

INFORMATION SYSTEMS EDUCATION JOURNAL

In this issue:

4. **Will Computer Engineer Barbie ® Impact Young Women's Career Choices?**
Cynthia J. Martincic, St. Vincent College
Neelima Bhatnagar, University of Pittsburgh at Johnstown
15. **Developing an Introductory Level MIS Project in Accordance with AACSB Assurance of Learning Standard 15**
Dana Schwieger, Southeast Missouri State University
25. **Adapting to Change in a Master Level Real-World-Project Capstone Course**
Charles C. Tappert, Pace University
Allen Stix, Pace University
38. **Market Basket Analysis for Non-Programmers**
Robert Yoder, Siena College
Scott Vandenberg, Siena College
Eric Breimer, Siena College
51. **Health Informatics as an ABET-CAC Accreditable IS Program**
Jeffrey P. Landry, University of South Alabama
Roy J. Daigle, University of South Alabama
Harold Pardue, University of South Alabama
Herbert E. Longenecker, Jr., University of South Alabama
S. Matt Campbell, University of South Alabama
63. **Factors Influencing Students' Decisions To Major In A Computer-Related Discipline**
Terri L. Lenox, Westminster College
Gayle Jesse, Thiel College
Charles R. Woratschek, Robert Morris University
72. **Beyond the Bake Sale: Fundraising and Professional Experience for Students Involved in an Information Systems Student Chapter**
Johnny Snyder, Colorado Mesa University
Don Carpenter, Colorado Mesa University
Gayla Jo Slauson, Colorado Mesa University
Joe Skinner, Colorado Mesa University
Cole Nash, ProVelocity
84. **Microsoft Enterprise Consortium: A Resource for Teaching Data Warehouse, Business Intelligence and Database Management Systems**
Jennifer Kreie, New Mexico State University
Shohreh Hashemi, University of Houston - Downtown
93. **Adjunct Communication Methods Outside the Classroom: A Longitudinal Look**
Anthony Serapiglia, St. Vincent College

The **Information Systems Education Journal (ISEDJ)** is a double-blind peer-reviewed academic journal published by **EDSIG**, the Education Special Interest Group of AITP, the Association of Information Technology Professionals (Chicago, Illinois). Publishing frequency is six times per year. The first year of publication is 2003.

ISEDJ is published online (<http://isedj.org>) in connection with ISECON, the Information Systems Education Conference, which is also double-blind peer reviewed. Our sister publication, the Proceedings of ISECON (<http://isecon.org>) features all papers, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the conference. At that point papers are divided into award papers (top 15%), other journal papers (top 30%), unsettled papers, and non-journal papers. The unsettled papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is about 45%.

Information Systems Education Journal is pleased to be listed in the 1st Edition of Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org.

2012 AITP Education Special Interest Group (EDSIG) Board of Directors

Alan Peslak
Penn State University
President 2012

Wendy Ceccucci
Quinnipiac University
Vice President

Tom Janicki
Univ of NC Wilmington
President 2009-2010

Scott Hunsinger
Appalachian State University
Membership Director

Michael Smith
High Point University
Secretary

George Nezek
Treasurer

Eric Bremier
Siena College
Director

Mary Lind
North Carolina A&T St Univ
Director

Michelle Louch
Sanford-Brown Institute
Director

Li-Jen Shannon
Sam Houston State Univ
Director

Leslie J. Waguespack Jr
Bentley University
Director

S. E. Kruck
James Madison University
JISE Editor

Nita Adams
State of Illinois (retired)
FITE Liaison

Copyright © 2012 by the Education Special Interest Group (EDSIG) of the Association of Information Technology Professionals (AITP). Permission to make digital or hard copies of all or part of this journal for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial use. All copies must bear this notice and full citation. Permission from the Editor is required to post to servers, redistribute to lists, or utilize in a for-profit or commercial use. Permission requests should be sent to Wendy Ceccucci, Editor, editor@isedj.org.

INFORMATION SYSTEMS EDUCATION JOURNAL

Editors

Wendy Ceccucci
Senior Editor
Quinnipiac University

Thomas Janicki
Publisher
University of North Carolina
Wilmington

Donald Colton
Emeritus Editor
Brigham Young University
Hawaii

Jeffry Babb
Associate Editor
West Texas A&M
University

Nita Brooks
Associate Editor
Middle Tennessee
State University

George Nezek
Associate Editor

ISEDJ Editorial Board

Samuel Abraham
Siena Heights University

Mary Lind
North Carolina A&T State Univ

Samuel Sambasivam
Azusa Pacific University

Alan Abrahams
Virginia Tech

Pacha Malyadri
Osmania University

Bruce Saulnier
Quinnipiac University

Gerald DeHondt II
Grand Valley State University

Cynthia Martincic
Saint Vincent College

Karthikeyan Umapathy
University of North Florida

Janet Helwig
Dominican University

Muhammed Miah
Southern Univ at New Orleans

Bruce White
Quinnipiac University

Scott Hunsinger
Appalachian State University

Alan Peslak
Penn State University

Charles Woratschek
Robert Morris University

Mark Jones
Lock Haven University

Peter Y. Wu
Robert Morris University

Market Basket Analysis for Non-Programmers

Robert Yoder
ryoder@siena.edu

Scott Vandenberg
vandenberg@siena.edu

Eric Breimer
ebreimer@siena.edu

Computer Science Department
Siena College
Loudonville, New York 12211, USA

Abstract

Market Basket Analysis is an important topic to cover in a Management Information Systems course. Rather than teach only the concept, our philosophy is to teach the topic using hands-on activities where students perform an analysis on a small but non-trivial data set. Our approach does not require knowledge of SQL, programming, or special software. Students use simple Microsoft Access functionality to find frequent itemsets and association rules. We believe this approach is a rigorous and engaging way to teach Market Basket Analysis that is most appropriate for an introductory course. Our follow-on Business Database course revisits the topic in a more technical manner where students write SQL queries. Thus, students are introduced to various SQL features through a fundamentally important topic that they have already seen in the prerequisite course. This paper describes the cognitive support structures used to introduce Market Basket Analysis, the details of how the activities are performed without SQL, and how we reinforce the topic with SQL in our Business Database course.

Keywords: management information systems, database systems, laboratory-based learning, association rules, market basket analysis, business intelligence

1. INTRODUCTION

For many businesses, understanding and acting upon purchasing patterns of customers using transaction data is a critical success factor. Students are accustomed to receiving "recommendations" for additional purchases of music, books, clothes, etc. How exactly are these "recommendations" generated and why are they often so relevant? The data mining technique called Market Basket Analysis (Julander, 1992) plays a critical role in many of the marketing strategies that students see every day. Our goal is to improve students' analytical and critical thinking skills by using an

in-depth study of the concepts and algorithms behind Market Basket Analysis. Data mining is a suggested topic for both the introductory MIS course and the business database course according to the IS 2010 Curriculum Guidelines published by AIS and ACM (Topi, 2010). Market Basket Analysis is a form of data mining that fulfills this requirement and can be relatively easily understood.

How can we motivate students to learn about this important and complex methodology using simple tools? In our Management Information Systems (MIS) course, we introduce our business students to Market Basket Analysis

(MBA) using case studies and a hands-on approach. The novelty of our approach is that students generate association rules (Agrawal, 1993) without directly using SQL or special software. Students implement the basic Apriori (pruning) Algorithm (Agrawal & Srikant, 1994) using the query design view in Microsoft Access. In our Business Database course, students revisit the topic but use SQL, which broadens their perspective and reinforces concepts from a more technical point of view.

Our goal is to go beyond textbook concept coverage and provide hands-on exercises without requiring special software that can obscure fundamental processes, and without programming in a language such as Java (Witten, 1999) or requiring advanced knowledge of algorithms (Zaki, 2000), which is inappropriate for an introductory course.

Our institution, courses, and labs

We are a liberal arts college of approximately 3000 students with an AACSB accredited School of Business. Our MIS course, a required core business course, is taught within the Computer Science department, which offers a BS in Computer Science and a minor in Information Systems. MIS is an introductory level course that requires only spreadsheet proficiency as a prerequisite. The course consists of two hours of lecture and two hours of lab each week. Lab sections are restricted to a maximum of 16 seats where students typically work in pairs at a computer with dual monitors. Our business database course, which has MIS as a prerequisite, is taught in a 16-seat classroom in which each student has a computer; the format is a mix of lecture and workshop activities.

The remainder of this paper will briefly trace the evolution of our MIS labs, describe the MBA lab activities and expected student outcomes in more detail, outline the approach to MBA taken in our business database course, and draw some conclusions based on our experiences teaching MBA to business students.

2. EVOLUTION OF OUR MIS LABS

The value of a hands-on approach to learning computing concepts has been recognized in the IS 2010 Curriculum Guidelines (Topi, 2010) and elsewhere; we also provide specific evidence of its utility for learning MBA in the Conclusions section of this paper. The material presented in labs is the driving force for integrating content and experience for all sections of the course. We

implemented a collaborative approach to scaling our MIS course by creating a shared repository of lab and lecture materials in Blackboard that fosters collaboration among faculty teaching the course. A faculty member can contribute new lab ideas and corrections using an editing review system, where at least two other faculty members must review the suggested changes and perform the lab in its entirety to ensure continuity and cohesiveness (Breimer, Cotler, & Yoder, 2009).

Our labs have a "triad" structure that incorporates (i) theory from the textbook or lectures and (ii) practical case studies with (iii) information technology, such as Excel, Access, Geographic Information Systems, and Radio Frequency ID readers. All labs have pre-lab reading assignments and an online quiz to introduce the lab. The in-lab experience is fast paced, where students are paired in teams to use technology and learn basic concepts to solve problems and to work through examples. Lastly, students individually complete a post-lab assignment to synthesize material and to reflect on the lessons learned in the lab. A web-based lab delivery and assessment system is under development.

3. OUR MARKET BASKET ANALYSIS LAB

The MBA lab, inspired by a chapter from Larose's (2005) data mining text, presents a complete analysis of transaction data for a farmer's vegetable stand. We added specifics on how to accomplish a market basket analysis using only the graphical interface of Access (no programming or explicit SQL). The dataset presented in the book is non-trivial but easy to understand. We show students a bottom-up approach to generate frequent itemsets (groups of items purchased together) and how to compute measures of Support and Confidence for the association rules derived from these itemsets. The next section provides definitions of these terms. A good introduction to MBA can be found in Berry & Linoff's text (2004).

Students learn that finding frequent itemsets requires computing cross products of sets, which increases the size of the problem exponentially. It becomes obvious that removing (pruning) non-frequent itemsets of size N before moving on to itemsets of size N+1 makes the analysis much more tractable. The Apriori principle is introduced as a theoretical justification for removing non-frequent itemsets without affecting the solution. Thus, students gain the valuable experience of actually performing a

market basket analysis, using theory that can be applied to a real business problem.

Pre-lab activities (20% of lab grade)

In order to smooth the presentation and understanding of Support and Confidence during the lab session, we provide students with a worksheet to review the concepts of probability and conditional probability. We also provide a set of PowerPoint slides, to be presented before the lab session, that introduces the concepts of Transactions, Itemsets, Association Rules, Support, and Confidence:

Transaction: the purchase of a collection or "basket" of items by a customer in a shopping cart.

Itemset: a subset of items selected for purchase in a single transaction for analysis purposes. We evaluate itemsets of size 1 (singles), 2 (doubles), and 3 (triples).

Association Rule: the itemsets that occur frequently in transaction data (Support), and items that often occur together in individual baskets (Confidence) are good candidates for appearing in general Association Rules. As an example, the association rule $A \rightarrow B$ means "if item A is purchased in an itemset, then item B is also purchased in that itemset".

Support: this is the percentage of transactions containing all items in the itemset compared to the total number of transactions (baskets). For example, if 432 baskets contain both Asparagus and Beans, and we have 1000 baskets, then the support is $432/1000 = 0.432$ (43.2%). Support is the probability that the items occur together in baskets, and is a measure of how frequent the itemset occurs within our transaction data. The association rules $Asparagus \rightarrow Beans$ and $Beans \rightarrow Asparagus$ will have the same support (43.2%).

Confidence: this measure applies only to rules, not to itemsets. Support by itself is an imperfect measure of the quality of a rule. We need a way to further measure the predictive accuracy of the candidate association rules that takes into account how strongly items are associated with each other. This measure is called Confidence, the probability that an item B is in the basket given that item A is already in the basket. You may recognize this as Conditional Probability (Bayes' theorem). Confidence can be computed by $Support(A \& B) / Support(A)$. Note that the Confidence for A,

given B is in the basket is different than the Confidence for B, given that A is in the basket. The assertion "All people who buy Peanut Butter also buy Bread" does not necessarily imply that "All people who buy Bread also buy Peanut Butter."

The pre-lab activities introduce Market Basket Analysis as a form of unsupervised data mining, which is in turn a form of business intelligence.

We ask students to read the brief introduction in the MIS course textbook (Kroenke, 2010), followed by a short (7 page) white paper on the benefits of MBA from the perspective of a business (Megaputer.com). The paper reviews MBA terminology and describes actions businesses can take based on these rules, such as product placement or pricing strategies.

Finally, the pre-lab introduces students to the data that will be used during the lab. The raw data are stored in a table with two columns: transaction (basket) number and item name (Figure 1).



Basket	Item
1	Broccoli
1	Corn
1	Green Peppers
2	Asparagus
2	Corn
2	Squash
3	Beans

Figure 1: Market Basket data format

The pre-lab guides the student through the process of creating a query to count, for each item, the number of baskets that contain that item. An online quiz concludes the pre-lab activities. All of our pre-lab activities are due by the start of the lab session.

Because of the complexity of the subject, this pre-lab is the most intensive of any of our labs. It is important that students carefully complete the pre-lab material beforehand.

In-lab activities (50% of lab grade)

The in-lab activity focuses on performing the steps in the MBA algorithm shown in Figure 2. During the lab, students answer questions on their worksheet that test their understanding of crucial steps and verify their progress. Students create tables containing all possible single, double, and triple itemsets; create queries to calculate the candidate association rules; prune

the rules table of non-frequent itemsets (those with less than 25% support); and copy only the frequent itemset rows from the transaction data that will participate in the next round.

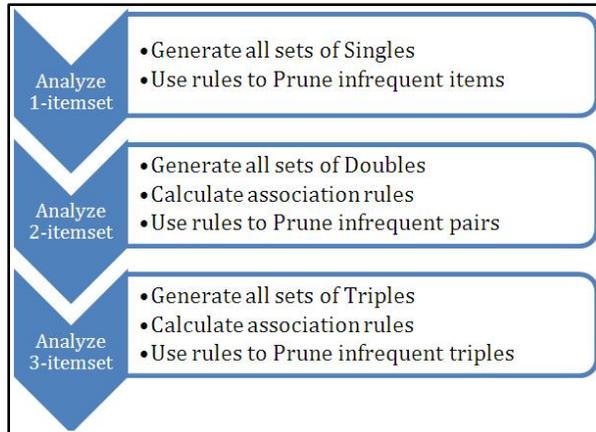


Figure 2: The MBA Algorithm

There are some nuances to making the tables and action queries that will be described here. We have designed the lab so that students can recover from mistakes easily without performing large sections of the lab over again. The detailed design for each query mentioned below is shown in the appendix.

For Single itemsets

The students complete the following steps:

- Copy the original transaction data, the "Veggie-Table," to a new table called Singles.
- Compute the Support for every single item: Using the Access query wizard, create a make-table query named Make-SingleRules that creates the table named SingleRules that includes Support as a calculated field by dividing *ItemCount* by *NumBaskets* (Figure 3). *ItemCount* is the number of baskets containing the item and *NumBaskets* is the number of transactions in the dataset. (Note that we call this table "SingleRules" for consistency with the upcoming "DoubleRules" and "TripleRules", even though technically these single itemsets do not give rise to association rules.)
- Remove infrequent items: create a *delete* action query named Prune-SingleRules to prune the SingleRules table by adding a criterion of < 0.25 to the Support column in the query design.
- Retain just the item names and basket numbers for these frequently-purchased

items: create a new query named Make-FreqSingles that makes the FrequentSingles table by selecting rows in the Singles table that have frequent items in them, effectively pruning the Singles table.

Item	ItemCount	NumBaskets	Support
Asparagus	6	14	42.86%
Beans	10	14	71.43%
Broccoli	5	14	35.71%
Corn	8	14	57.14%
Green Peppers	5	14	35.71%
Okra	1	14	7.14%
Squash	7	14	50.00%
Tomatoes	6	14	42.86%

Figure 3: The SingleRules Table

For Double itemsets

This phase of the lab requires the students to build upon what was created before, by creating itemsets of 2 items, computing Support for both items together, and keeping only frequent itemsets of size 2, as follows:

- Generate all pairs of items purchased together: create a make-table query named Make-Doubles that joins the FrequentSingles table with itself to generate all combinations of two items and place it in the new Doubles table. Note that students are using the Apriori Principle here since they are only including frequent single items, not all items.
- Data cleaning and consistency: to avoid itemset duplicates, such as {corn, corn} or pair reversals of items from the same basket, such as {asparagus, beans} and {beans, asparagus}, we add the criterion that the second item column be less than the first item column (sorted order).
- Add support: create a query named Make-DoubleRules that includes the Doubles table in the query with additional calculated fields to compute Support as *ItemCount* divided by *NumBaskets* for both items together, creating table DoubleRules (Figure 4).

FirstItem	SecondItem	NumBaskets	ItemCount	Support
Asparagus	Beans	14	5	35.71%
Asparagus	Broccoli	14	1	7.14%
Asparagus	Corn	14	2	14.29%

Figure 4: A portion of the DoubleRules table

- Remove infrequent itemsets: create a delete query named Prune-DoubleRules that

prunes the DoubleRules table by removing the low-support pairs using a criterion of <0.25 in the Support column. Thus, the second and third rows would be removed from the table in Figure 4.

- Prepare final frequent pairs list: finally, we create a query named Make-FreqDoubles that copies only the item pairs with high support, by including the Doubles and DoubleRules tables in the query with the criteria that the first item in the Doubles table matches the first item in DoubleRules, and that the second item in Doubles matches the second item in DoubleRules. In this manner, only the frequent item pairs are selected from the transaction basket data to participate in the next round. The resulting table is shown in Figure 5.

Basket	FirstItem	SecondItem
12	Asparagus	Beans
9	Asparagus	Beans
6	Asparagus	Beans
13	Asparagus	Beans
5	Asparagus	Beans
6	Asparagus	Squash
9	Asparagus	Squash
2	Asparagus	Squash
13	Asparagus	Squash
12	Asparagus	Squash
3	Beans	Corn

Figure 5: A portion of the FreqDoubles table

At this juncture, students perform a little analysis based on the SingleRules and DoubleRules tables they created earlier: using a supplied spreadsheet, students calculate the Support, Confidence, and overall Quality measures for Doubles itemsets and their reverse, e.g., {Asparagus, Beans} and {Beans, Asparagus}. Since the Confidence measure is a conditional probability, the order of items matters. We combine (multiply) the measures of Support for both items (the proportion of baskets with those items in them) and Confidence (how strongly the items are linked with each other) into a single measure called Quality that expresses Confidence adjusted for how often the items are sold together. In the "Doubles Analysis" worksheet in the appendix, Quality is computed as the "Support for I1, I2" column times the "Confidence" column.

For Triple itemsets

The students first create a list of all the triples of items that are purchased together: they create a Make-Table query named Make-Triples that

joins the FreqDoubles table to itself (as FreqDoubles_1) and creates a table named Triples. Note that the Apriori Principle is in play here again: they are using the "FreqDoubles" table as a starting point, not the "DoubleRules" table, which includes all pairs (not just the frequent ones). For Triples, the Basket ID and the first item from both tables must match. Thus, we are selecting rows from both tables that start with the same Basket ID (by a join) and first item, and generating all of their combinations, e.g., {4,Asparagus,*,*} in FreqDoubles with {4,Asparagus,*,*} in FreqDoubles_1. We also add a criterion that the second item name be less than the third item name to avoid duplication and to keep the triples in alphabetical order, as shown in Figure 6.

Basket	FirstItem	SecondItem	ThirdItem
6	Asparagus	Beans	Squash
9	Asparagus	Beans	Squash
13	Asparagus	Beans	Squash
12	Asparagus	Beans	Squash
13	Beans	Corn	Squash
3	Beans	Corn	Squash
2	Beans	Corn	Tomatoes

Figure 6: A portion of the Triples table

We leave the task of making the TripleRules table and pruning it to make FreqTriples as a paper exercise for the post-lab. There are only a few rows remaining at this point, and we want students to focus on the process and not on the complexities of the queries for triple itemsets.

Post-lab activities (30% of lab grade)

The in-lab activities focused on the creation of itemsets of sizes 1, 2, and 3; including the creation of frequent itemsets of size 1 and 2 and on association rules involving itemsets of size 2. The post-lab first asks the students to extend the analysis of the triples (itemsets of size 3) by deriving the association rules (with support and confidence) for these triples by filling in the TripleRules worksheet table (Figure 7).

Students then prune the TripleRules worksheet table by drawing a line through rows with Support of less than 25%. Lastly, using the Triples table from the in-lab session and the pruned TripleRules table, students manually create their own FreqTriples table, as shown in Figure 8.

TripleRules					
First Item	Second Item	Third Item	Triple Count	NumBaskets	Support
Asparagus	Beans	Squash	4	14	28.6%
Beans	Corn	Squash	2	14	14.3%
Beans	Corn	Tomatoes	3	14	21.4%
Beans	Squash	Tomatoes	2	14	14.3%

Figure 7: The TripleRules worksheet

FreqTriples			
Basket	First Item	Second Item	Third Item
6	Asparagus	Beans	Squash
9	Asparagus	Beans	Squash
12	Asparagus	Beans	Squash
13	Asparagus	Beans	Squash

Figure 8: The Frequent Triples worksheet

In the post-lab, we limit the student to rules in which the left-hand side contains two items: i.e., we are interested in rules of the form

$$A,B \rightarrow C$$

but not rules of the form $A \rightarrow B,C$. In the post-lab exercise, students are asked to fill in the "Triples Analysis" worksheet containing the Frequent Triples and their Support, Confidence, and Quality values (shown in the Appendix).

Using the results from the in-lab and post-lab activities, students are asked to make recommendations to the business owner based on the association rules they discovered.

The post-lab also includes some questions that invite the student to consider the importance of the Apriori rule when generating frequent itemsets. The Apriori rule states: *if an itemset is not frequent, then adding another item to the itemset will not make it more frequent*. This is the driving principle that permits pruning of non-frequent itemsets to reduce computational complexity while maintaining correctness.

4. MBA WORKSHOP IN OUR BUSINESS DATABASE COURSE

Our business database course has MIS as a prerequisite, so most students have already performed the lab described in the previous section. The database course textbook (Kroenke & Auer, 2010) contains just 2 pages on MBA and using SQL to derive rules, so the in-class workshop we describe next comprises the bulk of the students' learning materials for this topic.

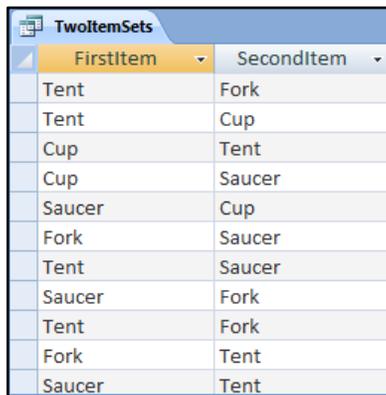
After basic concepts are briefly reviewed in a mini-lecture, the worksheet is distributed to students. The worksheet begins with a short pencil-and-paper exercise that presents some market basket data and asks the students to list all the itemsets that have at least 60% Support and, based on those itemsets, all the association rules that have at least 60% Confidence.

The remainder (and the bulk) of the exercise focuses on generating itemsets of size 2, then computing Support and Confidence for association rules based on those itemsets. Extending these techniques to triples and beyond is fairly straightforward and could be incorporated into a homework assignment or other activity.

The first task is to complete an SQL query that will generate all pairs of items purchased together (i.e., itemsets of size 2). The raw transaction data are in a table named *Order_Data*, with columns *OrderNumber* and *ItemName*. In the following query, the students are given the portion in **bold** and must complete the portions in *italics*:

```
SELECT O1.ItemName AS FirstItem,
       O2.ItemName AS SecondItem
FROM Order_Data AS O1, Order_Data AS O2
WHERE O1.OrderNumber = O2.OrderNumber AND
       O1.ItemName <> O2.ItemName;
```

This query works well as an introduction to the concept of self-join (joining a table to itself) in SQL. If the students have already seen that concept, then they can be asked to write the entire query from scratch. The intuition behind the WHERE clause is that you want two items that were (a) purchased in the same transaction (the "=" comparison) and (b) are not the same item (the "<>" comparison). The students then save this query as "TwoItemSets", in effect creating a view (in Access, a named query can be used in subsequent queries just as if it were a table). The result of this query is shown in Figure 9. Note the duplication of some pairs (which are purchased together more than once)



FirstItem	SecondItem
Tent	Fork
Tent	Cup
Cup	Tent
Cup	Saucer
Saucer	Cup
Fork	Saucer
Tent	Saucer
Saucer	Fork
Tent	Fork
Fork	Tent
Saucer	Tent

Figure 9: Pairs of items (“TwoItemSets”) using SQL (partial result).

The next step is to create and save a new query called “Rules” that will be used to compute the Support and Confidence for each entry in TwoItemSets. This is done in two steps, first by simply counting the number of occurrences of each itemset, then by converting that count into a percentage. The students are first asked to complete the Rules query (given portions in bold, to complete portions in italics):

```
SELECT FirstItem , SecondItem , COUNT(*) AS  
SupportCount  
FROM TwoItemSets  
GROUP BY FirstItem, SecondItem;
```

This is a fairly standard GROUP BY query, but it is a nice introduction to the concept of using a query as the basis for another query (outside of Access, these stored queries are called views). Now that the students have every itemset of size 2 and the count of orders containing that itemset, they can enhance the query to convert the SupportCount into a more useful percentage format (SupportCount divided by total number of orders). The total number of orders can be computed by the following query:

```
SELECT COUNT(DISTINCT OrderNumber) FROM  
Order_Data;
```

Unfortunately, Microsoft Access does not support the standard COUNT(DISTINCT) syntax, so the following must be used instead (this is given to the students):

```
SELECT COUNT(*) FROM (SELECT DISTINCT  
OrderNumber FROM Order_Data);
```

The students are then asked to embed (nest) that query in the SELECT clause to help compute the support as a percentage; the Rules query now looks like this:

```
SELECT FirstItem , SecondItem ,  
COUNT(*) AS SupportCount,  
COUNT(*) / (SELECT COUNT(*) FROM  
(SELECT DISTINCT OrderNumber  
FROM Order_Data)) AS SupportPercent  
FROM TwoItemSets  
GROUP BY FirstItem, SecondItem;
```

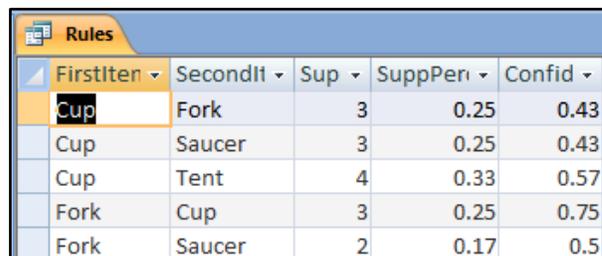
Note that some of the nested query was given to the students -- the total number of orders in the database.

The last main task in the exercise is to add a column to the Rules query that will compute the Confidence of the rule. At this point it becomes useful to point out to the students that they may have noticed some repetition in the Rules query/table: There is a row in the table for items A and B as well as a row in the table for items B and A. But those rows represent the same itemset. Now, however, that repetition becomes useful, since while the two rows represent the same itemset, they represent distinct association rules: as indicated earlier in this paper, A→B is not the same rule as B→A. We interpret the “FirstItem” column as containing the left-hand side of the rule.

The Confidence column can be computed by dividing the SupportCount for the itemset by the number of times the left-hand side of the rule (FirstItem) is purchased overall (this is simply the definition of “Confidence” for an association rule). The new column in the Rules query thus looks like this, where again the portion in italics must be added by the student:

```
COUNT (*) / (SELECT COUNT(*) FROM Order_Data  
WHERE ItemName = FirstItem) AS Confidence
```

The result of the Rules query is shown in Figure 10.



FirstItem	SecondItem	Sup	SupPer	Confid
Cup	Fork	3	0.25	0.43
Cup	Saucer	3	0.25	0.43
Cup	Tent	4	0.33	0.57
Fork	Cup	3	0.25	0.75
Fork	Saucer	2	0.17	0.5

Figure 10: partial final result of Rules query

Now the Rules query contains all rules involving itemsets of size 2, plus the Confidence and Support for each. The final activity in the worksheet is to write an SQL query to retrieve only those rules that have at least a minimum

level of support and confidence. For example, if we wish to see all rules with support at least 20% and confidence at least 50%, then we have the following query:

```
SELECT *
FROM Rules
WHERE SupportPercent >= .2 AND Confidence >= .5;
```

Note that throughout this worksheet, the amount of "blank space" to be filled in by the student (i.e. the amount of italics in the SQL queries above) can be varied. What is shown in italics here probably represents a minimum of what the students should be expected to generate. If more time is available, or if the students are more experienced with SQL, then they should be expected to contribute a higher percentage of the query texts. Also, as mentioned above, extending these techniques to triples and larger itemsets can be incorporated into the course as well.

5. RESULTS and CONCLUSION

In the final exam for one of our sections of the MIS course, the score on the market basket question (which tested their ability to understand association rules and to reason about and compute support and confidence) was about 8 points higher (out of 100) than was the score on the exam overall, indicating that the approach we have taken is working.

Although we did not survey students about this lab specifically, students seemed to tolerate this lab well. Students received an average of about 84% for the lab score. Most students liked the labs overall and found them to be "*challenging*" and indicated that "*lab material will be very useful in the future.*" Another student wrote: "*my favorite thing about the course was the labs. I was able to learn Microsoft Access through this course, which I know will be useful in many careers.*" We plan to add assessment measures for each lab in the future.

There was one negative comment in our MIS course evaluations about this lab: "*some of the labs especially the market basket lab were too complicated.*" We recognize that the MBA lab takes a little extra time to prepare for than most of our lab activities. A few students needed a little help with the post-lab worksheet, but once we showed them how to compute one row of the table, they could easily do the rest.

In the Business Database course, student evaluations indicate that the hands-on approach

is working, and there is no reason to suspect it is not working with the MBA topic in particular: Since changing to the hands-on approach, overall course ratings have risen (on a 10-point scale) nearly 1.5 points; the "are classes interesting" question has risen over 1.1 points, and the question about encouraging critical thought has risen over 0.6 points. In addition, student comments that the course material is sometimes "boring" or "dull" went from about 4 per section to 0 after the change.

Final exam scores in the Business Database course also indicate the benefits of our approach: on exams of roughly similar difficulty, scores rose by about 9 points when the approach changed to being more hands-on.

We want to encourage active learning that goes beyond lecture slides or text readings. We have to go the extra step. In an MIS course, we want students to use information systems to solve problems. We believe, based on our own experiences (anecdotal, student evaluations, and exams scores) that the best way to understand association rules, Support, and Confidence is through hands-on activities with actual data. Using database tools to perform a Market Basket Analysis achieves this goal. Students experience "mining" association rules by creating tables and queries using Microsoft Access. In our follow-on Business Database course, they experience MBA from a different perspective by writing SQL queries that generate the association rules. In both approaches, students learn MBA fundamentals and useful data analysis techniques in a hands-on fashion: they not only learn the concepts, but will be able to claim experience putting the concepts into practice, which is not possible without the hands-on experience we have provided for them.

6. REFERENCES

- Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD international conference on Management of data (SIGMOD '93)*, 207-216.
- Agrawal, R., & Srikant, R. (1994). Fast Algorithms for Mining Association Rules in Large Databases. *Proceedings of the 20th VLDB Conference*, 487-499.
- Berry, M., & Linoff, G. (2004). *Data Mining Techniques, Second Edition*. Wiley, Indianapolis, IN.

- Breimer, E., Cotler J., & Yoder R. (2009). Co-"Lab"oration: A New Paradigm for Building a Management Information Systems Course, *Information Systems Education Journal*, 8(2). From <http://isedj.org/8/2/>
- Julander, C. (1992) Basket Analysis: A New Way of Analysing Scanner Data. *International Journal of Retail & Distribution Management* 20 (7), 10-18.
- Kroenke, D. (2010) *Using MIS, 3rd Edition*, Pearson Prentice Hall, Upper Saddle River, NJ.
- Kroenke, D., & Auer, D. (2010). *Database Processing*, 11th Edition. Pearson Prentice Hall, Upper Saddle River, NJ.
- Larose, D. (2005). *Discovering Knowledge in Data: An Introduction to Data Mining*, John Wiley & Sons, Hoboken, NJ.
- Megaputer.com (undated). *Market Basket Analysis: A White Paper*. Retrieved November 20, 2007 from <http://megaputer.com>
- Topi, H., Valacich, J., Wright, R., Kaiser, K., Nunamaker, J., Sipior, J. & de Vreede, G. (2010) *IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems*. Retrieved August 20, 2011 from <http://www.acm.org/education/curricula/IS%202010%20ACM%20final.pdf>
- Witten, I., & Frank, E., (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, Waltham, MA.
- Zaki, M. (2000). Scalable algorithms for association mining. *IEEE Transactions on Knowledge and Data Engineering* 12, 372-390.

7. APPENDIX – Query design details and analysis worksheets

Field:	Item	ItemCount: Item	NumBaskets: DLast("[Basket]","Veggie")	Support: [ItemCount]/[NumBaskets]
Table:	Singles	Singles		
Total:	Group By	Count	Group By	Expression
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:				

Make-SingleRules Query Design

The screenshot shows the 'SingleRules' query design view. A table 'SingleRules' is selected, showing fields: Item, ItemCount, NumBaskets, and Support. Below the design grid, the following table is visible:

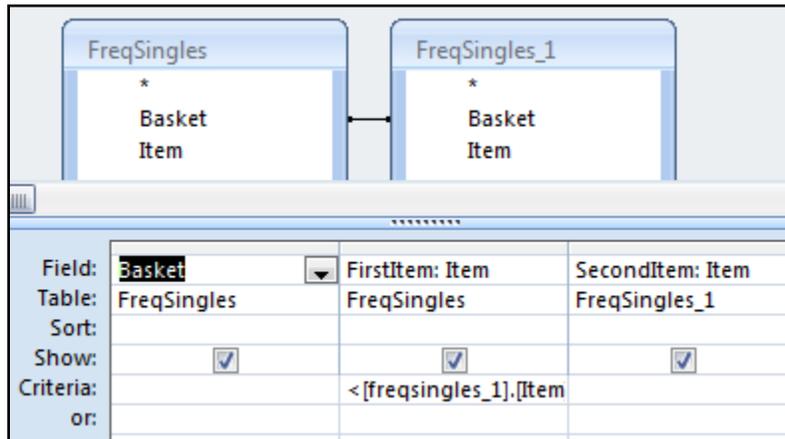
Field:	Item	ItemCount	NumBaskets	Support
Table:	SingleRules	SingleRules	SingleRules	SingleRules
Delete:	Where	Where	Where	Where
Criteria:				<0.25
or:				

Prune-SingleRules Query Design

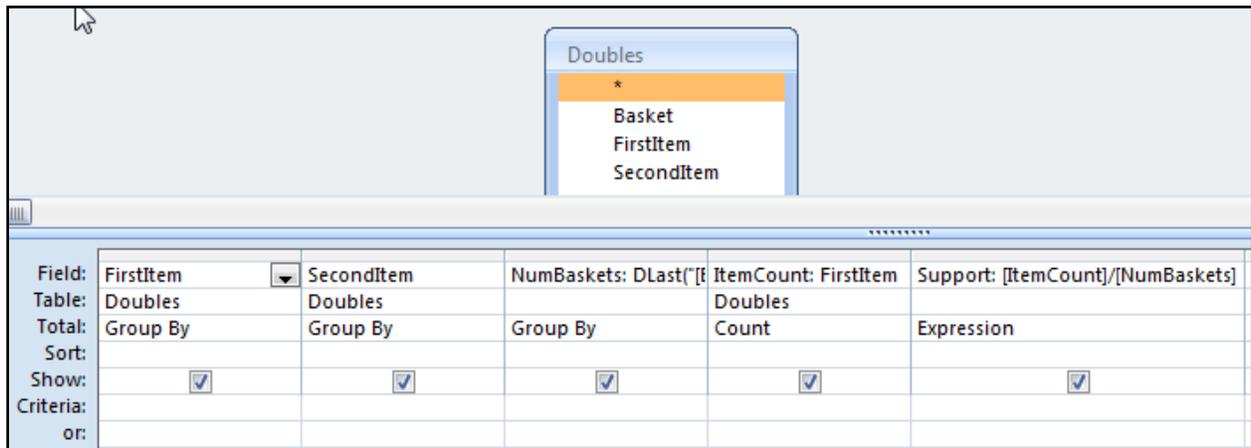
The screenshot shows the 'Make-FreqSingles' query design view. Two tables, 'SingleRules' and 'Singles', are selected. The 'SingleRules' table has fields: Item, ItemCount, NumBaskets, and Support. The 'Singles' table has fields: Basket and Item. Below the design grid, the following table is visible:

Field:	Basket	Item	
Table:	Singles	Singles	
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:		= [SingleRules].[Item]	
or:			

Make-FreqSingles Query Design



Make-Doubles Query Design



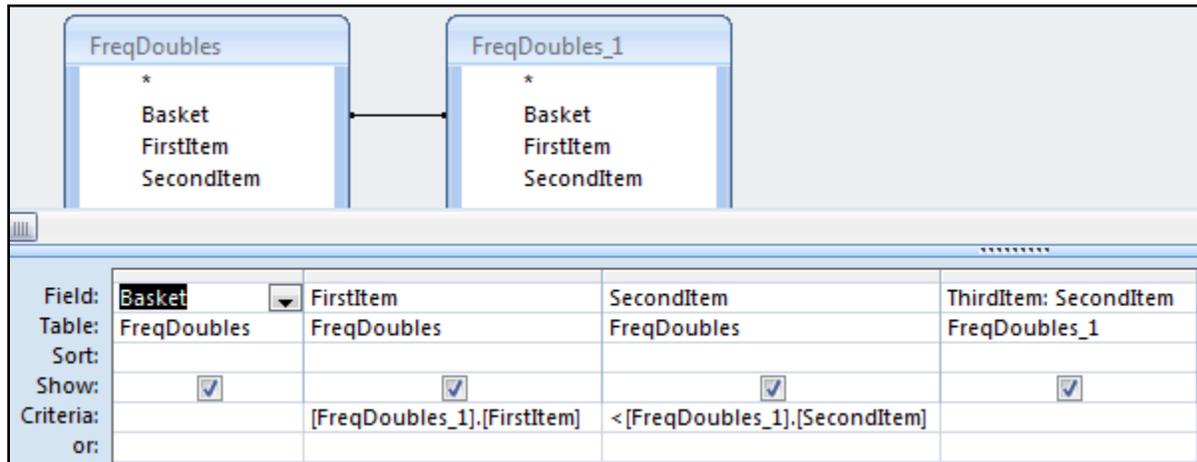
Make-DoubleRules Query Design

Field:	FirstItem	SecondItem	NumBaskets	ItemCount	Support
Table:	DoubleRules	DoubleRules	DoubleRules	DoubleRules	DoubleRules
Delete:	Where	Where	Where	Where	Where
Criteria:					<0.25
or:					

Prune-DoubleRules Query Design

Field:	Basket	FirstItem	SecondItem
Table:	Doubles	Doubles	Doubles
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		[doublerules].[firstitem]	[doublerules].[seconditem]
or:			

Make-FreqDoubles Query Design



Make-Triples Query Design

Item1	Item2	Support I1,I2	Support I1	Confidence	Quality
Asparagus	Beans	35.71%	42.86%	83.33%	29.76%
Asparagus	Squash	35.71%	42.86%	83.33%	29.76%
Beans	Squash	42.86%	71.43%	60.00%	25.71%
Broccoli	Gr Peppers	28.57%	35.71%	80.00%	22.86%
Corn	Beans	35.71%	57.14%	62.50%	22.32%
Gr Peppers	Broccoli	28.57%	35.71%	80.00%	22.86%
Squash	Beans	42.86%	50.00%	85.71%	36.73%
Squash	Asparagus	35.71%	50.00%	71.43%	25.51%

Doubles Analysis Worksheet

Triples Confidence and Quality									
Fill in veggie names from 3-itemsetF here			Count {A,B,C} 1	Num Baskets	{A,B,C} Support ²	Count pair purchased first ³	Support for Pair Purchased first	Confidence ⁴	Quality ⁵
A= aspar	B= beans	C= squash	4	14	4/14= 0.29	5	5/14=.36	.29/.36= .79	.28*.79= .224
A= aspar	C= squash	B= beans	4	14	4/14= 0.29	5	5/14=.36	.29/.37= .79	.28*.79= .24
B= beans	C= squash	A= aspar	4	14	4/14= 0.29	6	6/14=.43	.29/.43= .66	.28*.66= .19

Triples Analysis Worksheet