CSCM35: Big Data and Data Mining Coursework 1

Andy Gray
445348

09/04/20

1 Introduction

We have a practical task assigned to us, that is related to the field of data mining. This task aims to use the association rule, a rule-based machine learning technique, to discover interesting relationships within the provided large dataset. We will be creating to code process the data, as well as analysing the results to see if any insights can be gains and possible reasons to why these might be the case. Throughout this paper we will be explaining our understanding of the problems related to mining data for common patterns while performing visualisations and statistical analysis, as well as applying the association rule mining algorithm on the data, to be able to analyse the outcomes.

Data mining is a necessary part of obtaining knowledge through discovery in databases (KDD). KDD is the term used for the overall process, that involves turning the raw data into useful information. Data mining tasks split into two main categories. These are predictive and descriptive tasks. However, these two main categories split further into four core mining task. These four tasks are cluster analysis, predictive modelling, anomaly detection and association analysis [12]. We will be focusing on the association analysis within this paper.

[Needs expanding about overview of results]

We will first look at the algorithms used within the methodology, explaining how they work and what is the maths formulas behind them driving the algorithm. We will then explain the dataset, and the data preprocessing that occurred, followed by an explanation of the packages used and the parameters set for the algorithms. We will then explain the results and then discuss them and what insights we might have gained. To end, we will be then concluding what we have found.

2 Methodology

2.1 Algorithms Used Explanation

The first algorithm that we used is one that is from the frequent itemset mining methods, called Apriori [6]. Apriori is an unsupervised learning machine learning algorithm proposed by R. Agrawal and R. Srikant in 1994 [2, 5]. The algorithm focuses on using boolean association rules [2] from using prior knowledge of itemsets that contain the frequent properties. Apriori uses a level-wise search, which operates an iterative approach, where k-itemsets get used for exploring $(k+1)$-itemsets [8]. In order to improve efficiency, which will reduce the search space, an important characteristic called the Apriori property needs to be applied [6].

The Apriori property has a two-step process which involves the join and prunes step. For this explanation, F_k represents the k-itemset where L_k represents the candidate for the k-itemset. The process of joining is to generate a new itemset, L_{k+1}, from the F_K itemset. While the pruning stage aims to identify the itemsets in L_{k+1} that are infrequent from k, and then remove them [8]. What indicates if the item is infrequent depends on the support count, which is predefined beforehand. Therefore what the algorithm does, is: Let us assume that $k = 1$ and a support count of 2, we generate a frequent itemset, at first 1, which we will refer to as F_1. What this is doing is scanning the dataset to figure out the count of each occurrence of each item. The next step is the merge, or join, the datasets. Using F_k we can then create L_{k+1}. We then prune the data based on the support count eliminating any data that is infrequent, therefore leaving any data that is classed as frequent, adding it to F_{k+1}. This process is repeated until F_k is empty [8, 6].

The second algorithm that we have used is called the association rule. Rakesh Agrawal, Tomasz Imieliński and Arun Swami developed the algorithm in 1993 [1]. The association rule algorithm is an unsupervised machine learning algorithm [5]. What this algorithm focuses around is the support of the datasets' items and the confidence of the association. The math formula for the support is $\mathit{support}(A \Rightarrow B) = P(A \cup B)$, and the math formula for the confidence is $confidence(A \Rightarrow B) = P(B|A)$. Similar to the apriori, the support count will drop any relationships that do not meet the desired count. The formula to figure out if the relationships meet the support count is $confidence(A \Rightarrow B) = P(B|A) = \frac{\mathit{support}(A \cup B)}{\mathit{support}(A)} = \frac{support_count(A \cup B)}{support_count(A)}$ [8, 6]. However, the association rule relies on a procedure, like the apriori algorithm, to have been implemented on the dataset first before it can work effectively. While the association rule requires the support threshold, the confidence level, which we can use to make decisions based on the links, can be changed to additional metrics. The metric can be several different ones like conviction and leverage, the one that we will focus on is lift. The metric lift was introduced in 1997 by Sergey Brin, Rajeev Motwani, Jefferey D. Ullman and Shalom Tsur [3]. This metric figures out how the antecedent and consequent of a rule, A -¿ C, would occur together and not as statistically independent items. The lift score would indicate if A and C are independent by having a score of exactly 1. The math formula for lift is constructed as $lift(A \rightarrow C) = \frac{confidence(A \rightarrow C)}{\mathit{support}(C)}, range[0, \infty]$ [3, 8].

Overall the apriori algorithm will reduce the dataset by pruning it. The amount of pruning depends on the support count threshold that is applied. This will create the required frequent itemset which the association rule requires. The association rule will then go through the frequent itemset to acquire any patterns of items based on the support count and the metric. In our case, this is the lift or confidence metric.

2.2 Dataset and Data Preprocessing

The dataset that has we have acquired is a shopping dataset. It is 44MB in size and is in the format of CSV. There are eight attributes, within the dataset, with 541,910 records. The attributes are InvoiceNo, StockCode, Description, Quantity, InvoiceDate, UnitPrice, CustomerId, Country. There are 4,335 unique customers, 1,8405 individual invoices, 3,659 unique stock items and 37 unique countries.

The purpose of data preprocessing is to convert any raw data into a format that is appropriate for the following analysis of the data. Preprocessing can involve fusing data from several sources, as well as cleaning the raw data to remove any noise, duplicate observations or ambiguity [12]. The main aim of the preprocessing is to get data that is accurate, complete and consistent, but in the real world, we will usually get inaccurate, incomplete and inconsistent data [6]. The preprocessing stage can also involve just selecting the essential records and features that are desired and are relevant to the set data mining task [12]. We can now see that the main aim of data processing is to clean the data, we achieve this through filling in missing values, identifying or removing outliers, smoothing noisy data, and resolving and data inconsistencies [6].

The dataset had values missing in a number of the columns. The rows that had any missing values, within the features, were removed from the dataset. Also, any rows that had data that was an outlier, within its features, was removed from the dataset. These outliers included minus values. Once we had carried out these data cleaning actions, we then have 396,371 records remaining. The cleaning process indicates that we had removed a total of 145,539 records from the dataset.

Data transformation. -¿ takes raw values and puts it into boolean values it needs for the apriori.

$support(A \Rightarrow B) = P(A \cup B)$

2.3 Packages Used

We will be using the programming language Python 3 [10], as this allows us to use all the required algorithms needed to analyse the dataset, to check for any trends. We will be using the library package MLXtend[11] to be able to get access to the apriori and the association rule algorithm. We will be using Matplotlib's [7] package library for visualising our data, to allow us to be able to get insights and spot possible trends.

2.4 Parameters

3 Results

4 Discussion

5 Conclusion

References

[1] Agrawal, R., Imieliński, T., and Swami, A. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data* (1993), pp. 207–216.

[2] Agrawal, R., Srikant, R., et al. Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB* (1994), vol. 1215, pp. 487–499.

[3] Brin, S., Motwani, R., Ullman, J. D., and Tsur, S. Dynamic itemset counting and implication rules for market basket data. In *Proceedings of the 1997 ACM SIGMOD international conference on Management of data* (1997), pp. 255–264.

[4] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning* (2013), pp. 108–122.

[5] Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.* O'Reilly Media, 2019.

[6] Han, J., Pei, J., and Kamber, M. *Data mining: concepts and techniques.* Elsevier, 2011.

[7] Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in science & engineering 9*, 3 (2007), 90–95.

[8] jingjingslides. jingjingslides slides. In *Slide titles* (2020).

[9] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[10] Python Core Team. *Python: A dynamic, open source programming language.* Python Software Foundation, Vienna, Austria, 2020.

[11] Raschka, S. Mlxtend: Providing machine learning and data science utilities and extensions to python's scientific computing stack. *The Journal of Open Source Software 3*, 24 (Apr. 2018).

[12] Tan, P.-N., Steinbach, M., and Kumar, V. *Introduction to data mining.* Pearson Education India, 2016.