings rather than create a single combined ranking list. Another problem that explicates our model-building is ranking determination in tournaments. More specifically, assume there are n players, and each one plays against the other $n-1$ players exactly once. The goal is to rank all n players based on game results. However, it is not straightforward to create rankings for which everybody agrees. One possible consideration is to use the concept of an *upset*, an instance where a low-ranked player beats a high-ranked player. At this point, the goal is to find the ranking that minimizes the total number of upsets. Solution methods for the minimum feedback arc set on tournaments can be used (Kenyon-Mathieu & Schudy, 2007) to solve this problem. While the ranking problem in tournaments aims to determine a single ranking for n players, the objective of our journal rating imputation problem is to determine unknown ratings in multiple journal quality lists. Furthermore, while each pair of players can be ordered in the tournament setting, two journals are not necessarily comparable in our problem because (1) there are multiple rating providers that potentially rate those journals, (2) some rating providers may not rate the journals, and (3) defining a comparison rule in this structure is not straightforward. Despite the differences in their structure, we use a similar concept of upsets to define our objective function. More specifically, we define an upset as an instance where two journals are rated in opposite order by two rating providers and the objective function is to minimize the sum of upsets across all journals and all rating providers.

Models developed in the current research are multifunctional, with potential applicability to similar types of data. However, the current imputation models are not generic solutions to any data imputation problem, since its usage is restricted to categorical data and the algorithm is not designed for data in which upsets occur frequently. For example, in many instances of journal rating data, the proportions of upsets are no more than 9% among all possible pairs of journals and pairs of rating providers. For this reason, the primary focus is on imputing the missing journal ratings in multiple journal quality lists.

How does the current research effort assist in closing some of the "gaps" associated with rating the quality of unknown journals? Specifically, our potential contributions can be summarized as follows:

1. We propose a mixed integer programming model to impute missing values in ordered categorical data. The model is applied to the journal rating data, yet it can also be used for other general categorical data where an upset can be regarded as an unusual conflict.

2. The proposed mixed integer programming model and the associated Linear Program (LP) relaxation outperform existing multiple imputation algorithms. The mixed integer programming model achieves the highest accuracy for the most realistic test datasets in the computational experiment, while the LP relaxation balances accuracy and time.

3. By applying our models to impute missing journal ratings, we provide a more reliable and objective method to rate journals that are not listed in the ratings by the user's preferred and trusted rating providers.

4. From both implementation and managerial perspectives, the current research contributes to higher levels of confidence and credibility in the journal rating process.

The remainder of the paper is organized as follows. In Section 2, the significance and criticality of the challenges of journal ratings in faculty evaluation procedures are highlighted. In Section 3, we present and discuss the models posited in the current paper. Section 4 presents the settings and results of the computational experiments with our models and compares them to the results of various other approaches.

2 Significance of the Problem: A Practical Perspective

Credible journal rating systems have considerable practical utility to an array of audiences (markets), both internal and external to the institutions. Internally, the credibility of the journal rating system adds to the integrity of the faculty performance evaluation system. To the extent faculty take "ownership" of the faculty performance evaluation effort, with journal rankings being a central plank, there is likely to be increased satisfaction with the results of the evaluation process. Concomitantly, administrators leading faculty performance evaluation initiatives are likely to have more confidence in the evaluation system, if there are journal rating methodologies that are trusted and generally accepted by a large cadre of disciplinary scholars. Thus, from the perspective of internal constituents, the proposed journal rating methodology allows users of the system to have an additional measure of faith in the likelihood that the ratings assigned to journals adequately and accurately reflect quality.

Externally, journal ratings offer a core group of benefits to a range of stakeholders. First, institutional governing bodies can be more confident that quality research is a key output of the organization. As governing boards of higher education respond to increasingly challenging inquiries from stakeholders about the real value/cost of academic research, the proposed system of journal ratings may engender additional confidence in the validity of rating systems as surrogate indicators of the quality of research performance. As a comparative tool, the proposed journal rating scheme allows assessments across time frames, institutions, and disciplines. Second, the proposed method of rating journals offers external branding benefits. When recruiting faculty, administrators, students, and donors, branding becomes a critical element in the decision mix. Publishing in highly ranked journals becomes a proxy for institutional quality. As universities become increasingly research intensive, there is even more urgency to devise tools, methods, and systems that enhance and enrich our capacities to assess, measure, and evaluate more adroitly the value of the research/scholarly enterprise for both internal and external constituents. In the absence of other more or less objective criteria for performance evaluation, credible journal rankings provide a measure of organizational distinctiveness, quality, and relevance.

3 Model Description

This section presents a mixed integer linear program (MILP) model that finds the unknown ratings of incomplete journal rating data obtained by combining multiple journal quality lists. The rows and columns of the data matrix represent the journals and rating providers, respectively, and we assume that each journal in the data is rated by at least one rating provider. Each entry of the data is the rating for the corresponding journal rated by the corresponding rating provider. We refer to this rating data as the *combined quality list*. Obviously, the data contain several missing entries. See Harzing's website (Harzing, 2017) to find an example of a combined quality list. By setting the unknown entries of the data as decision variables, the optimization model aims to find missing ratings that minimize the number of upsets, instances where two journals are conversely rated by two different rating providers.

For clarity, the index sets and constants are defined prior to discussing the model specification. Given a combined quality list, define

- $\mathcal{L}$: the set of rating providers in the combined quality list
- $\mathcal{N}$: the set of journals in the combined quality list
- $\mathcal{L}_i$: the set of rating providers who rate journal $i \in \mathcal{N}$
- $\mathcal{N}_l$: the set of journals rated by rating provider $l \in \mathcal{L}$
- $\mathcal{C}_l = \{1, \cdots, c_l\}$[2]: the set of rating scales for rating provider $l \in \mathcal{L}$, where c_l is the number of rating categories by rating provider $l \in \mathcal{L}$. The lower the rating scale is, the better quality it represents. To avoid trivialities and reflect the nature of ratings in the model, we assume that $c_l > 1$.
- $r_i^{(l)} \in \mathcal{C}_l$: the known rating of journal $i \in N_l$ by rating provider $l \in \mathcal{L}$
- $\mathcal{N}_l^+ = \{(i,j) \in \mathcal{N}_l \times \mathcal{N}_l : r_i^{(l)} < r_j^{(l)}\}$: the set of pairs (i,j) of journals, where journal i has a superior rate than journal j by rating provider $l \in \mathcal{L}$
- $\mathcal{N}_l^0 = \{(i,j) \in \mathcal{N}_l \times \mathcal{N}_l : r_i^{(l)} = r_j^{(l)}\}$: the set of journal pairs, where journals i and j have the same ratings by rating provider $l \in \mathcal{L}$

Based on the notations, the (i,l)-component of the data matrix is $r_i^{(l)}$ if the journal $i \in \mathcal{N}$ is rated by $l \in \mathcal{L}$ and missing otherwise. We next introduce primary decision variables $x_i^{(l)}$, which are (imputed) ratings of journal i by rating provider l. By adding constraints $x_i^{(l)} = r_i^{(l)}$ for all $l \in \mathcal{L}$ and $i \in \mathcal{N}_l$ and $x_i^{(l)} \in \mathcal{C}_l$ for all $i \in \mathcal{N}$ and $l \in \mathcal{L}$, the values of the variables are consistent with existing rating information and rating scales. We next introduce a mathematical definition of an upset using the above information. For a fixed $i < j \in \mathcal{N}$ and $l_1 < l_2 \in \mathcal{L}$, a quadruple (i,j,l_1,l_2) is called an *upset* if $(x_i^{(l_1)} - x_j^{(l_1)})(x_i^{(l_2)} - x_j^{(l_2)}) < 0$. The inequality means that two journals i and j are reversely rated by rating providers l_1 and l_2. Notice that upsets are defined only for indices $i < j \in \mathcal{N}$ and $l_1 < l_2 \in \mathcal{L}$ to prevent an upset being counted more than once. We next introduce binary variables $z_{ij}^{(l_1,l_2)}$ for $i < j$ and $l_1 < l_2$ to model upsets as follows:

$$z_{ij}^{(l_1,l_2)} = \begin{cases} 1 & \text{if } (i,j,l_1,l_2) \text{ is an upset} \\ 0 & \text{Otherwise} \end{cases} \tag{3.1}$$

[2]Without loss of generality, we assume that each rating provider uses the numeric scales of the form $\{1, \ldots, c\}$, where c is the number of rating categories.

Then the objective function is

$$\sum_{i<j\in\mathcal{N}}\sum_{l_1<l_2\in\mathcal{L}} z_{ij}^{(l_1,l_2)}.$$

We will show later that (3.1) is implemented in our MILP formulation. To this end, we introduce another set of auxiliary variables as follows:

$$z_{ij}^{(l)} = \begin{cases} 1 & \text{if } x_i^{(l)} < x_j^{(l)} \\ 0 & \text{Otherwise} \end{cases}, \tag{3.2}$$

so that (i, j, l_1, l_2) is an upset if and only if either $z_{ij}^{(l_1)} = z_{ji}^{(l_2)} = 1$ or $z_{ij}^{(l_2)} = z_{ji}^{(l_1)} = 1$.

We next present our MILP formulation as follows:

$$\begin{aligned}
\min \quad & \sum_{i<j\in\mathcal{N}}\sum_{l_1<l_2\in\mathcal{L}} z_{ij}^{(l_1,l_2)} \\
\text{s.t.} \quad & z_{ij}^{(l_1,l_2)} \geq z_{ij}^{(l_1)} + z_{ji}^{(l_2)} - 1, && i<j\in\mathcal{N},\ l_1<l_2\in\mathcal{L}, && \text{(3.3a)}\\
& z_{ij}^{(l_1,l_2)} \geq z_{ij}^{(l_2)} + z_{ji}^{(l_1)} - 1, && i<j\in\mathcal{N},\ l_1<l_2\in\mathcal{L}, && \text{(3.3b)}\\
& z_{ij}^{(l)} \geq \frac{1}{c_l - 1}(x_j^{(l)} - x_i^{(l)}), && i\neq j\in\mathcal{N},\ l\in\mathcal{L}, && \text{(3.3c)}\\
& x_i^{(l)} = r_i^{(l)}, && i\in\mathcal{N}_l,\ l\in\mathcal{L}, && \text{(3.3d)}\\
& z_{ij}^{(l)} = 1, z_{ji}^{(l)} = 0, && (i,j)\in\mathcal{N}_l^+, && \text{(3.3e)}\\
& z_{ij}^{(l)} = z_{ji}^{(l)} = 0, && (i,j)\in\mathcal{N}_l^0, && \text{(3.3f)}\\
& x_i^{(l)} \in \{1,\ldots,c_l\}, && i\in\mathcal{N}, l\in\mathcal{L}, && \text{(3.3g)}\\
& z_{ij}^{(l)} \in \{0,1\}, && i,j\in\mathcal{N},\ l\in\mathcal{L}, && \text{(3.3h)}\\
& 0 \leq z_{ij}^{(l_1,l_2)} \leq 1, && i<j\in\mathcal{N},\ l_1<l_2\in\mathcal{L} && \text{(3.3i)}
\end{aligned}$$

Constraints (3.3a) and (3.3b) are added to implement (3.1) and the constraints (3.3c) is for (3.2). These constraints are commonly used in linearizing product terms of binary variables. See Theorem 3.1 for justifications of the implementations. By adding constraints (3.3d)–(3.3f), and (3.3g), the values of decision variables are consistent with existing rating information. Constraints (3.3g)–(3.3h) require integrality. Constraint (3.3i) is a relaxation of the original integrality constraint $z_{ij}^{(l_1,l_2)} \in \{0,1\}$. The relaxation does not affect the integrality of $z_{ij}^{(l_1,l_2)}$ components of optimal solutions of (3.3), which will be demonstrated in Theorem 3.1.

We show in the following theorem that any optimal solution (x^*, z^*) to the MILP model (3.3) satisfies (3.1), implying the integrality of $(z^*)_{ij}^{(l_1,l_2)}$.

Theorem 3.1. *Let (x^*, z^*) be an optimal solution to* (3.3)*. Then,*

$$(z^*)_{ij}^{(l_1,l_2)} = \begin{cases} 1 & \text{if } (i,j,l_1,l_2) \text{ is an upset in the imputed data } x^* \\ 0 & \text{Otherwise} \end{cases}$$

□

Proof. See Appendix A for a proof. □

In general, a mixed integer program is expensive to solve to optimality, depending heavily on the numbers of integer variables and constraints in the formulation. Notice that the MILP (3.3) has $O(|\mathcal{N}||\mathcal{L}|)$ general integer variables, $O(|\mathcal{N}|^2|\mathcal{L}|)$ binary integer variables, $O(|\mathcal{N}|^2|\mathcal{L}|^2)$ continuous variables, and $O(|\mathcal{N}|^2|\mathcal{L}|^2)$ constraints. Thus, the problem is intractable when $|\mathcal{N}|$, $|\mathcal{L}|$, or the number of missing entries are large. We can partially resolve this issue by reducing the size of the problem without sacrificing the model characteristics. Consider two journals $i_1, i_2 \in \mathcal{N}$ and assume that they are identically rated by an identical set of rating providers. In other words, the rows with respect to i_1 and i_2 are identical which can be written in our notation that $\mathcal{L}_{i_1} = \mathcal{L}_{i_2}$ and $r_{i_1}^{(l)} = r_{i_2}^{(l)}$ for $l \in \mathcal{L}_{i_1} (= \mathcal{L}_{i_2})$. It is clear that if (i_1, j, l_1, l_2) is an upset, then so is (i_2, j, l_1, l_2). Therefore, by retaining only one representative row amongst the clones, we still maintain the characteristics of the previous model provided that a reasonable adjustment on the objective function is made. The optimal ratings for the removed rows can be easily recovered by assigning the ratings of the representative row. The total number of upsets also can be easily recovered because an upset (i, j, l_1, l_2) in the optimal solution for the modified data is a representative of the set of upsets (i', j', l_1, l_2) in the optimal solution for the original dataset where row i (resp. j) and i' (resp. j') are identical. Since the number of such upsets (i', j', l_1, l_2) is the product of the number of clones of i and j, the number of upsets in the original data corresponding to an upset (i, j, l_1, l_2) in the modified data is

$$(\#\text{ of clones of the row } i) \times (\#\text{ of clones of the row } j).$$

More specifically, let u_i for $i \in \mathcal{N}$ be the number of journals including itself that have the same known and missing ratings as journal $i \in \mathcal{N}$. As we have discussed earlier, the new objective function is simply the sum of $u_i u_j z_{ij}^{(l_1,l_2)}$ over all distinct pairs of journals (i, j) and pairs of rating providers (l_1, l_2) in the modified data. To present a reduced MILP formulation, we define the notations as follows:

- $\bar{\mathcal{N}}$: the set of journals in the combined quality list with non-duplicate rating patterns
- $\bar{\mathcal{N}}_l$: the set of journals rated by rating provider $l \in \mathcal{L}$
- $\bar{\mathcal{N}}_l^+ = \{(i, j) \in \bar{\mathcal{N}}_l \times \bar{\mathcal{N}}_l : r_i^{(l)} < r_j^{(l)}\}$: the set of journal pairs where journal i has a superior rate than journal j by rating provider $l \in \mathcal{L}$
- $\bar{\mathcal{N}}_l^0 = \{(i, j) \in \bar{\mathcal{N}}_l \times \bar{\mathcal{N}}_l : r_i^{(l)} = r_j^{(l)}\}$: the set of journal pairs where journals i and j have equivalent ratings by rating provider $l \in \mathcal{L}$

Then, a reduced model, which we denote by **MILP**, is obtained by replacing $\mathcal{N}, \mathcal{N}_l, \mathcal{N}_l^+, \mathcal{N}_l^0$ in (3.3) with $\bar{\mathcal{N}}, \bar{\mathcal{N}}_l, \bar{\mathcal{N}}_l^+, \bar{\mathcal{N}}_l^0$, respectively. See (B.1) of Appendix B for the full description of **MILP**.

Compared to (3.3), **MILP** has a significantly reduced number of variables and constraints. If there are 20% duplicate rows in $\mathcal{N}$, which implies $|\bar{\mathcal{N}}| \approx 0.8|\mathcal{N}|$, then the numbers of variables and constraints of **MILP** are approximately 64% of (3.3).

Nevertheless, the numbers of variables and constraints still increase quadratically as the number of non-duplicate journals $|\bar{\mathcal{N}}|$ or rating providers $|\mathcal{L}|$ increases. Furthermore, a mixed integer program is not solvable in polynomial time unless P = NP. We observed from our computational experience with small test data (412 journals and 7 rating providers) that **MILP** have several million variables and constraints and it took several hours to solve it to optimality. For this reason, we consider the LP relaxation of **MILP** by relaxing the integrality constraints $x_i^{(l)} \in \{1, \ldots, c_l\}$ and $z_{ij}^{(l)} \in \{0, 1\}$ with their continuous relaxations $1 \le x_i^{(l)} \le c_l$ and $0 \le z_{ij}^{(l)} \le 1$, respectively. We denote the resulting formulation by **LP**. See (B.2) of Appendix B for the full description of **LP**.

Although **LP** can be solved much faster than **MILP**, the optimal solution of a linear program is fractional in general, causing a rounding issue when converting a fractional rating to an integral rating that matches one of the rating scales used by the corresponding rating provider. Throughout our computational experiments, we observe that the optimal solution of **LP** has integer values for x_l^l components of optimal solutions, while the values of z_{ij}^l and $z_{ij}^{l_1,l_2}$ at optimal solutions are fractional. It is also observed that, in many test datasets, optimal journal ratings obtained by **LP** formulation show similar performance as to those by **MILP** formulation.

Remark 1. The construction of multiple imputation algorithms is based on the missing at random (MAR) assumption, and it is required to obtain unbiased estimators for missing entries. Therefore, the application of multiple imputation algorithms for data that does not satisfy the MAR assumption results in lack of statistical justifications on the imputation, and thus the imputation is unreliable. On the other hand, our models do not require this assumption because the models are not based on theoretical distributions of entries given existing rating information. Our models are deterministic optimization models that impute missing ratings by penalizing occurrences of upsets. Therefore, although the models use certain information about the existing ratings, the imputation does not require the knowledge of the distribution and the randomness assumption.

In the remainder of the section, we prove the existence of an optimal solution whose rating components are integral.

Theorem 3.2. *There exists an optimal solution to* **LP** *whose* $x_i^{(l)}, l \in \mathcal{L}, i \in \mathcal{N}$ *components are integral.* □

Proof. See Appendix A for a proof. □

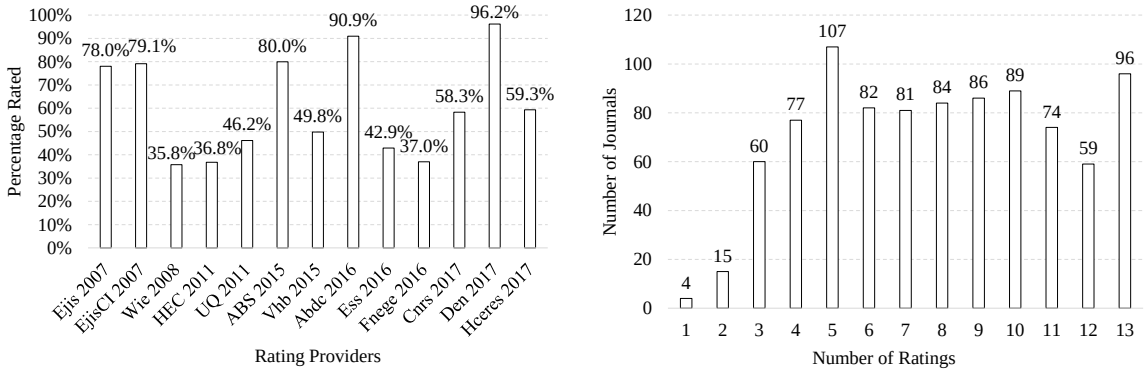
4 Computational Experiments

In this section, we present computational experiments on the journal rating data to show the performance of our models compared with benchmark algorithms. The code of the proposed algorithms and the data files used in the experiment are available in the online supplement. We use the 60th edition of the journal rating data from `https://harzing.com/` (Harzing, 2017) published on September 14, 2017. It is a combined quality list of 13 rating providers and 914 journals. We exclude one of the rating providers, EJL 2016, because the rating categories are not comparable. The resulting data with 13 remaining rating providers has 11,882 total entries, with 4,660 missing entries. Figure 2 presents the summary statistics of the original data. Figure 2(a) shows proportions of the journals evaluated by each rating provider that range from 35.8% (WIE 2008) to 96.2% (DEN 2017). Figure 2(b) illustrates how many journals are rated and by how many rating providers. For example, the last bar indicates that 96 journals are rated by all 13 rating providers, and the first bar shows that there are four journals rated by only one rating provider. We observe from the figure that each journal is rated by at least one rating provider, satisfying the assumption that the incomplete data have no empty rows.

We use two different approaches to generate test datasets. In the first approach, (1) we extracted submatrices of the original data matrix that do not contain any missing entries. This submatrix is referred to as a *complete submatrix*. We use MILP formulations to obtain three complete subdata. See Appendix C for a more detailed description of the MILP formulations and their derivation. (2) For each complete subdata, we generate several test datasets by artificially removing randomly chosen entries with various missing rates to observe performances in different missing scenarios.

In the second approach, we aim to maintain all the journals and rating providers of the original Harzing's dataset when we generate the test datasets. Similarly to 10-fold cross validation, we artificially remove 10% of the non-missing entries across all rating providers to generate a test dataset. The only difference from the original 10-fold cross validation is that the 10 sets of removed entries do not partition the original non-missing entries. That is, we allow the subsets to intersect. We adopted this setting because some journals are rated by a few rating providers; hence, some of the test datasets may include a journal that is not rated by any rating providers when classic 10-fold cross validation is applied, violating the assumption of our models. The detailed procedure of generating test datasets is presented in Section 4.1. Using the generated test datasets, we run our models (**MILP** and **LP**) and the ordered multiple imputation methods (`polr`, `pmm`, `sample`, and `cart`) under the R package `mice` as well as `SI`, a state-of-the-art matrix completion method in the R package `softImpute`, and a baseline approach `RMean` to impute the missing ratings. See the reference manuals (Buuren & Groothuis-Oudshoorn, 2011) and (Hastie & Mazumder, 2015) of the R packages `mice` and `softImpute` for more details about the benchmark imputation algorithms. The `RMean` first scales the ratings into $[0, 1]$ by dividing the ratings of a column by the maximum rating category. Then, it simply imputes the missing entries of a certain row by the average of the non-missing entries in the row.

To compare the quality of imputation methods, we use accuracy, RMSE, MAD, percentage of upsets (%Upset), and running time as performance measures. The definitions of the measures are introduced in Section 4.4.



(a) Rating Providers

(b) Journals

Figure 2: Summary statistics of the original data

4.1 Synthetic Datasets

To systematically evaluate the performance of various imputation models, we generate test datasets from Harzing's combined quality list. We adopted two different approaches to data generation: (1) deleting entries from complete submatrices and (2) deleting entries from the original data.

4.1.1 Extraction from complete submatrices

In the first approach, we generate three submatrices of the Harzing's combined quality list that contain no missing values. We assume the ratings of each rating provider of a complete submatrix span all the rating categories of the rating provider. For example, suppose ABDC 2015 is a rating

provider in a complete submatrix. Then, since four rating categories $\{A^*, A, B, C\}$ are used in Harzing's original data, the set of ratings of ABDC 2015 in complete submatrices must be identical to $\{A^*, A, B, C\}$. The first complete data matrix is obtained by solving MILP (C.1) in Appendix C. The resulting submatrix consists of ratings of 412 journals rated by seven rating providers (EJIS 2007, EJISCI 2007, ABS 2015, ABDC 2016, CNRS 2017, DEN 2017, and HCERES 2017). We next extract another complete submatrix that contains no upsets to test how our models can perform at their best. It is obtained by solving (C.1) with additional constraints (C.2). Since the solver (CPLEX 12.7.1) failed to solve the MILP with all 14 rating providers due to a memory issue, alternatively, we extracted a zero-upset matrix from the first complete data. The resulting submatrix has 155 journals and four rating providers (EJIS 2007, EJISCI 2007, ABS 2015, and ABDC 2016). We next extract another complete submatrix that contains many upsets to see the worst-case scenarios of our models. The dataset is generated by giving different weights to each cell and by solving a weighted version of MILP (C.1). The weight for journal i and rating provider l is defined as max{1, the number of upsets caused by r_{il}}. In the weighted version, journals and rating providers with larger weight sums are more likely to be selected. The resulting submatrix has 220 journals and four rating providers (EJIS 2007, EJISCI 2007, ABDC 2016, and DEN 2017). See Table 1 for a summary profile of the three complete datasets. Note that the numbers in the last column are the proportions of upsets out of all possible combinations $(i, j, l_1, l_2) \in \mathcal{N} \times \mathcal{N} \times \mathcal{L} \times \mathcal{L}$ such that $i < j$ and $l_1 < l_2$, calculated in percentage.

$$\text{(Proportion of upsets (\%))} = \frac{\text{(\# upsets)}}{\frac{|\mathcal{N}|(|\mathcal{N}|-1)}{2}\frac{|\mathcal{L}|(|\mathcal{L}|-1)}{2}} \times 100 \tag{4.1}$$

	# journals ($\lvert\mathcal{N}\rvert$)	# rating providers ($\lvert\mathcal{L}\rvert$)	# upsets	Proportion of upsets (%)
Complete matrix A	412	7	76799	4.31 %
Complete matrix B	208	4	0	0.00 %
Complete matrix C	220	4	14522	8.66 %

Table 1: Complete Datasets

We next artificially remove randomly selected entries from the complete datasets. Four classes of the datasets are generated using different removal schemes.

- *Class 1* datasets are generated based on complete matrix A in Table 1. To imitate the characteristics of the original data (with 914 journals and 13 rating providers), we remove entries of columns using the same missing rates of the corresponding rating providers in the original dataset. Those missing rates are listed in Table 2. The overall missing rate is 22.47%. A total of 10 test datasets are generated.

	EJIS 2007	EJISCI 2007	ABS 2015	ABDC 2016	CNRS 2017	DEN 2017	HCERES 2017
Missing rates	21.99%	20.90%	20.02%	9.08%	41.68%	3.83%	40.70%

Table 2: Missing rates for Class 1 datasets

- *Class 2* datasets are generated based on complete matrix A in Table 1 using various missing rates. Each removed dataset is generated by applying a uniform missing rate across all rating providers. Ten such datasets are obtained using ten missing rates 5%, 10%, ..., 50%.

- *Class 3* datasets are generated based on complete matrix B in Table 1. The missing rates of the corresponding rating providers of the original data are used, which are 20.02%, 9.08%, 41.68%, and 40.70%. A total of 10 test datasets are generated.

- *Class 4* datasets are generated based on complete matrix C in Table 1. The missing rates of the corresponding rating providers of the original data are used, which are 21.99%, 20.90%, 9.08%, and 3.83%. A total of 10 test datasets are generated.

Table 3 shows basic characteristics of the generated synthetic datasets. For each class, 10 datasets were generated and labeled with three-digit numbers (IDs), where the first digit represents class and the following two digits identify the dataset. After the random removal process, we further reduce the size of the datasets by deleting duplicate rows, except for one representative row, as we described earlier in the construction of the reduced MILP. The last column of each class in Table 3, n_{dup} represents the number of duplicate rows in the dataset (except the representatives), which can be calculated by $n_{dup} = |\mathcal{N}| - |\bar{\mathcal{N}}|$. We observe from datasets in Class 2 that n_{dup} decreases as the missing rate increases, making the problem size larger.

Class 1					Class 2					Class 3					Class 4				
ID	$\|\mathcal{N}\|$	$\|\mathcal{L}\|$	%miss	n_{dup}	ID	$\|\mathcal{N}\|$	$\|\mathcal{L}\|$	%miss	n_{dup}	ID	$\|\mathcal{N}\|$	$\|\mathcal{L}\|$	%miss	n_{dup}	ID	$\|\mathcal{N}\|$	$\|\mathcal{L}\|$	%miss	n_{dup}
101	412	7	22.47%	61	201	412	7	5.10%	189	301	208	4	27.87%	122	401	220	4	13.86%	125
102	412	7	22.47%	70	202	412	7	9.95%	118	302	208	4	27.87%	121	402	220	4	13.86%	125
103	412	7	22.47%	79	203	412	7	15.05%	94	303	208	4	27.87%	121	403	220	4	13.86%	127
104	412	7	22.47%	71	204	412	7	19.90%	69	304	208	4	27.87%	124	404	220	4	13.86%	124
105	412	7	22.47%	77	205	412	7	25.00%	46	305	208	4	27.87%	122	405	220	4	13.86%	128
106	412	7	22.47%	76	206	412	7	30.10%	39	306	208	4	27.87%	122	406	220	4	13.86%	119
107	412	7	22.47%	65	207	412	7	34.95%	27	307	208	4	27.87%	123	407	220	4	13.86%	126
108	412	7	22.47%	70	208	412	7	40.05%	37	308	208	4	27.87%	125	408	220	4	13.86%	121
109	412	7	22.47%	82	209	412	7	44.90%	41	309	208	4	27.87%	119	409	220	4	13.86%	121
110	412	7	22.47%	66	210	412	7	50.00%	41	310	208	4	27.87%	120	410	220	4	13.86%	124

Table 3: Synthetic datasets: Class 1 has missing rates similar to the raw data, Class 2 has increasing missing rates, Class 3 contains no-upset datasets, Class 4 contains more-upset datasets

4.1.2 Extraction by deleting randomly chosen known ratings from the original data

In the second approach, we aim to generate test datasets that include all the journals and rating providers in Harzing's original combined quality list. Ten test datasets are generated by removing 10% of the randomly chosen non-missing entries, which is similar to 10-fold cross validation. Notice that each journal of a test dataset must be rated by at least one rating provider. Since there are only 13 rating providers in Harzing's data, several journals are rated by nine or fewer rating providers, making the classic 10-fold cross validation infeasible because of the assumption that each journal must be rated by at least one rating provider. Therefore, we consider a variant of 10-fold cross validation by allowing ten subsets of non-missing entries to intersect. We categorize the resulting datasets as Class 5. The overall missing rate of each of the Class 5 datasets is 45.3%, while that of the original data is 39.2%.

4.2 Benchmark Methods

Among various algorithms and models in the literature, we chose four categorical imputation algorithms, `polr`, `pmm`, `sample`, and `cart`, in the R package `mice` as benchmarks. In the pilot computational study, these four algorithms exhibited better performance than other categorical multiple

imputation algorithms in relationship to our journal rating datasets. Each of the benchmark algorithms in the `mice` package (Buuren & Groothuis-Oudshoorn, 2011) requires setting parameter m that indicates the number of imputations before pooling them into one dataset. Although it is recommended in (Buuren & Groothuis-Oudshoorn, 2011) to use $m \in \{10, \ldots, 20\}$, the current study chose $m = 5$ because of observations resulting from the computational pilot study, that indicated no significant quality improvement in the imputed ratings when larger values of the parameter ($m = 10$ or 15) are used versus the algorithms with $m = 5$.

Test datasets were also tested using `SI`, a matrix completion algorithm (Mazumder, Hastie, & Tibshirani, 2010), implemented in the R package `softImpute` (Hastie & Mazumder, 2015). Since the imputed values with `SI` are fractional, we round the imputed values to obtain integral ratings so that the experiment results are comparable with those from other methods. We denote the method by `SI` in the experiment results presented in Figures 3 through 6.

We repeat the experiments with `RMean`, another simple imputation method. For each column, `RMean` first scales each rating r in the column using the formula $\frac{r-1}{c-1}$, where c is the number of categories so that the known ratings $1, 2, 3, \ldots, c$ are transformed to $0, \frac{1}{c-1}, \frac{2}{c-1}, \ldots, 1$. Then, for each row, it replaces the missing entries of the row with the average of the non-missing entires of the same row. MS Excel is used to implement `RMean`; hence, we do not report the running time for experiments with `RMean` because its computational efficiency is obvious.

4.3 Experiment Setting

Then, **MILP** and **LP** are implemented in C# and solved by CPLEX 12.7.1. Four multiple imputation benchmark methods (`polr, pmm, sample`, and `cart`) in the R package `mice` and a matrix completion algorithm, `SI`, in the R package `softImpute` are tested in R 3.4.3 to compare the performance. For all experiments, a personal computer with 8 GB RAM and Intel Core i7 (2.40 GHz dual-core) is used.

For **MILP**, we run CPLEX for up to three hours. If CPLEX does not terminate within three hours, then we adopt the current best solution at termination as a proxy for the optimal solution. Additionally, the CPLEX optimality gap (in percentages) is reported to demonstrate solution quality with respect to optimality for the original objective function.

4.4 Performance Measures

To compare the performance of algorithms, five performance measures are considered: (1) accuracy, (2) RMSE, (3) MAD, (4) proportion of upsets (%Upsets), and (5) running time (in seconds). Let n_{miss} be the number of artificially removed entries in each synthetic dataset. In other words, for Class 1, 2, 3, and 4 datasets, n_{miss} equals the number of missing entries of each test dataset, while for Class 5 datasets, it equals the number of the 10% removed entries. Notice that the complete submatrices summarized in Table 1 have no missing entries. Recall that the rating of journal i by rating provider l in the complete dataset is denoted by $r_i^{(l)}$ and the corresponding entry in each synthetic dataset is denoted by $x_i^{(l)}$.

1. Accuracy (%): It measures how much portion of missing entries are correctly imputed. In our notations,

$$(\text{Accuracy}) = \frac{\sum_{l \in \mathcal{L}} \sum_{i \in \mathcal{N} \setminus \mathcal{N}_l} \mathbb{1}_{x_i^{(l)} = r_i^{(l)}}}{n_{miss}} \times 100,$$

where $\mathbb{1}_{x_i^{(l)}=r_i^{(l)}} = 1$ if $x_i^{(l)} = r_i^{(l)}$ and 0 otherwise.

2. RMSE: root mean square error

$$(\text{RMSE}) = \sqrt{\frac{\sum_{l\in\mathcal{L}}\sum_{i\in\mathcal{N}\setminus\mathcal{N}_l}\left(x_i^{(l)} - r_i^{(l)}\right)^2}{n_{miss}}}$$

3. MAD: mean absolute deviation

$$(\text{MAD}) = \frac{\sum_{l\in\mathcal{L}}\sum_{i\in\mathcal{N}\setminus\mathcal{N}_l}\left|x_i^{(l)} - r_i^{(l)}\right|}{n_{miss}}$$

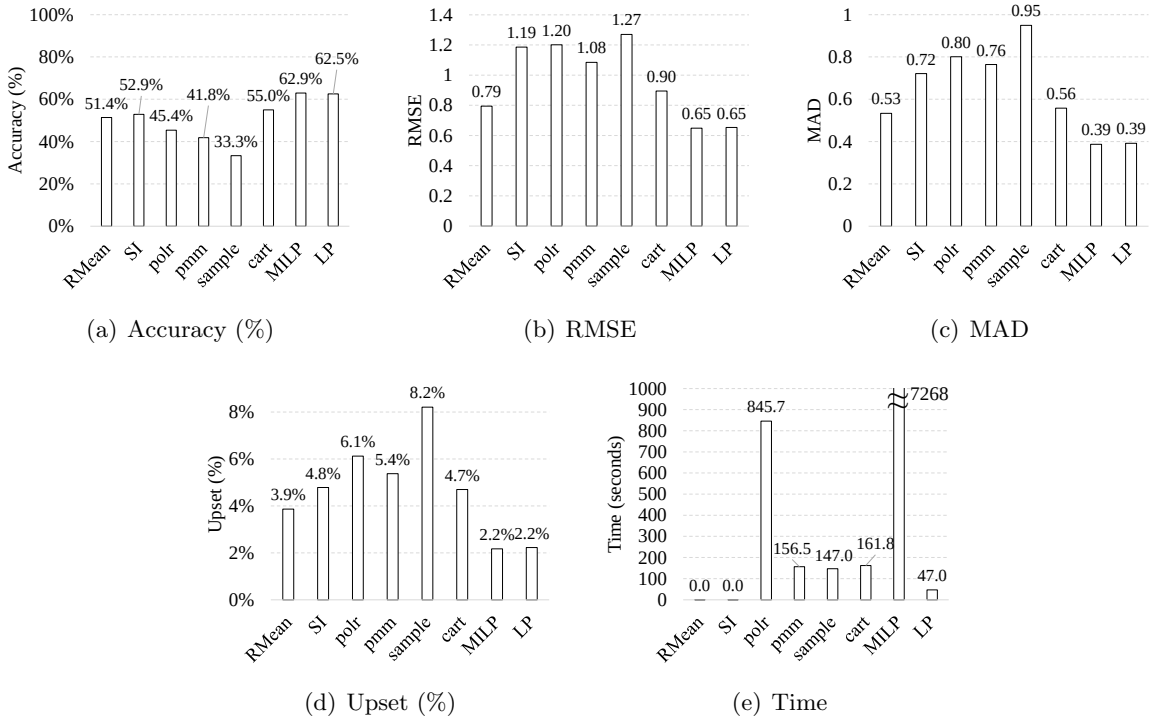
4. %Upsets: The proportion of upsets as defined in (4.1)

5. Time: running time in seconds

4.5 Experiment Results

Overall, our models **MILP** and **LP** outperform the benchmark algorithms on most datasets. The results are compared using column graphs in Figures 3, 5, and 6 for Classes 1, 3, and 4, respectively. Each figure contains five plots, comparing the methods using the five performance measures. In all plots, the average values of the ten datasets are presented, and the horizontal and vertical axes represent the methods and performance measures, respectively. For the results of Class 2 datasets presented in Figure 4, the actual values of the performance measures are presented instead of the averages. The actual values are presented because the missing rates of the ten datasets are different, and we observed some trends as the missing rates increase. The trends are captured in the series plots in Figure 4 to compare the four best-performing algorithms, `cart`, `SI`, **MILP**, and **LP**, where the horizontal axis represents the missing rates.

In Figure 3, the result for Class 1 datasets, which we constructed to see the performances in a reasonably realistic setup, are presented. Figures 3(a) through 3(c) show that **MILP** and **LP** outperform all benchmarks in accuracy, RMSE, MAD, and %Upsets. In addition, **MILP** and **LP** have at least 7.5% higher accuracy than the second-best algorithm, and this difference is significant. Between **MILP** and **LP**, there is no significant difference in the four performance measures, while Figure 3(e) shows that **MILP** is much slower than **LP**. Among the benchmark algorithms, `RMean`, `SI`, and `cart` show competitive performances; accuracy, RMSE, and MAD are reasonably good, and the running times are not the slowest. Especially, `RMean` and `SI` returned the imputation result very quickly, while the other algorithms used up to several thousand seconds of time. Because our **MILP** and **LP** directly optimize the number of upsets, the two models have the smallest %Upsets. For all other benchmark algorithms, %Upsets values are closely related to accuracy, RMSE, and MAD, which supports our objective function choice of upset minimization.

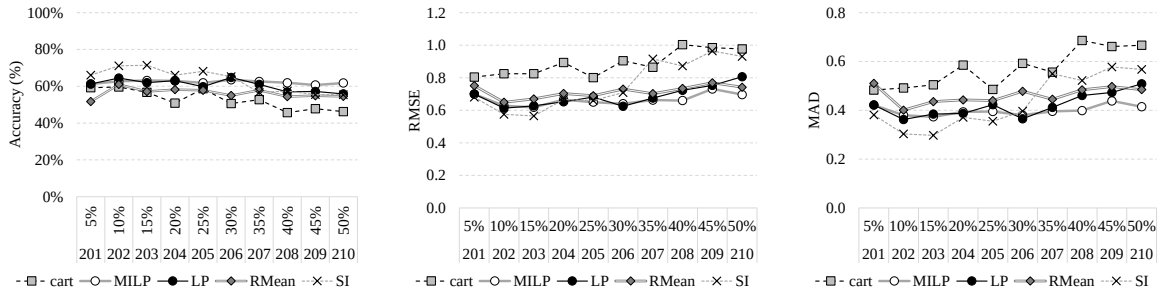


(a) Accuracy (%) (b) RMSE (c) MAD

(d) Upset (%) (e) Time

Figure 3: Average performance for Class 1 datasets

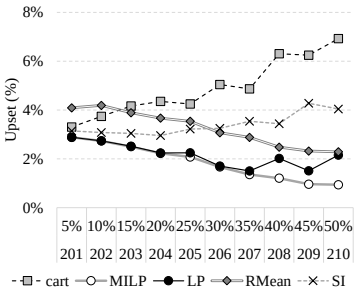
We summarize the experiment results for Class 2 datasets in Figure 4 to present some trends as missing rates increase. Our models outperform the benchmark methods when the missing rates are 30% or more, while `SI` performs better than the current models when the missing rates are 25% or less. In terms of running times, **MILP** used a significant amount of time when the missing rates were large, while **LP** solves them in a reasonable time period. For datasets with higher missing rates, **MILP** does not terminate with optimality. However, the returned solutions with **MILP** outperform **LP** in terms of imputation quality (accuracy, RMSE, MAD, and %Upsets). We further observe that our models show relatively stable performance across various missing rates.



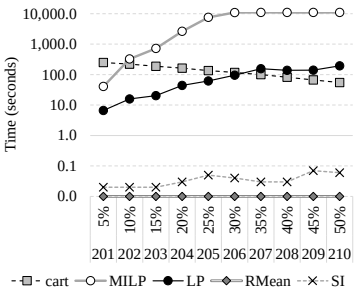
(a) Accuracy (%)

(b) RMSE

(c) MAD



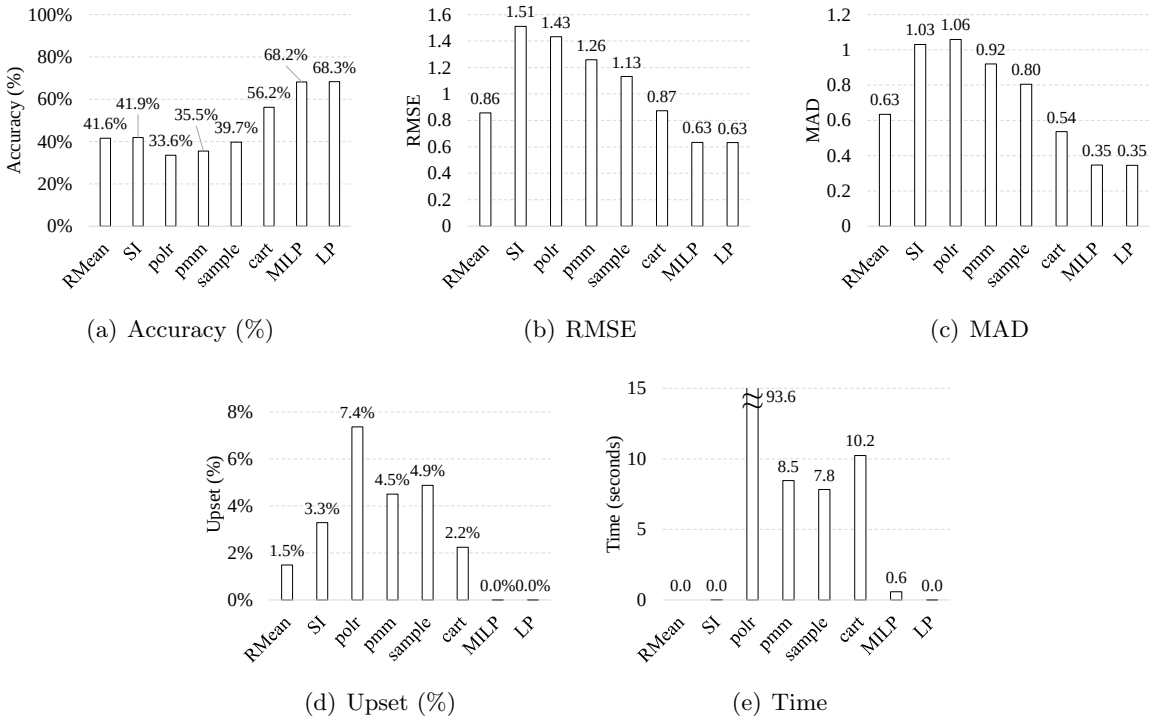
(d) Upset (%)



(e) Time

Figure 4: Performance of three algorithms for Class 2 datasets

In Figure 5, the results for Class 3 datasets, which we constructed to check the best performances of our models, are presented. For all performance measures except for time, **MILP** and **LP** significantly outperform other benchmark methods. The **MILP** and **LP** approaches have at least a 12.0% higher accuracy than the second-best algorithm, which is a significant difference. Between **MILP** and **LP**, there are no significant differences across all five performance measures. Since Class 3 datasets are small-sized (208 journals with four rating providers), both **MILP** and **LP** return the results within a second. Among the benchmark algorithms, `cart` shows competitive performance in terms of accuracy, RMSE, and MAD, and the running times are not the slowest. Unlike the results for Classes 1 and 2, `RMean` has a lower accuracy than the competitors.



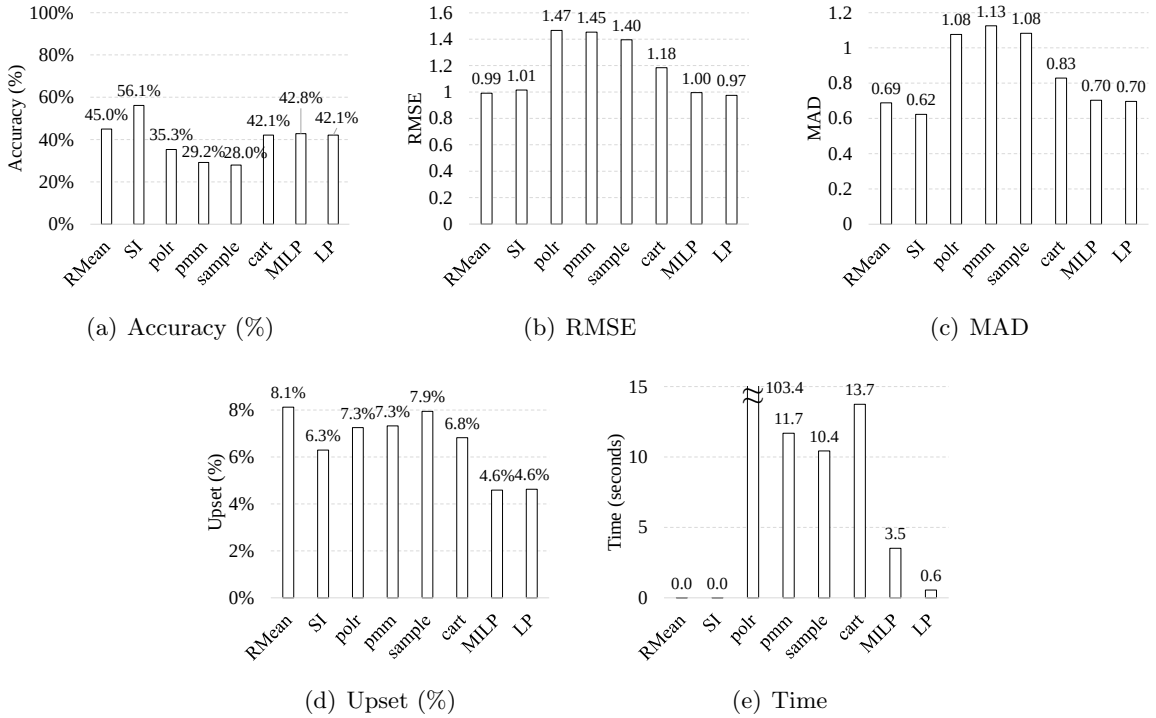
(a) Accuracy (%) (b) RMSE (c) MAD

(d) Upset (%) (e) Time

Figure 5: Average performance for Class 3 datasets

Figure 6 shows the results for Class 4 datasets, which have many upsets. As expected, the predictive qualities of **MILP** and **LP** are not as good as those for other test datasets, although both models still provide the best %Upset result. Note that complete matrix C, which was used to generate Class 4 datasets, has 8.66% upset rates, and our models provide solutions with a lower %Upset. This indicates that the current model may not be the best when the original data have many upsets. For this class of datasets, `SI` outperforms the others in terms of accuracy and MAD. For RMSE, the `SI`, `RMean`, **MILP**, and **LP** algorithms show similar performances. Overall, **MILP** and **LP** are in the second-best group with `RMean` and perform better than the `mice` methods. On the other hand, `SI` has higher absolute residuals for inaccurately imputed entries than our models. Thus, when the volatility of inaccuracies is one of the key decision criteria, decision-makers may prefer the current models over `SI`.

For Class 5 datasets, which we obtained by deleting the existing ratings in the original data, CPLEX fails to solve both **MILP** and **LP** within the three-hour time limit. Recall that each dataset includes all 914 journals and 13 rating providers with 5,382 missing entries (missing rate: 45.3%). To resolve this issue, rather than imputing the entire dataset, we (1) partition the set of journals into two subsets, (2) impute the two subdata using **LP**, and (3) combine the quality list. This setting allows **LP** to find an optimal solution within an hour. Since we observed that `SI` and `RMean` outperform all other benchmark algorithms from the previous computational results, we present computational results for only **LP** and `SI`. For a fair comparison, we applied `SI` for original test datasets and partitioned datasets. Since the imputation by `RMean` is not affected by partitioning, we applied `RMean` only to the original test datasets. See Table 4 for the summary results. First, we observe that when we use `SI`, there is no considerable difference between imputation on the original data and that on the partitioned data. In addition, **LP** shows significantly better performance in terms of accuracy, RMSE, and MAD, while it is still computationally expensive.



(a) Accuracy (%) (b) RMSE (c) MAD

(d) Upset (%) (e) Time

Figure 6: Average performance for Class 4 datasets

	`SI`	`SI` (partitioned)	`RMean`	**LP** (partitioned)
Accuracy (%)	54.9%	54.7%	53.31%	**63.3%**
RMSE	0.87	0.88	0.8060	**0.71**
MAD	0.44	0.45	0.5205	**0.41**
Upset (%)	5.5%	6.2%	3.90%	**2.0%**
Time (sec)	**0.06**	**0.05**	NA	3022.89

Table 4: Summary results on Class 5 datasets

We next summarize the results of the computational experiments and offer some thoughts and observations regarding the practical utility of the research findings.

- In Class 1 experiments, **MILP** and/or **LP** significantly outperform the benchmark methods in terms of accuracy, RMSE, MAD, and %Upsets. Class 1 datasets are constructed using the same missing rates of the rating providers as those in Harzing's original combined list. Our models may be preferable in this more realistic scenario.
- The **MILP** algorithm encountered computational challenges in Class 5 experiments where the test datasets include all 914 journals and 13 rating providers, and the missing rate is 45.3%. However, **LP** was solvable after partitioning the data, and it performed significantly better than `SI`, which is the best-performing benchmark method.
- When the data, by its nature, has a small number of upsets, our models perform exceptionally better than benchmark methods as we observed in Class 3 experiments.
- In Class 2 experiments, **MILP** and **LP** showed particularly good performance unless the missing rates are 25% or less, while `SI` performed best when missing rates are low.

- Even in the worst-case scenario highlighted in Class 4 experiments, **MILP** and **LP** show competitive results in terms of RMSE, MAD, and %Upset. Our models were better than all `mice` methods. In these experiments, `SI` performed best.

- No significant differences in performance were observed between **MILP** and **LP** except for running times. When decision-makers are presented with a large problem size, **MILP** is computationally expensive, thereby rendering more promise to **LP** in practice.

5 Conclusions

Estimating missing journal ratings in a quality list is an important issue in evaluating scholarship of faculty in tenure and promotion procedures in most academic institutions. We proposed a mixed integer programming model to impute missing ratings in the aggregated data of multiple quality lists. This was done without directly using the explanatory factors of each quality list nor making randomness assumptions that hinders the use of well-known multiple imputation algorithms. Computational results show that our models outperform the existing multiple imputation algorithms in terms of prediction accuracy, RMSE, MAD, and elapsed time. We suggest that the linear program constructed by relaxing the integrality constraints in the original mixed integer program is an efficient and reasonably effective method to determine the unknown ratings for problems that **MILP** cannot handle.

Finally, we leave further comments regarding the practical usage of our models.

- Our models can be applicable to data with similar structure with journal rating data, in which an upset can be regarded as an "unusual" conflict. In other words, the features (or rating providers) of the data must be coherent. In other words, our models are not suitable for rating data in which rating providers have considerably different rating criteria. For this reason, it is important to carefully select rating providers to take advantage of our models. Therefore, we suggest researchers or practitioners use discretion in checking whether the target data have the inherent property. Calculating %Upsets with non-missing quadruples (i, j, l_1, l_2) may be an alternative. For Harzing's original combined list, 3.27% of all non-missing quadruples are upsets.

- Practitioners may want to use a certain subset of journals in the analysis. For example, we extracted a complete submatrix within the operations research/management sciences subject area and observed that the proportion of upsets is only 1.78%, which is smaller than the proportion of 2.52% for complete matrix A. Similarly, selecting a subset of the rating providers may help improve the quality of our models.

- The **MILP** algorithm is a good alternative for imputation of journal rating data provided that it is solvable and the missing rates are not low. With rating data with a small missing rate, `SI` may be preferable. When **MILP** is too expensive to solve (when either the problem size or missing rate is large), **LP** is a strong alternative because it is much faster than **MILP** and still performs very well.

References

Allison, P. D. (2000). Multiple imputation for missing data: A cautionary tale. *Sociological Methods & Research*, *28*(3), 301–309.

Allison, P. D. (2017). The peculiarities of missing at random. Accessed February, 2018, available at `https://statisticalhorizons.com/missing-at-random`.

Bennett, J., Lanning, S., & etc. (2007). The netflix prize. In *Proceedings of KDD Cup and Workshop* (Vol. 2007, p. 35). Association for Computing Machinery.

Bertsimas, D., Pawlowski, C., & Zhuo, Y. D. (2018). From predictive methods to missing data imputation: An optimization approach. *Journal of Machine Learning Research*, *18*(196), 1–39.

Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in R. *Journal of Statistical Software*, *45*(3).

Candès, E. J., & Recht, B. (2009). Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, *9*(6), 717.

Candès, E. J., & Tao, T. (2010). The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, *56*(5), 2053–2080.

Cheang, B., Chu, S. K. W., Li, C., & Lim, A. (2014). OR/MS journals evaluation based on a refined PageRank method: An updated and more comprehensive review. *Scientometrics*, *100*(2), 339–361.

Dwork, C., Kumar, R., Naor, M., & Sivakumar, D. (2001). Rank aggregation methods for the web. In *Proceedings of the 10th International Conference on World Wide Web* (pp. 613–622). New York, NY, USA: Association for Computing Machinery.

Harzing, A.-W. (2017). Journal quality list. Accessed September 14, 2017, available at `https://www.harzing.com`.

Hastie, T., & Mazumder, R. (2015). softimpute: Matrix completion via iterative soft-thresholded svd (r package version 1.4). `https://CRAN.R-project.org/package=softImpute`.

Hastie, T., Mazumder, R., Lee, J. D., & Zadeh, R. (2015). Matrix completion and low-rank svd via fast alternating least squares. *The Journal of Machine Learning Research*, *16*(1), 3367–3402.

Hult, G. T. M., Neese, W. T., & Bashaw, R. E. (1997). Faculty perceptions of marketing journals. *Journal of Marketing Education*, *19*(1), 37-52.

Jain, P., Netrapalli, P., & Sanghavi, S. (2013). Low-rank matrix completion using alternating minimization. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing* (pp. 665–674). Palo Alto, CA: Association for Computing Machinery.

Kenyon-Mathieu, C., & Schudy, W. (2007). How to rank with few errors. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing* (pp. 95–103). Association for Computing Machinery.

Kou, Z., Chang, Y., Zheng, Z., & Zha, H. (2010). Learning to blend rankings: A monotonic transformation to blend rankings from heterogeneous domains. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management* (pp. 1921–1924). Toronto, ON, Canada: Association for Computing Machinery.

Mazumder, R., Hastie, T., & Tibshirani, R. (2010). Spectral regularization algorithms for learning large incomplete matrices. *Journal of Machine Learning Research*, *11*(Aug), 2287–2322.

Mingers, J., & Harzing, A.-W. (2007). Ranking journals in business and management: A statistical analysis of the harzing data set. *European Journal of Information Systems*, *16*(4), 303–316.

Moussa, S., & Touzani, M. (2010). Ranking marketing journals using the google scholar-based hg-index. *Journal of Informetrics*, *4*(1), 107 - 117.

Olson, J. E. (2005). Top-25-business-school professors rate journals in operations management and related fields. *Interfaces*, *35*(4), 323-338.

Pedings, K. E., Langville, A. N., & Yamamoto, Y. (2012, Jun 01). A minimum violations ranking method. *Optimization and Engineering*, *13*(2), 349–370.

Raisali, F., Hassanzadeh, F. F., & Milenkovic, O. (2013, July). Weighted rank aggregation via relaxed integer programming. In *2013 IEEE International Symposium on Information Theory* (p. 2765-2769). Istanbul, Turkey: IEEE.

Recht, B. (2011). A simpler approach to matrix completion. *Journal of Machine Learning Research*, *12*(Dec), 3413–3430.

Rubin, D. B. (2004). *Multiple imputation for nonresponse in surveys* (Vol. 81). John Wiley & Sons.

Schafer, J. L. (1999). Multiple imputation: A primer. *Statistical Methods in Medical Research*, *8*(1), 3–15.

Sterne, J. A., White, I. R., Carlin, J. B., Spratt, M., Royston, P., Kenward, M. G., & etc. (2009). Multiple imputation for missing data in epidemiological and clinical research: Potential and pitfalls. *BMJ*, *338*, b2393.

Tse, A. (2001). Using mathematical programming to solve large ranking problems. *Journal of the Operational Research Society*, 1144–1150.

Van Buuren, S. (2012). *Flexible imputation of missing data.* New York, NY: CRC press.

A Appendix: Proofs of the theorems

Theorem 3.1. *Let (x^*, z^*) be an optimal solution to* (3.3)*. Then,*

$$(z^*)_{ij}^{(l_1,l_2)} = \begin{cases} 1 & \text{if } (i,j,l_1,l_2) \text{ is an upset in the imputed data } x^* \\ 0 & \text{Otherwise} \end{cases}$$

Proof. Suppose (i,j,l_1,l_2) is an upset in the imputed data x^*, where $i<j$ and $l_1<l_2$. Without loss of generality, we assume that $(x^*)_i^{(l_1)} - (x^*)_j^{(l_1)} < 0$ and $(x^*)_i^{(l_2)} - (x^*)_j^{(l_2)} > 0$. By constraints (3.3c) and (3.3h), $(z^*)_{ij}^{(l_1)} = (z^*)_{ji}^{(l_2)} = 1$. By constraint (3.3a), $(z^*)_{ij}^{(l_1,l_2)} = 1$. We next consider the case, where (i,j,l_1,l_2) is not an upset in the imputed data x^*. Suppose $(x^*)_i^{(l_1)} - (x^*)_j^{(l_1)} < 0$ and $(x^*)_i^{(l_2)} - (x^*)_j^{(l_2)} < 0$. We claim that $(z^*)_{ji}^{(l_1)} = (z^*)_{ji}^{(l_2)} = 0$. To prove by contradiction, without loss of generality, assume that $(z^*)_{ji}^{(l_1)} = 1$. It can be implied by (3.3a) that $(z^*)_{ij}^{(l_1,l_2)} = 1$. On the other hand, consider a feasible solution (x^*, z^{**}), where $z^{**} = z^*$ except that $(z^{**})_{ji}^{(l_1)} = 0$ and $(z^{**})_{ij}^{(l_1,l_2)} = 0$. It can be easily shown that (x^*, z^{**}) is feasible; hence, the objective function value at (x^*, z^{**}) improves the optimal value, making a contradiction. Therefore, $(z^*)_{ji}^{(l_1)} = 0$. Proof for the case that $(x^*)_i^{(l_1)} - (x^*)_j^{(l_1)} > 0$ and $(x^*)_i^{(l_2)} - (x^*)_j^{(l_2)} > 0$ can be done similarly. Finally, consider the case that $(x^*)_i^{(l_1)} - (x^*)_j^{(l_1)} = 0$ or $(x^*)_i^{(l_2)} - (x^*)_j^{(l_2)} = 0$. Without loss of generality, assume that $(x^*)_i^{(l_1)} - (x^*)_j^{(l_1)} = 0$. This implies that $(z^*)_{ij}^{(l_1)} = (z^*)_{ji}^{(l_1)} = 0$ because $z_{ij}^{l_1}$ and $z_{ji}^{l_2}$ are unconstrained, while (3.3a) and (3.3b) force those values as small as possible to minimize the objective function value, setting them zero at optimality. It can be shown with arguments used in previous cases that $(z^*)_{ij}^{(l_1,l_2)} = 0$, concluding the result. □

Theorem 3.2. *There exists an optimal solution to* **LP** *whose $x_i^{(l)}, l \in \mathcal{L}, i \in \mathcal{N}$ components are integral.*

Proof. We first show the existence of an optimal solution (x^*, z^*) such that

$$(z^*)_{ij}^{(l)} = \max\left\{\frac{1}{c_l - 1}((x^*)_j^{(l)} - (x^*)_i^{(l)}), 0\right\} \tag{A.1}$$

for all $l \in \mathcal{L}$ and $i \neq j \in \bar{\mathcal{N}}$. Let (x', z') be an optimal solution to **LP**. By feasibility of (x', z'),

$$(z')_{ij}^{(l)} \geq \max\left\{\frac{1}{c_l - 1}((x')_j^{(l)} - (x')_i^{(l)}), 0\right\}$$

for all $l \in \mathcal{L}$ and $i \neq j \in \bar{\mathcal{N}}$. To prove by contradiction, suppose there exist $l_0 \in \mathcal{L}$ and $i_0, j_0 \in \bar{\mathcal{N}}$ such that $(z')_{i_0 j_0}^{(l_0)} > \max\left\{\frac{1}{c_{l_0}-1}((x')_{j_0}^{(l_0)} - (x')_{i_0}^{(l_0)}), 0\right\}$. Let us consider a vector by decreasing the value of $(z')_{i_0 j_0}^{(l_0)}$ until the constraint is tight. That is, we construct a new vector (x^*, z^*) such that $(x^*, z^*) = (x', z')$ except that

$$(z^*)_{i_0 j_0}^{(l_0)} = \max\left\{\frac{1}{c_{l_0} - 1}((x^*)_{j_0}^{(l_0)} - (x^*)_{i_0}^{(l_0)}), 0\right\}.$$

Observe that the change does not affect the feasibility because it decreases the right-hand-sides of the first two constraints of **LP**. Furthermore, the objective value does not change, proving that

(x^*, z^*) is an optimal solution. By repeating the same arguments for all $l \in \mathcal{L}$ and $i \neq j \in \bar{\mathcal{N}}$ with $(z')_{ij}^{(l)} > \frac{1}{c_l - 1}((x')_j^{(l)} - (x')_i^{(l)})$, we can obtain a desired alternate optimal solution, which satisfies (A.1). Therefore, we can assume that $z_{ij}^{(l)}$ is a convex piecewise linear function over $[0, c_l]^{n_l}$ for all $l \in L$ and $i \neq j \in N$.

Next, note that, by the first two constraints of **LP**, any optimal solution (x^*, z^*) satisfies

$$(z^*)_{ij}^{(l_1,l_2)} = \max\left\{(z^*)_{ij}^{(l_1)} + (z^*)_{ji}^{(l_2)} - 1, (z^*)_{ij}^{(l_2)} + (z^*)_{ji}^{(l_1)} - 1, 0\right\}$$

for all $l_1 < l_2 \in \mathcal{L}$ and $i < j \in \bar{\mathcal{N}}$. Therefore, we can reformulate **LP** by imposing the conditions

$$\begin{aligned} z_{ij}^{(l_1,l_2)} &= \max\left\{z_{ij}^{(l_1)} + z_{ji}^{(l_2)} - 1, z_{ij}^{(l_2)} + z_{ji}^{(l_1)} - 1, 0\right\}, && i < j \in \mathcal{N}, l_1 < l_2 \in \mathcal{L} \\ z_{ij}^{(l)} &= \max\left\{\tfrac{1}{c_l - 1}(x_j^{(l)} - x_i^{(l)}), 0\right\}, && i \neq j \in \mathcal{N}, l \in \mathcal{L} \end{aligned}$$

Since weighted sums and the maximum of convex piecewise linear functions is a convex piecewise linear function, the objective function can be rewritten as a convex piecewise linear function in x variables. Therefore, there exists an optimal solution to **LP** whose x components form an extreme point of the epigraph of the objective function. Notice that the base piecewise function $z_{ij}^{(l)} = \max\left\{\frac{1}{c_l - 1}(x_j^{(l)} - x_i^{(l)}), 0\right\}$ subdivides, where $x_j^{(l)} = x_i^{(l)}$; hence, each portion of the entire piecewise linear function is subdivided by equations of the form $x_i^{(l)} = x_j^{(l)}$. Therefore, the x component of each extreme point is a solution to the system of equations, where each equation is either $x_i^{(l)} = r_i^{(l)}$ for some $i \in \bar{\mathcal{N}}_l$ or $x_i^{(l)} = x_j^{(l)}$ for some $i, j \notin \bar{\mathcal{N}}_l$. Since the system of equations has the unique solution and $r_i^{(l)}, i \in \bar{N}_l$ are integers, all x components of an extreme point are integral, showing the existence of an integral optimal solution to **LP**. □

B Appendix: Formulations

In this section, the full formulations of **MILP** and **LP**, which were discussed in Section 3, are presented. The MILP formulation presented in this section is a weighted version of the MILP formulation (3.3), where weights are defined based on the number of duplicate records. For detailed derivations and explanations of the constraints, check (3.3) in Section 3. Note that the following models are the final models that are used in the computational experiment in Section 4.

$$
\begin{aligned}
\textbf{MILP} \quad \min \quad & \sum_{i<j\in\bar{\mathcal{N}}_l} \sum_{l_1<l_2\in\mathcal{L}} u_i u_j z_{ij}^{(l_1,l_2)} \\
\text{s.t.} \quad & z_{ij}^{(l_1,l_2)} \geq z_{ij}^{(l_1)} + z_{ji}^{(l_2)} - 1, && i<j\in\bar{\mathcal{N}},\ l_1<l_2\in\mathcal{L}, \\
& z_{ij}^{(l_1,l_2)} \geq z_{ij}^{(l_2)} + z_{ji}^{(l_1)} - 1, && i<j\in\bar{\mathcal{N}},\ l_1<l_2\in\mathcal{L}, \\
& z_{ij}^{(l)} \geq \frac{1}{c_l-1}(x_j^{(l)} - x_i^{(l)}), && i\neq j\in\bar{\mathcal{N}},\ l\in\mathcal{L}, \\
& x_i^{(l)} = r_i^{(l)}, && i\in\bar{\mathcal{N}}_l,\ l\in\mathcal{L}, \\
& z_{ij}^{(l)} = 1, z_{ji}^{(l)} = 0, && (i,j)\in\bar{\mathcal{N}}_l^{+}, \\
& z_{ij}^{(l)} = z_{ji}^{(l)} = 0, && (i,j)\in\bar{\mathcal{N}}_l^{0}, \\
& x_i^{(l)} \in \{1,\ldots,c_l\}, && i\in\bar{\mathcal{N}}, l\in\mathcal{L}, \\
& z_{ij}^{(l)} \in \{0,1\}, && i,j\in\bar{\mathcal{N}},\ l\in\mathcal{L}, \\
& 0\leq z_{ij}^{(l_1,l_2)} \leq 1, && i<j\in\bar{\mathcal{N}},\ l_1<l_2\in\mathcal{L}
\end{aligned}
\tag{B.1}
$$

$$
\begin{aligned}
\textbf{LP} \quad \min \quad & \sum_{i<j\in\bar{\mathcal{N}}_l} \sum_{l_1<l_2\in\mathcal{L}} (u_i u_j) z_{ij}^{(l_1,l_2)} \\
\text{s.t.} \quad & z_{ij}^{(l_1,l_2)} \geq z_{ij}^{(l_1)} + z_{ji}^{(l_2)} - 1, && i<j\in\bar{\mathcal{N}},\ l_1<l_2\in\mathcal{L}, \\
& z_{ij}^{(l_1,l_2)} \geq z_{ij}^{(l_2)} + z_{ji}^{(l_1)} - 1, && i<j\in\bar{\mathcal{N}},\ l_1<l_2\in\mathcal{L}, \\
& z_{ij}^{(l)} \geq \frac{1}{c_l-1}(x_j^{(l)} - x_i^{(l)}), && i\neq j\in\bar{\mathcal{N}},\ l\in\mathcal{L}, \\
& x_i^{(l)} = r_i^{(l)}, && i\in\bar{\mathcal{N}}_l,\ l\in\mathcal{L}, \\
& z_{ij}^{(l)} = 1, z_{ji}^{(l)} = 0, && (i,j)\in\bar{\mathcal{N}}_l^{+}, \\
& z_{ij}^{(l)} = z_{ji}^{(l)} = 0, && (i,j)\in\bar{\mathcal{N}}_l^{0}, \\
& 1\leq x_i^{(l)} \leq c_l, && i\in\bar{\mathcal{N}}, l\in\mathcal{L}, \\
& 0\leq z_{ij}^{(l)} \leq 1, && i,j\in\bar{\mathcal{N}},\ l\in\mathcal{L}, \\
& 0\leq z_{ij}^{(l_1,l_2)} \leq 1, && i<j\in\bar{\mathcal{N}},\ l_1<l_2\in\mathcal{L}
\end{aligned}
\tag{B.2}
$$

C Appendix: Selecting Complete subdata

To verify the performance or the prediction power of our model in our experiment setting, we must have a dataset without missing values. From this new complete dataset, we randomly delete some elements of the matrix, run the imputation models and algorithms, and compare the predicted and

actual rating values. From the original data with missing values, we construct a submatrix without missing values by selecting subsets of rows and columns.

First, we define binary constants that indicate whether a cell has a rating.

$$w_{il} = \begin{cases} 1 & \text{if } l \in \mathcal{L} \text{ and } i \in \mathcal{N}_l \\ 0 & \text{otherwise} \end{cases}$$

We next define the decision variables $x_i, i \in \mathcal{N}$ and $y_l, l \in \mathcal{L}$ to model

$$x_i = \begin{cases} 1 & \text{if } i \in \mathcal{N} \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

$$y_l = \begin{cases} 1 & \text{if } l \in \mathcal{L} \text{ is selected} \\ 0 & \text{otherwise} \end{cases}.$$

The desired optimization model returns subdata whose rows and columns are corresponding to index sets $\{i \mid x_i = 1\}$ and $\{l \mid y_l = 1\}$, respectively. Any feasible choice of $\{i \mid x_i = 1\}$ and $\{l \mid y_l = 1\}$ must satisfy the condition that $w_{il} = 1$ if $x_i = 1$ and $y_l = 1$. This relationship is easily modeled by the inequality $x_i y_l \leq w_{il}$. To linearize the product term, we introduce auxiliary variables z_{il} to linearize the product term $x_i y_l$ and add additional constraints $z_{il} \leq x_i, z_{il} \leq y_l$, and $z_{il} \geq x_i + y_l - 1$.

Various objective functions can be considered to generate complete subdata, such as the number of rating providers, number of journals, weighted sum of the two, and number of total number of entries. Among several alternatives, we decide to maximize the number of selected cells in the submatrix, and the following MILP model is obtained.

$$\begin{aligned}
\max \quad & \sum_{i \in \mathcal{N}} \sum_{l \in \mathcal{L}} z_{il} \\
\text{s.t.} \quad & z_{il} \leq x_i, && i \in \mathcal{N},\ l \in \mathcal{L} && \text{(C.1a)} \\
& z_{il} \leq y_l, && i \in \mathcal{N},\ l \in \mathcal{L} && \text{(C.1b)} \\
& z_{il} \geq x_i + y_l - 1, && i \in \mathcal{N},\ l \in \mathcal{L} && \text{(C.1c)} \\
& z_{il} \leq w_{il}, && i \in \mathcal{N},\ l \in \mathcal{L} && \text{(C.1d)} \\
& x_i, y_l, z_{il} \in \{0, 1\}, && i \in \mathcal{N},\ l \in \mathcal{L} && \text{(C.1e)}
\end{aligned}$$

Constraints (C.1a)–(C.1d) define z_{il}. Constraints (C.1a)–(C.1c) are well-known constraints that linearize the product term $z_{il} = x_i y_l$. Observe that z_{il} is forced to be zero if any of x_i or y_l is zero by constraints (C.1a), (C.1b), and (C.1d) and z_{il} is forced to be one if both x_i and y_l are equal to one by constraints (C.1c) and (C.1d). Constraint (C.1d) forces z_{il} to be zero if the corresponding cell is missing in the original dataset. Finally, all variables are required to be integers by Constraint (C.1e).

We also create complete subdata that do not include upsets in the ratings. The subdata will have different characteristics from the previous complete subdata, as no upsets exist in the data matrix. An MILP model can be formulated by adding an additional constraint to prevent upsets. If $r_i^{(l)} \leq r_j^{(l)}$ and $r_i^{(k)} > r_j^{(k)}$, then two journals i and j have an upset relationship by rating providers l and k. To prevent having upsets in the submatrix, we can add constraint

$$x_i + x_j + y_l + y_k \leq 3, \qquad \text{if } r_i^{(l)} \leq r_j^{(l)} \text{ and } r_i^{(k)} > r_j^{(k)} \text{ for } i, j \in \mathcal{N}, k, l \in \mathcal{L} \qquad \text{(C.2)}$$

to the above model (C.1).