CS 5785 – Applied Machine Learning – Lec. 15

Prof. Nathan Kallus, Cornell Tech
Scribe: TBD

Oct. 24, 2017

1 Association Rule Mining: Market Basket Analysis

We have seen some unsupervised learning methods by now:

- PCA, used mainly for dimensionality reduction.
- K-Means, used for clustering.

In this lecture we will cover a new method of unsupervised learning called *Market Basket Analysis*, used for *Association Rule learning/mining*.

1.1 A Motivating Story

Market Basket Analysis is a data mining technique used to study consumer purchases. It can help retailers understand what items are usually purchased together, develop marketing promotions, quantify purchasing trends for loyalty cards, manage inventory, etc.

Data scientists at Target generated a minor scandal through a provocative use of market basket analysis; see Figure **??**. Their algorithms predicted, based on a set of items bought by a certain household, that there was a pregnant woman in the house. Following that conclusion they mailed baby and pregnancy related promotions to the household. The father got angry and complained to Target that no one in his family is pregnant and that they should stop sending them such promotions. Later, the father found out that his teenage daughter was pregnant. After this incident, Target continued to use market basket analysis, but cleverly camouflaged their curated advertising pages amongst a set of standard promotions to avoid scaring off customers or creating a fear of being watched.

1.2 Support and Confidence

Definition Support of an itemset is the precentage of the dataset containing this itemset.

For example, in the dataset shown in Figure **??** the support for the itemset {soymilk} is $\frac{4}{5}$, since *soymilk* appears in 4 out of the 5 itemsets. The the support for the itemset {soymilk, diapers} is $\frac{3}{5}$, since *soymilk* and *diapers* appear together in 3 out of the 5 itemsets.

Figure 1: A New York Times Magazine article about Target's Market Basket Analysis scandal.

Transaction number	Items
0	soy milk, lettuce
1	lettuce, diapers, wine, chard
2	soy milk, diapers, wine, orange juice
3	lettuce, soy milk, diapers, wine
4	lettuce, soy milk, diapers, orange juice

Figure 11.1 A simple list of transactions from a natural foods grocery store called Hole Foods

Figure 2: Example itemsets from course book

diapers → beer

The most famous example of association analysis is diapers Þ beer. It has been reported that a grocery store chain in the Midwest of the United States noticed that men bought diapers and beer on Thursdays. The store could have profited from this by placing diapers and beer close together and making sure they were full price on Thursdays, but they did not.†

† DSS News, "Ask Dan! What is the true story about data mining, beer and diapers?" http://www.dssresources.com/newsletters/66.php, retrieved March 28, 2011.

Figure 3: Most common associative rule example

Definition Confidence of an association rule $\{A\} \rightarrow \{B\}$ is:

$$\frac{support(\{A, B\})}{support(\{A\})}$$

For example, the confidence of the association rule $\{diapers\} \rightarrow \{wine\}$ based on the dataset described in Figure **??** is:

$$conf(\{diapers\} \rightarrow \{wine\}) = \frac{support(\{diapers, wine\})}{support(\{diapers\})}$$

$$\Downarrow$$

$$conf(\{diapers\} \rightarrow \{wine\}) = \frac{\frac{3}{5}}{\frac{4}{5}} = \frac{3}{4} = 0.75$$

Meaning that in 75% of the transactions in the dataset that contain *diapers* our rule is correct, i.e., the itemsets contain also *wine*. In Figure **??** we see an example of a surprising association rule with high confidence, between diapers and beer.

What would be interesting is to find all the itemsets with a support greater than some given threshold, for example 80%. We could always use the *brute force* method and generate a list of every possible combination of items and for each combination count how frequently it occurs. In the dataset given in Figure **??** brute force is a viable option, but when the list of items is larger we will run into problems very quickly.

We make the following assumptions about the itemsets:

- Ignore the order of the objects, i.e., it doesn't matter which item passed through the checkout counter first.
- Ignore item count beyond 1 i.e. purchased item count+=1, regardless of quantity purchased

Let's say we are in a store with only 4 items. What are the possible combinations of these items? These are enumerated in Figure **??**. For each possible itemset size $0 \geq k \geq 4$ we have $\binom{4}{k}$ possible itemsets. This results in a total of 16 possibilities. This seems simple enough, but a store selling merely 100 items generates 1.26×10^{30} possible itemsets!

1.3 The Apriori Algorithm

There is a simple principle that can help us reduce the computational complexity of this called the *Apriori principle*. The Apriori principle will help us get what we want and will set the stage for association rule mining.

This term emphasizes the unique origin of this algorithm. This algorithm was devised by data miners (as opposed to data scientists or statisticians). The latter still cringe at the misuse of Latin (it should be "a priori") but the original term stuck.

Definition The apriori principle says that:
1. If an itemset is frequent, then all of its subsets are frequent.
2. If an itemset is infrequent, then its supersets are also infrequent.

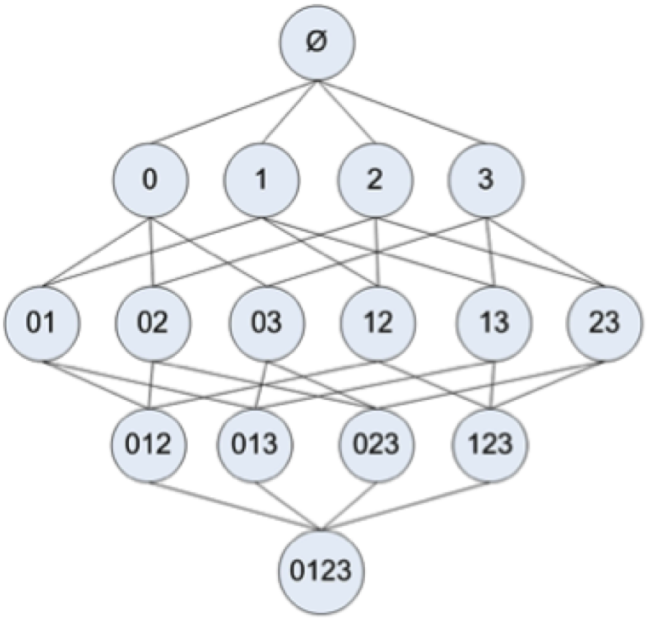

Figure 11.2 All possible itemsets from the available set {0, 1, 2, 3}

Figure 4: [Harrington]

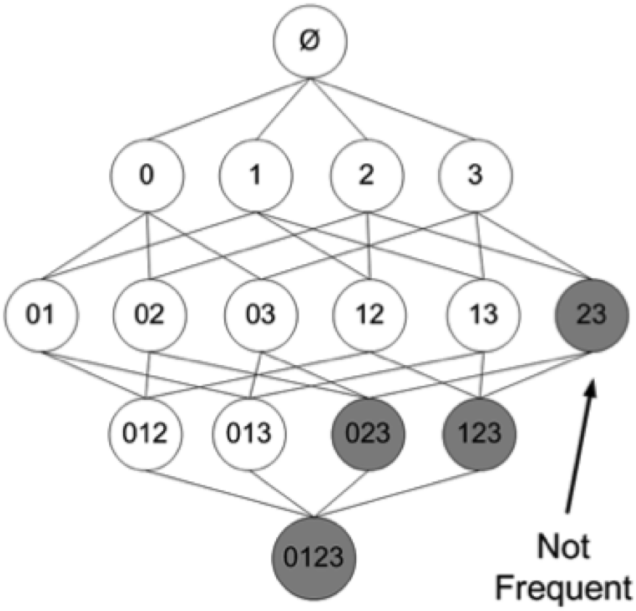

Figure 11.3 All possible itemsets shown, with infrequent itemsets shaded in gray. With the knowledge that the set {2,3} is infrequent, we can deduce that {0,2,3}, {1,2,3}, and {0,1,2,3} are also infrequent, and we don't need to compute their support.

Figure 5: [Harrington]

Figure **??** shows an example of the apriori principle in action. If we know that the itemset {2,3} is infrequent, i.e., does not meet a given threshold, we don't need to compute the support for of its supersets {0,2,3}, {1,2,3}, or {0,1,2,3} since there is no way they can meet our requirement.

Data: Dataset D, minimum support level t
Result: All itemsets in the dataset that have support larger than t

- Find all itemsets of size 1 that have a support level $\geq t$
- Combine the surviving itemsets to make itemsets of size 2
- Again, scan through each transaction and discard itemsets that fail to meet the support threshold t.
- Repeat this procedure as long as there are surviving itemsets (note that the maximum number of possible iterations is the amount of unique items in a basket).

Algorithm 1: Apriori Algorithm

We use the apriori principle to solve our problem in a way that only requires one pass through the dataset for each itemset size, see Algorithm **??**. The apriori algorithm gives us the list of high support itemsets which we are going to cast into a set of *association rules*. **Worst Case:**Algorithm will iterate through every item in the basket **Expectation** is that the *Apriori Algorithm* will significantly cut down from a brute force implementation

1.4 Association Rule Mining

We use the following notation:

- $\mathcal{K}$ is an itemset
- $\forall k \in \mathcal{K}$, $Z_k \in \{0, 1\}$ is an indicator for the kth item.

This model makes the following assumptions:

- Order of item appearance in set is ignored
- Item counts above 1 are ignored (i.e. the appearance of an item in a set is the only thing that matters)
- The Apriori Principle as stated above (if an itemset is infrequent then all of its supersets are also infrequent)

The items $k \in \mathcal{K}$ are partitioned into disjoint subsets $A \cup B = \mathcal{K}$ and written:

$$A \Rightarrow B$$

"antecedent" $\Rightarrow$ "consequent"

The support of the rule, denoted $T(A \Rightarrow B)$, is the same as the support of the itemset $\mathcal{K}$ from which it was derived:

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

Think of this as an estimate of observing both itemsets A and B in a randomly selected market basket, i.e., $Pr(A \text{ and } B)$. We can also define the *confidence*, or *predictability*, $C(A \Rightarrow B)$ of the rule as:

$$C(A \Rightarrow B) = \frac{T(A \Rightarrow B)}{T(A)}$$

Where:

- $T(A)$ estimates $Pr(A)$, i.e., $Pr(\prod_{k \in A} Z_k = 1)$.
- The *expected confidence* of the rule is defined as the support of consequent $T(B)$, which estimates $Pr(B)$
- The *lift* of the rule:

$$L(A \Rightarrow B) = \frac{C(A \Rightarrow B)}{T(B)} = \frac{\text{confidence}}{\text{expected confidence}}$$

The lift is an estimate of the association measure $\frac{Pr(A \text{ and } B)}{Pr(A)Pr(B)}$. For random chance the lift equals 1.

Example Given an itemset $\mathcal{K} = \{\text{peanut butter}, \text{jelly}, \text{bread}\}$, consider the rule $\{\text{peanut butter}, \text{jelly}\} \Rightarrow \{\text{bread}\}$. A support of 0.03 for this rule means that peanut butter, jelly and bread appeared together in 3% of the market baskets. Confidence of 0.82 for the rule implies that when peanut butter and jelly were purchased, 82% of the time bread was also purchased. If bread appeared in 43% of all market baskets then the rule would have a lift of 1.95.

Our goal is to mine association rules $A \Rightarrow B$ with high support and high confidence. The apriori algorithm gives us all the itemsets with high support. We want to find rules with high confidence such that for a given threshold c it holds that:

$$\{A \Rightarrow B \quad | \quad C(A \Rightarrow B) > c\}$$

An itemset $\mathcal{K}$ of size $|\mathcal{K}|$ has $2^{|\mathcal{K}|-1} - 1$ possible partitions into rules of the form $A \Rightarrow (\mathcal{K} - A)$ for a subset $A \subset \mathcal{K}$. A variant of the apriori algorithm can be used to find rules that survive the threshold c; see Figure **??**. The result of this algorithm is a collection of association rules satisfying:

$$T(A \Rightarrow B) > t \quad \text{and} \quad C(A \Rightarrow B) > c$$

All the resulting rules are stored in a database for easy querying.

Example *Display all transactions in which ice skates are the consequent that have confidence over* 80% *and support of more than* 2%. This would give us insight into items (antecedents) that predicate sales of ice skates.

The advantages of the Apriori Algorithm are that it works on huge datasets and that the rules are easy to interpret. The disadvantage of using the Apriori Algorithm is that rules with high confidence or lift, but low support, are not discovered.

For example, the rule vodka⇒ caviar has high confidence but the consequent, caviar, has very low sales volume (solutions for this situations might include separating to different associative rule sets by price range or adding specific additional models for products that have a high profit margin).

Figure **??** shows a few sample results from running the Apriori Algorithm on historical congressional voting patterns.

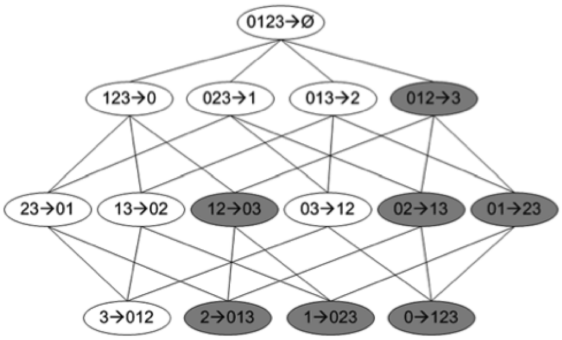

Figure 11.4 **Association rule lattice for the frequent itemset {0,1,2,3}. The gray area shows rules with a low confidence. If we find that 0,1,2 → 3 is a low confidence rule, then all other rules with 3 in the consequent (shaded) will also have a low confidence.**

Figure 6: [Harrington]

If		Then	confidence
Prohibiting Federal Funding of National Public Radio -- Yea	→	Republican	99.6%
Prohibiting Use of Federal Funds For Planned Parenthood -- Nay	→	Democrat	95.1%
Prohibiting the Use of Federal Funds for NASCAR Sponsorships -- Nay And Repealing the Health Care Bill -- Yea	→	Republican And Terminating the Home Affordable Modification Program -- Yea	95.8%

Figure 11.6 **Association rules {3} → {0}, {22} → {1}, and {9,26} → {0,7} with their meanings and confidence levels**

Figure 7: [Harrington]