Market Basket Analysis for Digital Businesses

Handout + Assignments

Joseph Richards, Professor
CSU Sacramento

Contents

1 Introduction: What Market Basket Analysis Is (and Why It's Useful)

Market Basket Analysis (MBA) is a data mining technique used to discover co-occurrence relationships among activities performed by users. In retail, it uncovers which products customers tend to buy at the same time. This insight is gold for merchandising and layout because it allows retailers to move from guessing to data-driven decisions about what to place where.

What It Means: Shaping the Shopping Experience

At its core, this strategy is about using the **co-occurrence structure** of your sales data to engineer a more intuitive and profitable shopping environment, both in-person and online. It's about arranging your store—physical or digital—to mirror how customers naturally shop.

- **Physical Co-location:** This is the most direct application. If your data shows a strong association between pasta and pasta sauce, placing them in the same aisle is a no-brainer. But it can be more nuanced. Perhaps you find a strong link between greeting cards and high-end chocolate boxes. Placing a small, elegant display of chocolates in the card aisle can capture a significant number of impulse buys. This transforms the store layout from a simple product catalog into a guided shopping experience.
- **Digital Adjacency:** In e-commerce, this translates to the layout of a digital page. When a customer views a product page for a DSLR camera, the "Frequently Bought Together" or "Customers Also Bought" section should be populated by items with the highest **lift**—like memory cards, a camera bag, and a tripod. This isn't just a random upsell; it's a data-backed recommendation that serves a genuine customer need. See Figure: 1

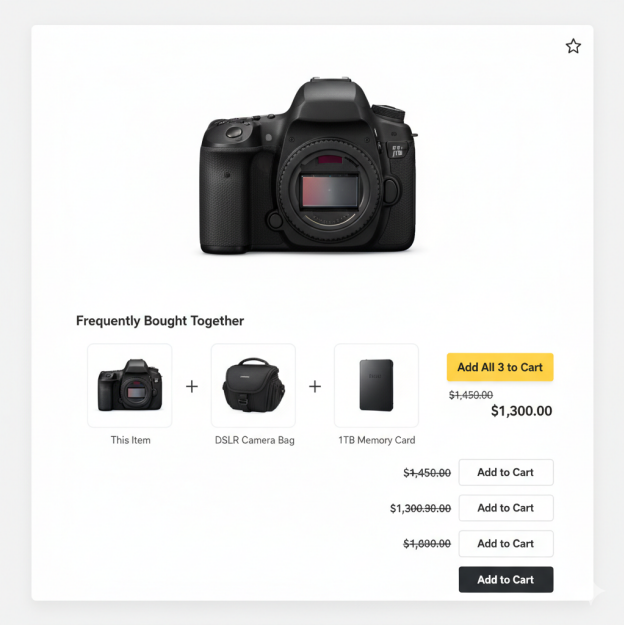

Figure 1: An e-commerce "Frequently Bought Together" section.

- **Scenario**

  An online electronics store detects that a user has recently browsed several pages for **4K Smart TVs** on their website, but hasn't made a purchase. Market Basket Analysis shows a very high lift between 4K Smart TVs and **Soundbars** (people often buy them together to complete their home theater experience) and also with premium **HDMI Cables**.

Market Basket Analysis (MBA) Application

Instead of just retargeting the user with an ad for the TV they viewed, the retailer uses MBA insights to craft a more compelling ad. The ad creatively presents the primary item (the 4K TV) alongside its highly associated, complementary products (Soundbar, HDMI Cable). The goal is to remind the user about their TV interest *and* suggest a complete solution. The ad itself acts as the virtual adjacency.

Implementation and Example

How to Implement

1. **Behavioral Trigger:** User views multiple 4K Smart TV product pages.
2. **MBA Lookup:** The advertising system queries the MBA database for items with the highest lift when purchased with 4K Smart TVs. It identifies Soundbars and premium HDMI cables as strong candidates.
3. **Dynamic Ad Generation:** A banner ad is dynamically generated. It features the 4K Smart TV prominently, visually places the Soundbar and HDMI cables nearby, and uses copy like "Complete your Home Cinema."
4. **Targeting:** This ad is served to the user on other websites they visit (e.g., a sports news site, a recipe blog).
5. **Landing Page:** Clicking the ad leads to a dedicated "Home Cinema Bundles" page or the TV's product page with prominent "Frequently Bought Together" recommendations.

Mini Example

A user browses a 'Samsung 65" 4K Smart TV'. MBA finds a 'Bose Soundbar 900' and 'AudioQuest HDMI Cable' are commonly bought together. A display ad is shown to the user on a third-party website, showcasing the 'Samsung TV' with the 'Bose Soundbar' and 'HDMI Cable' in an attractive layout, with text like "The Ultimate Entertainment Duo Awaits!"

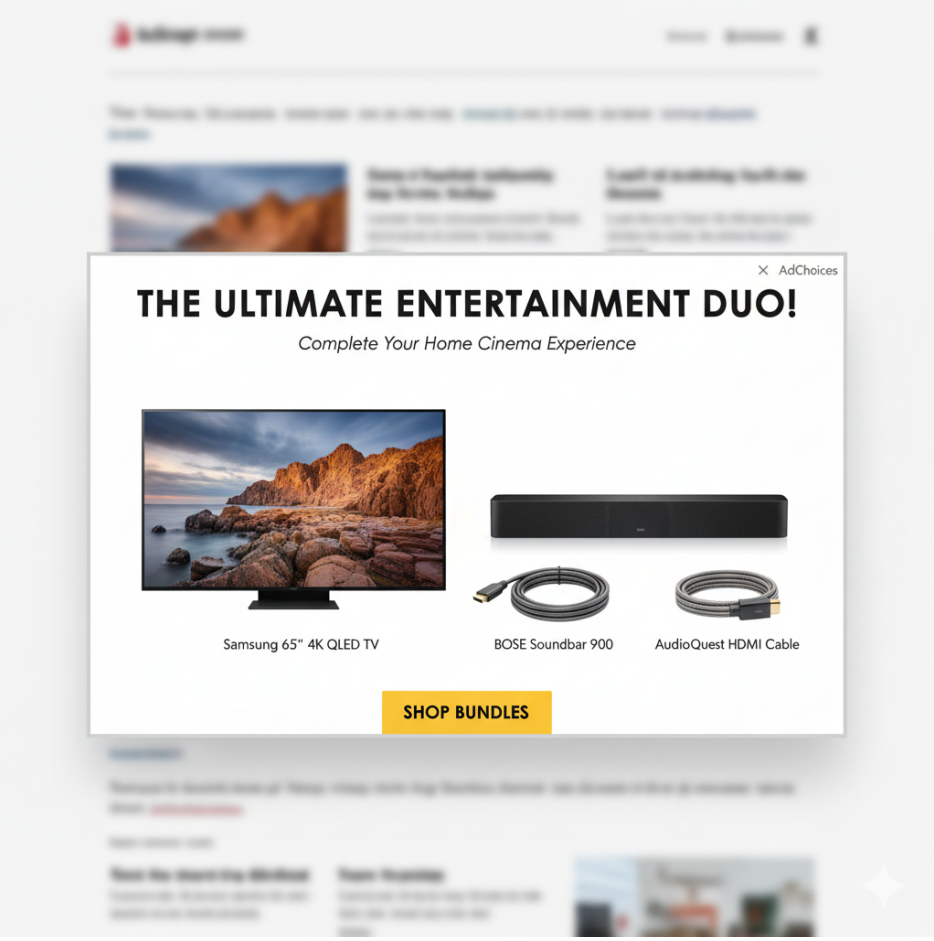

Figure 2: A banner ad demonstrating digital adjacency. It targets a user interested in a TV by showing it alongside its high-lift complements: a soundbar and HDMI cable.

- **Pre-built Bundles:** This involves packaging frequently co-purchased items into a single, convenient, and often slightly discounted product. The goal is to increase the average transaction value by making it easy for the customer to buy the complete "solution." A great example is a "New Parent Kit" containing diapers, wipes, and baby formula, or a "Summer Grilling Kit" with burger buns, ketchup, mustard, and relish.

How to Implement: The Technical Workflow

Implementing this strategy involves a clear, data-driven process from raw transaction data to a new online or offline store layout.

i. **Build a Co-occurrence Graph**
Think of this as creating a social network for your products. Each item is a node, and a connection (an edge) exists between two items if they are frequently bought together. The strength of this connection is not just about frequency, but about how *unexpectedly* frequent it is. That's where metrics like **lift** and **leverage** come in.

- **Lift:** This metric tells you how much more likely a customer is to buy Item B if they have already put Item A in their basket. The formula is:

$$\text{Lift}(A \Rightarrow B) = \frac{P(A \cap B)}{P(A)P(B)}$$

A **lift value > 1** indicates a positive association. For example, if $\text{Lift}(\text{Beer} \Rightarrow \text{Diapers}) = 3.5$, it means customers who buy beer are 3.5 times more likely to also buy diapers than the average customer is. This is a strong signal for co-promotion.

- **Leverage:** This metric measures the difference between how often two items appear together in transactions and how often you'd expect them to appear together if they were independent. The formula is:

$$\text{Leverage}(A \Rightarrow B) = P(A \cap B) - P(A)P(B)$$

Positive leverage indicates that the two items appear together more often than by random chance. It's useful for finding associations that occur frequently in absolute terms.

ii. **Identify Communities (Clusters)**
Once the graph is built, you can use algorithms to find "communities"—groups of products that are all highly interconnected. These represent natural product families based on customer behavior.

- **Example:** You might uncover a "Sunday Brunch" cluster consisting of eggs, bacon, orange juice, and champagne.
- **Action:** This cluster tells you these items should be merchandised near each other. The **cluster exemplar**—the most central or popular item, like eggs—is the perfect candidate to feature on a promotional end-cap display, with the other items placed nearby to encourage the full basket purchase.

iii. **Form Bundles from Dense Subgraphs**
Within your graph, look for "dense subgraphs"—small groups where almost every item is strongly linked to every other item. These are prime candidates for bundles.

- **Example:** You identify a triad of **{Tortilla Chips** $\leftrightarrow$ **Salsa** $\leftrightarrow$ **Guacamole}** where all three have high lift with each other.
- **Action:** Create a "Fiesta Bundle" with all three items. The pricing must be **margin-aware**. If the combined margin of the three items sold separately is $4.00, you might offer the bundle with a $0.50 discount. The small discount is enough to incentivize the customer to buy all three, increasing the total transaction margin from what might have been a single item sale.

iv. **Segment and Recompute**
Customer behavior is not static. It changes with seasons, holidays, and trends.

- **Segmentation:** A store near a university will have a different co-occurrence graph than one in a suburban neighborhood. Analyze these store clusters separately.
- **Recomputation:** The "charcoal and hot dogs" association spikes in the summer. The "wrapping paper and tape" association spikes in December. You must **recompute your association rules monthly or quarterly** to ensure your merchandising reflects this natural drift in purchasing habits.

Operational Tips: From Insight to Action

- **Minimize Travel Distance:** If coffee and coffee filters have a high lift, placing them at opposite ends of the store creates unnecessary friction. Reducing the physical or digital clicks required to purchase a pair of items makes the joint purchase more likely.
- **Cap Visual Density:** An end-cap or a digital landing page should be clean and focused. Don't try to merchandise 20 different items from a cluster. Pick the top 3-5 hero items. A cluttered display confuses customers and reduces conversion.
- **A/B Test Relentlessly:** Don't roll out a new layout to all your stores at once. Use **hold-out stores** for comparison.
  - **Test Group (10 stores):** Implement a new "Game Day" end-cap featuring chips, salsa, and beer.
  - **Control Group (10 similar stores):** Leave the layout unchanged.
  - **Analysis:** After a set period, compare sales of these items, the total basket size, and category performance between the two groups. This proves whether the change had a real financial impact.

KPIs: Measuring Success

To know if your strategy is working, you need to track the right metrics.

- **Attachment Rate:** For the {chips+salsa+guac} bundle, what percentage of customers who buy the chips also buy the other two components? A successful strategy will increase this rate.
- **Category Lift vs. Control:** Did placing a new brand of craft beer next to gourmet pretzels increase sales for the *entire* craft beer category in your test stores compared to your control stores?
- **Dwell Time:** Are customers physically pausing and spending more time in front of your new displays? (This can be measured with in-store sensors or video analytics). Increased engagement is often a leading indicator of sales.
- **GMROS (Gross Margin Return on Space):** This is a critical retail KPI. It tells you the profit you're generating from the square footage dedicated to a display. A successful end-cap will have a much higher GMROS than a standard aisle shelf, justifying its use of prime real estate.

Pitfalls: What to Watch Out For

- **Overfitting to Seasonality:** If you only analyze December data, you'll conclude that eggnog and wrapping paper are a powerful year-round duo. You must use a long-term dataset or explicitly model seasonality to avoid making short-sighted decisions.
- **Creating Traffic Bottlenecks:** The famous "beer and diapers" story is a great example of a high-lift pair. However, placing a massive beer display in a narrow baby aisle could create a traffic jam, frustrating shoppers and hurting the shopping experience for everyone. Always consider store flow and aisle width.
- **Cannibalizing Premium Sales:** Be careful with bundles. If you create a bundle with a mid-range wine and a generic cheese, the discount might convince shoppers who normally buy your **high-margin premium wine** to trade down. The goal is to get the customer who buys *only* cheese to add the wine, not to get the premium wine buyer to spend less. The bundle should **increase the total basket margin**, not just shift sales to a lower-margin combination.

2 Core Theory

Transactions and Itemsets. We analyze a dataset of N transactions (baskets). Each basket is a *set* of purchased items. An *itemset* is any subset of items, e.g., $\{\text{Milk}, \text{Cheese}\}$.

Association Rules. A rule is a directed statement $X \Rightarrow Y$, where X and Y are disjoint itemsets. It summarizes a tendency: when X occurs, Y also occurs with some strength.

Key Measures (Rule Interestingness). Let $\text{count}(\cdot)$ denote the number of baskets containing an itemset, and N the total number of baskets.

$$\textbf{Support:} \quad \text{supp}(X \Rightarrow Y) = \frac{\text{count}(X \cup Y)}{N}.$$

$$\textbf{Confidence:} \quad \text{conf}(X \Rightarrow Y) = \frac{\text{count}(X \cup Y)}{\text{count}(X)} = \Pr(Y \mid X).$$

$$\textbf{Lift:} \quad \text{lift}(X \Rightarrow Y) = \frac{\text{supp}(X \cup Y)}{\text{supp}(X)\,\text{supp}(Y)} = \frac{\text{conf}(X \Rightarrow Y)}{\text{supp}(Y)}.$$

Interpretation: lift > 1 indicates positive association, lift $= 1$ independence, and lift < 1 negative association.

Market Basket Analysis: Core Measures Intuition

Let N denote the total number of transactions in the dataset. For itemsets X and Y, write X for the number of transactions that contain all items in X, and $X \cap Y$ for the number that contain all items in both X and Y.

Definition 1 (Support)**.** *Support determines how often a rule is applicable to the dataset. A rule with very low support may occur simply by chance. For the rule* $\{X\} \Rightarrow \{Y\}$*,*

$$(\{X\} \Rightarrow \{Y\}) \;=\; \frac{X \cap Y}{N}.$$

Here, X and Y are the antecedent and consequent, respectively; N is the total number of transactions. Support measures how frequently the collection of items occurs together as a fraction of all transactions (i.e., the fraction of transactions that contain both X and Y).
Support can also be computed for a single itemset (standalone). Example: *if "Milk" appears in* 6 *out of* 9 *transactions, then*

$$(\{\textit{Milk}\}) = \frac{6}{9} \approx 0.67.$$

Definition 2 (Confidence)**.** *Confidence measures the reliability of the inference made by a rule. For the rule* $\{X\} \Rightarrow \{Y\}$*, confidence is the conditional probability of Y given X:*

$$(\{X\} \Rightarrow \{Y\}) \;=\; \frac{X \cap Y}{X} \;=\; \frac{(X \cup Y)}{(X)} \;=\; \Pr(Y \mid X).$$

Intuitively, it answers: how often do items in Y appear in transactions that contain X?

Definition 3 (Lift)**.** *For the rule* $\{X\} \Rightarrow \{Y\}$*, the* lift *(or lift ratio) compares the observed co-occurrence of X and Y to what would be expected if X and Y were independent:*

$$\text{lift}(\{X\} \Rightarrow \{Y\}) \;=\; \frac{(X \cup Y)}{(X)\,(Y)} \;=\; \frac{X \cap Y\, N}{X\, Y}.$$

A lift greater than 1 *indicates that X and Y co-occur more often than expected by chance (positive association); a lift less than* 1 *indicates negative association.*

Practical note on thresholds. Analyses typically set a minimum support threshold to focus on rules that occur frequently enough to be meaningful. The choice of threshold is domain-dependent: setting it too high may discard interesting patterns; setting it too low may admit many uninteresting rules.
After finding frequent itemsets (by a minimum support), rules are generated and then filtered by minimum confidence (and often lift, conviction, or leverage) to retain business-meaningful patterns.

Practical Cautions. Thresholds that are too high miss insights; too low produce noise. Validate operational impact (e.g., A/B tests) and align with business objectives (margin, availability, cannibalization).

3 Worked Example — "How Market Basket Works"

Consider 9 baskets over four items: Milk (M), Cheese (C), Apples (A), Banana (B).

Baskets:

$$1)\ \{M,C\}\quad 2)\ \{M,A,C\}\quad 3)\ \{A,B\}\quad 4)\ \{M,C\}\quad 5)\ \{A,B\}$$
$$6)\ \{M,C,B\}\quad 7)\ \{M,C\}\quad 8)\ \{C,B\}\quad 9)\ \{C,M\}$$

Marginal supports:

$$\text{supp}(M)=\tfrac{6}{9}=0.667,\quad \text{supp}(C)=\tfrac{7}{9}=0.778,\quad \text{supp}(A)=\tfrac{3}{9}=0.333,\quad \text{supp}(B)=\tfrac{4}{9}=0.444.$$

Example Rules and Calculations

Below, $\text{count}(X \cup Y)$ is the number of baskets containing all items in X and Y.

Rule $X \Rightarrow Y$	**count**$(X \cup Y)$	**count**(X)	**Support**	**Confidence**	**Lift**
$M \Rightarrow C$	6	6	$\frac{6}{9}=0.667$	$\frac{6}{6}=1.000$	$\frac{1.000}{0.778}=1.286$
$A \Rightarrow M$	1	3	$\frac{1}{9}=0.111$	$\frac{1}{3}=0.333$	$\frac{0.333}{0.667}=0.500$
$A \Rightarrow B$	2	3	$\frac{2}{9}=0.222$	$\frac{2}{3}=0.667$	$\frac{0.667}{0.444}=1.500$
$\{C,M\} \Rightarrow B$	1	6	$\frac{1}{9}=0.111$	$\frac{1}{6}=0.167$	$\frac{0.167}{0.444}=0.375$
$\{M,A\} \Rightarrow C$	1	1	$\frac{1}{9}=0.111$	$\frac{1}{1}=1.000$	$\frac{1.000}{0.778}=1.286$

Table 1: Support, confidence, and lift for selected rules in the mini dataset.

Interpretation.

- $M \Rightarrow C$ and $\{M,A\} \Rightarrow C$ both have lift ≈ 1.29: strong positive linkage to *Cheese* when *Milk* (and *Apples*) are present.
- $A \Rightarrow M$ has lift 0.5: in this sample, seeing *Apples decreases* the likelihood of *Milk* relative to baseline.
- $A \Rightarrow B$ has lift 1.5: a classic complementary relationship.
- $\{C,M\} \Rightarrow B$ has lift 0.375: customers buying *Milk+Cheese* are less likely than average to add *Bananas*.

Apriori Algorithm: Why Apriori instead of “just count everything”?

The **Apriori algorithm** is a clever way to find items that are frequently purchased together in a dataset of transactions. Think of it as the engine behind "Customers who bought this also bought..." recommendations you see on e-commerce sites.

Its main goal is to perform **market basket analysis** by discovering "association rules," like "If a customer buys bread, they are 80% likely to also buy milk."

If your catalog has 100 items, the number of possible 3-item combos is $\binom{100}{3} = 161{,}700$; 4-item combos are 3,921,225. With 1000 items, it’s billions to trillions. Brute-force counting all subsets of all baskets is infeasible.

Apriori finds *frequent itemsets* using a bottom-up (level-wise) strategy. It starts with single items that meet a minimum occurrence threshold, then grows candidates by adding one item at a time, pruning any candidate whose *subsets* were not frequent (the Apriori property). The process stops when no new frequent itemsets can be formed. More detailed explanation is provided below.

The Core Idea: The Apriori Principle

The algorithm is built on one simple, common-sense idea:

> **If a set of items is frequently purchased together, then any smaller subset of those items must *also* be frequently purchased.**

For example, if {Bread, Milk, Eggs} is a popular combination, then {Bread, Milk} must also be popular.
The reverse is even more important for how the algorithm works:

> **If an item or a small set of items is *unpopular*, then any larger set containing it will also be unpopular.**

This is the key to its efficiency. If very few people buy "anchovies," the algorithm doesn’t waste time checking for combinations like “{anchovies, peanut butter, milk}”. It *prunes* them away early.

How It Works: The Steps

Imagine you have a long list of shopping receipts. The algorithm works step-by-step to find the popular combinations.

1. **Set a Minimum Threshold (Support)**
   First, you decide what "frequent" means. For example, you might say, "I only care about item combinations that appear in at least 10% of all transactions." This percentage is called the **minimum support**.

2. **Find All Frequent *Single* Items**
   The algorithm scans all the receipts and counts how many times each individual item (like "Milk," "Bread," "Eggs") appears. It keeps only the items that meet your minimum support threshold and discards the rest.

3. **Generate and Test Pairs**
   Using the list of frequent single items from Step 2, it creates all possible pairs (e.g., {Milk, Bread}, {Milk, Eggs}). It then scans the receipts again to count how many times each *pair* appears. Any pair that doesn't meet the minimum support is discarded.

4. **Generate and Test Triplets (and so on...)**
   Now, it takes the frequent *pairs* and combines them to create triplets (e.g., {Milk, Bread, Eggs}). Again, it scans the receipts, counts the triplets, and discards any that are not frequent enough. This process repeats—creating bigger and bigger sets (quadruplets, etc.) from the frequent sets found in the previous step—until it can't find any more frequent combinations.

5. **Create Association Rules**
   From the final list of frequent itemsets (like {Milk, Bread}), the algorithm generates strong association rules. To do this, it calculates a metric called **confidence**.

   - **Confidence** measures the likelihood of buying item B if you've already bought item A.
   - For the rule Bread $\rightarrow$ Milk, the confidence would be:

$$\frac{\text{Number of transactions with } \{\text{Bread, Milk}\}}{\text{Total number of transactions with Bread}}$$

   If the confidence is high (e.g., 80%), you've found a useful rule!

Mini example (min occurrence threshold = 3). Orders:

- order 1: `banana`, `bread`, `juice`
- order 2: `cheese`, `juice`
- order 3: `banana`, `bread`, `cheese`
- order 4: `banana`, `bread`
- order 5: `banana`, `cheese`

Iteration 1 (size-1 candidates): count each item.

itemset	occurrence count
{`banana`}	4.00
{`bread`}	3.00
{`juice`}	2.00
{`cheese`}	3.00

Since {`juice`} occurs only 2 times < 3, it is *not* frequent and is pruned.

Iteration 2 (size-2 candidates): build pairs from the remaining frequent singletons (`banana`, `bread`, `cheese`) and count.

itemset	occurrence count
{banana, bread}	3.00
{banana, cheese}	2.00
{bread, cheese}	1.00

Only {banana, bread} meets the threshold (3). No size-3 candidates exist that can meet the threshold given these counts, so the algorithm stops. (For pairwise relationships, reaching size 2 is sufficient.)

Association Rules Mining

With frequent itemsets in hand, we form rules of the form $\{\mathtt{A}\} \to \{\mathtt{B}\}$ (here, pairs yield two directional rules). These are common in recommender systems: "customers who bought A also bought B."

Support

Support is the fraction of orders that contain the itemset:

$$\text{support}\big(\{\text{banana}, \text{bread}\}\big) = \frac{3}{5} = 60\%.$$

In large grocery datasets (many SKUs, small baskets), a low minimum support (e.g., 0.01%) can be appropriate, but it should reflect domain needs.

Confidence

Confidence for $\{\mathtt{A}\} \to \{\mathtt{B}\}$ is the conditional probability of B given A:

$$\text{confidence}\{\text{A} \to \text{B}\} = \frac{\text{support}\{\text{A}, \text{B}\}}{\text{support}\{\text{A}\}} = \Pr(B \mid A).$$

Direction matters:

$$\text{confidence}\{\text{B} \to \text{A}\} = \frac{\text{support}\{\text{A}, \text{B}\}}{\text{support}\{\text{B}\}}.$$

Example (from the mini data).

$$\text{confidence}\{\text{banana} \to \text{bread}\} = \frac{\text{support}\{\text{banana}, \text{bread}\}}{\text{support}\{\text{banana}\}} = \frac{3/5}{4/5} = 0.75\ (75\%),$$

$$\text{confidence}\{\text{bread} \to \text{banana}\} = \frac{\text{support}\{\text{banana}, \text{bread}\}}{\text{support}\{\text{bread}\}} = \frac{3/5}{3/5} = 1.00\ (100\%).$$

Lift

Lift tests whether co-occurrence is above (or below) chance:

$$\text{lift}\{\text{A}, \text{B}\} = \frac{\text{support}\{\text{A}, \text{B}\}}{\text{support}\{\text{A}\} \cdot \text{support}\{\text{B}\}} = \text{lift}\{\text{B}, \text{A}\}.$$

Example.

$$\text{lift}\{\text{banana}, \text{bread}\} = \frac{3/5}{(4/5) \cdot (3/5)} = \frac{0.6}{0.48} = 1.25.$$

Interpretation:

- lift $= 1$: no association (co-occur as often as random).
- lift > 1: positive association (co-occur more than random).
- lift < 1: negative association (co-occur less than random).

Here, `banana` and `bread` appear together $1.25\times$ more than random, indicating a positive relationship.

Armed with Apriori and association rules, you can now analyze transactions to uncover meaningful pairs and drive recommendations, promotions, and layout decisions.

4 Practical Applications of Market Basket Analysis

When one hears Market Basket Analysis, one thinks of shopping carts and supermarket shoppers. It is now increasingly used to analyze online behavior of users.

It is important to realize that there are many other areas in which Market Basket Analysis can be applied. An example of Market Basket Analysis is a list of potentially interesting products that one sees as Amazon's or Netflix's recommendations. For example, Amazon informs the customer who bought the item being purchased about other products bought by customers who purchased the same item. A list of applications of Market Basket Analysis in various industries is listed below:

- **Online services.** Product recommendations, online advertising based on click-through data analysis, user behavior on online content.
- **Retail.** In retail, Market Basket Analysis can help determine what items are purchased together, purchased sequentially, and purchased by season. This can assist retailers in product placement decisions on shelves and promotion optimization (for instance, combining product incentives). Does it make sense to sell (or promote) soda and chips or soda and crackers?
- **Failure analysis and prediction.** By building profiles of past failures (related to operations, machinery, safety, etc.), one can develop probabilistic scenarios of specific failure situations.
- **Banks.** In financial services (banking, for instance), Market Basket Analysis can be used to analyze credit card purchases of customers to build profiles for fraud detection purposes and cross-selling opportunities.
- **Insurance.** In insurance, Market Basket Analysis can be used to build profiles to detect medical insurance claim fraud. By building profiles of claims, you can then determine if more than one claim belongs to a particular claimant within a specified period of time.
- **Medical.** In healthcare or medical settings, Market Basket Analysis can be used for comorbid conditions and symptom analysis, with which a profile of illness can be better identified. It can also be used to reveal biologically relevant associations between different genes or between environmental effects and gene expression.

5 In Class Activity: Market Basket Analysis with Online Retail Data

Objective: This in-class activity will guide you through a practical application of Market Basket Analysis (MBA) using the Apriori algorithm. You will clean and prepare a real-world transaction dataset, mine for insightful association rules, and translate those findings into actionable business recommendations.

Learning Outcomes:

- Prepare and transform raw transactional data into a suitable format for association rule mining.
- Apply the Apriori algorithm to discover frequent itemsets and generate meaningful association rules.
- Interpret and evaluate association rules using key metrics like **support**, **confidence**, and **lift**.
- Visualize rule relationships using scatter plots and interactive graphs.
- Formulate data-driven business recommendations based on analytical findings.

---

1. Setup and Initial Exploration

Your first step is to set up your R environment and perform an initial data load to ensure everything is working correctly.

Tasks:

1. **Open Template:** Open the provided R Markdown template, `apriori_online_retail.Rmd`.
2. **Install & Load Packages:** Run the first code chunk to ensure all necessary packages are installed and loaded.
3. **Initial Knit:** Generate your first HTML report by knitting the document with the default parameters. This confirms that your environment is correctly configured. You can do this using the RStudio IDE or by running the following command in your console:

```
rmarkdown::render(
  "apriori_online_retail.Rmd",
  params = list(
    data_path = "online_retail_1000.csv",
    country   = "United Kingdom",
    min_supp  = 0.01,
    min_conf  = 0.5,
    maxlen    = 4
  )
)
```

---

2. Part A: Data Preparation and Validation

Objective: To clean the raw transactional data, transform it into a "basket" format, and perform an initial exploratory analysis of item frequencies.
The dataset is a 1,000-transaction sample from the "Online Retail" dataset. Each row represents a single line item on an invoice. Before we can analyze buying patterns, we must aggregate these items into distinct shopping baskets.

Key Variables:

Variable	Description
`InvoiceNo`	The unique identifier for each transaction or receipt.
`StockCode`	The internal code for each product.
`Description`	The text description of the product.
`Quantity`	The number of units of a product on a line item.
`UnitPrice`	The price for a single unit of the product.
`Country`	The country where the customer resides.

Tasks:

1. **Filter Invalid Transactions:** Clean the dataset by removing all canceled orders and any line items with non-positive `Quantity` or `UnitPrice`.
2. **Isolate Target Country:** Filter the data to include transactions only from the country specified in the `params$country` variable.
3. **Aggregate into Baskets:** Transform the cleaned data into a `transactions` object suitable for the `arules` package.
4. **Analyze Item Frequency:** Generate and display a relative frequency plot for the top 20 most purchased items.

Analysis Questions:

Based on the item frequency plot, answer the following in a few bullet points within your R Markdown document:

- Which 2–3 items appear surprisingly frequently? Are these core products or accessories?
- Do you notice any data quality issues or unusual items in the top 20, such as generic codes (e.g., "POSTAGE"), gift-wrapping services, or other non-product entries?

---

3. Part B: Mining and Visualizing Association Rules

Objective: To apply the Apriori algorithm to discover association rules and use visualizations to identify interesting patterns and rule clusters.

Tasks:

1. **Run Apriori:** Execute the Apriori algorithm using the parameters defined in the document.
2. **Filter for Strong Rules:** Refine your rule set by keeping only the rules with a **lift greater than 1.2**.
3. **Inspect and Export:** Sort the filtered rules by lift and inspect the top 20. Export the complete set to a CSV file named `rules_export_from_synth_csv.csv`.
4. **Visualize Rules:** Generate a scatter plot of support vs. confidence (shaded by lift) and an interactive graph plot of the top 50 rules.

Analysis Questions:

Examine the interactive graph plot. Identify two distinct "clusters" of rules.

- For each cluster, provide a descriptive name (e.g., "Afternoon Tea Accessories").
- List two representative items from each cluster that exemplify the theme.
- Provide your written descriptions on the R Markdown file itself.

---

4. Part C: Interpretation and Business Recommendations

Objective: To translate the discovered association rules into concrete, testable business strategies. Provide answers in the R Markdown document itself.

Tasks:

1. **Select High-Impact Rules:** From your filtered list, choose three commercially interesting rules that balance high lift and confidence with reasonable support.
2. **Formulate Recommendations:** For each rule, write a concise, actionable recommendation (2–4 sentences) that includes:
   - **The Action:** What specific business action should be taken?
   - **The Rationale:** Why is this action justified? Reference the rule's support, confidence, and lift.
   - **The Test:** How would you measure the success of this action?

Sensitivity Analysis

If you finish early, investigate how changing the Apriori parameters affects the results. Re-run your analysis using one of the following scenarios:

- **Scenario A (Stricter):** `min_supp = 0.02`, `min_conf = 0.6`

- **Scenario B (Looser):** `min_supp = 0.005`, `min_conf = 0.4`

In your R Markdown document, add a brief summary noting the changes in the total number of rules found and the median values for support, confidence, and lift.

---

5. Submission

By the end of the class, please upload the following two files to the LMS:

1. `apriori_online_retail.html`: Your final knitted report, including all code, plots, and written answers.
2. `rules_export_from_synth_csv.csv`: The CSV file containing the filtered, high-lift association rules.

---

6 In Class Activity: GMROS Analysis

How to go from raw transactions to *GMROS-aware* display decisions:

1. Build clean inputs (catalog, transactions).
2. **Mine rule stats $X \to Y$ (with a simple explanation)**.
3. Compute baseline GMROS by item.
4. Convert rule strength into incremental units, margin, and GMROS for a display.
5. Pick the best display candidates and rank them.
6. Validate, stress-test, and adapt to constraints.

Data and output used in this demo (CSV).

- catalog.csv (price, cost, shelf space, unit margin)
- transactions_long.csv; transactions_wide.csv (one-hot)
- item_baseline_gmros.csv
- rules_scored_endcap.csv
- top_targets_summary.csv
- calc_demo_top3.csv (worked math)

Step 1 — Build the inputs

1A. Catalog (profit + space)

For each item i, record:

$$\text{unit\_margin}(i) = \text{price}(i) - \text{cost}(i), \qquad \text{shelf\_space}(i) \ \ (\text{sq ft}).$$

1B. Transactions (basket × item one-hot)

Create a one-hot matrix $M \in \{0,1\}^{N \times I}$:

$$M_{t,i} = \begin{cases} 1 & \text{if basket } t \text{ contains item } i, \\ 0 & \text{otherwise.} \end{cases}$$

Here N is the number of baskets, I the number of items.

Step 2 — Mine rule stats $X \to Y$

What you start with

A table M: *rows = baskets*, *columns = items*, and $N =$ of baskets.

Count the basics

- **count**(X): how many baskets contain X *(sum the X column: # of 1s in column X)*.
- **count**(Y): how many baskets contain Y.
- **count**$(X \wedge Y)$: how many baskets contain *both* X and Y *(rows where $X = 1$ and $Y = 1$; equivalently, the dot product of columns X and Y)*.

Turn counts into probabilities (supports)

$$s(X) = \frac{\text{count}(X)}{N}, \qquad s(Y) = \frac{\text{count}(Y)}{N}, \qquad s(X,Y) = \frac{\text{count}(X \wedge Y)}{N}.$$

Rule quality for $X \to Y$

$$\textbf{Confidence:} \quad \text{conf}(X \to Y) = \frac{\text{count}(X \wedge Y)}{\text{count}(X)} = \frac{s(X,Y)}{s(X)} = \Pr(Y \mid X).$$

$$\textbf{Lift:} \quad \text{lift}(X \to Y) = \frac{\text{conf}(X \to Y)}{s(Y)}.$$

Interpretation: lift > 1 means X and Y co-occur more than chance (positive association).

Tiny numeric example:

If $N = 100$, $\text{count}(X) = 30$, $\text{count}(Y) = 40$, $\text{count}(X \wedge Y) = 18$:

$$s(X) = 0.30, \quad \text{conf}(X \to Y) = \tfrac{18}{30} = 0.60, \quad \text{lift} = \tfrac{0.60}{0.40} = 1.5.$$

Step 3 — Baseline GMROS (by item)

GMROS (Gross Margin Return on Space) on the regular shelf:

$$\text{GMROS}_{\text{baseline}}(i) = \frac{\text{units}(i) \times \text{unit\_margin}(i)}{\text{shelf\_space}(i)} \quad (\$/\text{sq ft}).$$

This answers: *How many dollars of gross margin do we earn per square foot of space from item i today?*

Step 4 — From rule strength to display GMROS

We want the incremental benefit of displaying Y near X.

4A. Incremental propensity

$$\Delta p = \Pr(Y \mid X) - \Pr(Y) = \text{conf}(X \to Y) - s(Y) \quad (\geq 0 \text{ when helpful}).$$

4B. Incremental units for Y

For N baskets and $s(X) = \Pr(X)$:

$$\text{incr\_units}(Y; X) = N \cdot s(X) \cdot \Delta p.$$

1) Baseline vs. Conditional Probability

Let $s(\cdot)$ denote support and $\Pr(\cdot)$ probability. For a rule $X \to Y$:

$$\Pr(Y) = s(Y) \qquad \text{and} \qquad \Pr(Y \mid X) = \text{confidence}(X \to Y).$$

Define the *incremental propensity* (per X-basket) as

$$\Delta p = \Pr(Y \mid X) - \Pr(Y) = \text{conf}(X \to Y) - s(Y) \ \ (\geq 0 \text{ when helpful}).$$

Equivalent via lift: since $\text{lift}(X \to Y) = \dfrac{\Pr(Y \mid X)}{\Pr(Y)}$,

$$\Delta p \ = \ \Pr(Y)\,\big(\text{lift}(X \to Y) - 1\big) \ = \ s(Y)\,\big(\text{lift} - 1\big).$$

2) How Many Baskets Are Affected?

Only baskets containing X are "exposed" to the co-placement:

$$\#\text{ of } X\text{-baskets} \ = \ N \cdot s(X) \ = \ \text{count}(X).$$

3) Expected Incremental Units of Y

Each X-basket adds Y with extra probability Δp. Across all X-baskets:

$$\text{incr\_units}(Y; X) = \underbrace{N\,s(X)}_{\#\ X\text{-baskets}} \cdot \underbrace{\Delta p}_{\text{extra } Y \text{ per such basket}} = N\,s(X)\,\big[\Pr(Y \mid X) - \Pr(Y)\big].$$

Useful alternate forms:

$$\text{incr\_units}(Y; X) = \text{count}(X)\,\Delta p = N\,s(X)\,s(Y)\,\big(\text{lift} - 1\big).$$

4) Tiny Numeric Example

Suppose $N = 600$ and for the rule *Beer* $\to$ *Diapers* we have

$s(X) = \Pr(\text{Beer}) = 0.1733, \quad s(Y) = \Pr(\text{Diapers}) = 0.1717, \quad \text{conf}(X \to Y) = \Pr(Y \mid X) = 0.4808.$

Then

$\Delta p = 0.4808 - 0.1717 = 0.3091, \qquad \text{incr\_units} = 600 \times 0.1733 \times 0.3091 \approx 32.15$ extra units of Diapers.

4C. Convert to margin and GMROS (display)

The goal is to convert expected incremental units into dollars per square foot.

1) From incremental units to gross margin (no uplift). Given the expected incremental units of Y from co-placement with X, denoted $\text{incr\_units}(Y; X)$, and the unit margin of Y,

$$\text{incr\_margin}_0(Y; X) = \text{incr\_units}(Y; X) \times \text{unit\_margin}(Y).$$

2) Account for "prime real estate" visibility. If a display (e.g., an end-cap) yields a visibility multiplier m (e.g., $m = 3$),

$$\text{incr\_margin}(Y; X) = \text{incr\_units}(Y; X) \times \text{unit\_margin}(Y) \times m.$$

3) Normalize by space to get GMROS. Let s be the square footage allocated to the display for Y. GMROS (Gross Margin Return on Space) is

$$\text{GMROS}_{\text{display}}(Y; X) = \frac{\text{incr\_margin}(Y; X)}{s \text{ sq ft}}.$$

In the special case of a 1 sq ft end-cap ($s = 1$) with $m = 3$,

$$\text{incr\_margin}(Y; X) = \text{incr\_units} \times \text{unit\_margin}(Y) \times 3, \qquad \text{GMROS}_{\text{endcap}}(Y; X) = \frac{\text{incr\_margin}}{1 \text{ sq ft}}.$$

Mini example (numbers from the numerical example). For the rule *Beer* $\rightarrow$ *Diapers* we had $\text{incr\_units} \approx 32.15$, $\text{unit\_margin}(\text{Diapers}) = \$4.99$. With $m = 3$ and $s = 1$:

$$\text{incr\_margin} \approx 32.15 \times 4.99 \times 3 \approx \$481.24, \qquad \text{GMROS}_{\text{endcap}} \approx \$481.24/\text{sq ft}.$$

Sensitivity (e.g., $m = 2$, $s = 2$):

$$\text{incr\_margin} \approx 32.15 \times 4.99 \times 2 \approx \$320.90, \qquad \text{GMROS} \approx 320.90/2 \approx \$160.45/\text{sq ft}.$$

Step 5 — Pick candidates and rank

For each *consequent* Y:

1. Consider all rules $X \rightarrow Y$ that pass practical thresholds (e.g., support $\geq 6\%$, confidence $\geq 30\%$).
2. Keep the X that *maximizes* $\text{GMROS}_{\text{endcap}}(Y; X)$.
3. Join Y's $\text{GMROS}_{\text{baseline}}$ to compare with the display case.
4. Rank by $\text{GMROS}_{\text{endcap}}$ (desc) and apply store constraints (space, adjacencies, inventory).

Step 6 — Validate and stress-test

- **Sensitivity:** vary the visibility multiplier (e.g., 1.5×–3×) and space per SKU (1–4 sq ft).
- **Hold-out check:** validate that $X \rightarrow Y$ truly raises Y vs. baseline in a test period/zone.
- **Hurdle rates:** require $\text{GMROS}_{\text{endcap}}$ to exceed a category-specific threshold.

Worked rows (from this demo)

Top three rules from the run and their GMROS arithmetic
(see calc_demo_top3.csv *forallnumbers*).

Rule $X \Rightarrow Y$	$s(X)$	$s(Y)$	$s(X \wedge Y)$	Confidence	Lift
Beer→Diapers	0.17	0.17	0.08	0.48	2.80
Cereal→Milk	0.36	0.36	0.23	0.65	1.81
Milk→Cereal	0.36	0.36	0.23	0.65	1.81

Rule $X \Rightarrow Y$	Δp	Incr. units	Unitmargin ($)	GMROSendcap	GMROSbase	ΔGMROS
Beer→Diapers	0.31	32.15	4.99	481.24	171.32	309.92
Cereal→Milk	0.29	62.60	2.39	448.87	259.32	189.55
Milk→Cereal	0.29	62.60	2.29	430.08	245.03	185.05

Reading this: Beer→Diapers has strong lift and Diapers has a high unit margin; on a 1 sq ft end-cap with a 3× visibility bump, the *incremental* GMROS is $481.24/sq ft—well above Diapers' baseline $171.32/sq ft.

7 Market Basket Analytics Assignments

1) Twitter Market Basket Analysis

Task. Reproduce the Python code and run the code to generate all the plots in the paper (Link: Download the file. You may use the GitHub account of the paper: `https://github.com/kruser1/twitter-apyori` for code that could be used to begin with. Bonus points will be given for code that is substantially different (for example, by using different libraries, etc.) from the code published with the paper.

Deliverables.

(a) Using the data employed by the paper, identify rules with lift values greater than 1.0 in the Twitter accounts following Biden and Trump respectively. Save the rules in a CSV file that shows for each rule: the antecedent(s) and consequent(s), and lift.

(b) If you were to increase campaign visibility for one of the presidential candidates, say, Donald Trump, identify three rules with two antecedents and one consequent that produce the highest lift. Do the same for the candidate Kamala Harris.

(c) Identify two strategies for Twitter (X.com) that you would recommend to the campaign teams of the two presidential candidates based on the information from Question 3. Be as specific as possible and provide a clear rationale supported by your analysis for each strategy. Assume that you are explaining your strategy to an audience unfamiliar with the jargon or method used in this analysis.

(d) Upload the Python code (.py only) and CSV files. A Word file should be included to describe answers to questions 2 to 4.

2) Instacart Analysis

Instacart, an online grocer, has graciously made some of their datasets accessible to the public. The order and product datasets that we will be using can be downloaded from the link below, along with the data dictionary:

> "The Instacart Online Grocery Shopping Dataset 2017", Accessed from https://www.instacart.com/datasets/grocery-shopping-2017 on September 1, 2017.

3) Market Basket Analysis for YouTube Video Suggestions

Introduction

This assignment focuses on applying Market Basket Analysis concepts to optimize the layout of a YouTube banner featuring six video suggestions. The goal is to maximize the likelihood of users clicking on combinations of videos, thereby increasing overall engagement, by intelligently arranging the suggestions based on their associative strength.

Dataset

You are provided with a dataset (*video-mba-Fall2024.csv*) representing user viewing behavior on a YouTube channel. Each "transaction" in this dataset corresponds to a single user session where they watched at least one of the six videos (`Video-1` to `Video-6`) available in the suggestion pool. A transaction might look like:

- `{Video-1, Video-3}`
- `{Video-2, Video-5, Video-6}`
- `{Video-1}`
- `{Video-3, Video-4}`

(Note: You will be provided with this data for analysis.)

Task

Your primary task is to apply Market Basket Analysis, specifically focusing on the **lift** metric, to optimize the arrangement of six video suggestions on a YouTube banner.

1. **Data Preprocessing:**
   - Load and understand the provided dataset of video viewing sessions.
2. **Apriori Algorithm Application: Implement on Python or R:**
   - Implement or use a library (e.g., `mlxtend`) to apply the Apriori algorithm to discover frequent itemsets (combinations of videos watched together).
   - Calculate the lift for one antecedent and one consequent item-sets.
   - Calculate the lift for two antecedents and one consequent item-sets.
   - Calculate **support**, **confidence**, and critically, **lift** for the above item-sets.

3. **Analyze Lift for Banner Optimization: Implement on Excel**

   - The **Lift** metric $\text{Lift}(A \Rightarrow B) = \dfrac{\text{conf}(A \Rightarrow B)}{\text{supp}(B)}$ is key for this assignment.
     - A **Lift** > 1 indicates that items A and B are more likely to be viewed together than expected by chance, suggesting a positive association.
     - A **Lift** $= 1$ indicates no association.
     - A **Lift** < 1 indicates a negative association (less likely to be viewed together).
   - Your goal is to arrange the six videos (`Video-1` to `Video-6`) on the banner (as depicted in Figure: 3, with 3 videos on top and 3 on the bottom) such that the **Click-Through Rate (CTR) is maximized due to strong positive associations (high lift values)** between adjacent or visually grouped videos.The optimal arrangement of the videos will likely be different from the one that is shown in Figure: 3.

Figure 3: Example YouTube Banner with Six Video Suggestions (Video-1 to Video-6)

   - To do this, analyze the lift values between different pairs or groups of videos. Consider how placing videos with high lift values next to each other (horizontally or vertically) might encourage users to click on subsequent videos if they've already watched one.
   - Propose an optimal layout for the six videos on the banner based on your Market Basket Analysis (MBA) findings, explicitly justifying your arrangement with relevant lift scores.

4. **Proposed Banner Layout and CTR Rationale: Implement on Excel**

   You may place the six videos such that horizontally and vertically adjacent tiles have the highest mutual lifts.

   Intuition:

   - If a viewer clicks one video, high $L_{ij} > 1$ adjacent tiles are those they are *more likely than chance* to watch next.

- Summing lifts over all neighbor pairs concentrates the strongest associations where the eye travels first (row-wise and between aligned columns).

This arrangement is expected to increase multi-click probability relative to any arrangement that ignores pairwise association strength.A naive layout (e.g., random order or sorting purely by popularity/support) can separate complementary videos and place negatively associated items together, suppressing follow-on clicks. An ideal layout should explicitly:

(a) **Groups high-lift pairs** side-by-side (left–right and top–bottom), raising the chance of a second click after the first.

(b) **Avoids low/negative-lift adjacencies**, reducing dead-ends where interest in one video predicts disinterest in its neighbors.

5. **Potential Limitations of your approach: Hints**

   (a) *Position bias unmodeled.* Users over-click the upper-left and center.

   (b) *Correlation $\neq$ causation.* Lift captures co-viewing, not causal influence; confounders (trending topics, upload time) can inflate lifts.

   (c) *Stationarity.* Lifts are treated as static, though interests drift over time.

   (d) *Cold/sparse pairs.* Rare pairs produce noisy lift estimates.

6. **Potential improvements for your approach: Hints**

   (a) *Weighted objective with position bias.* Consider weighting the lift values by position of the video on the banner. Such weights can be calculated from historical CTR heatmaps.

   (b) *Temporal decay / recency.* Estimate lift_t with exponential decay so recent behavior dominates; re-optimize layout on a rolling schedule.

   (c) *Personalization.* Compute lifts per segment (e.g., geography, device, new vs. returning) and render segment-specific banners.

   (d) *Calibration via online testing.* A/B test versus naive/popularity baselines; update lifts with doubly robust or causal bandit methods to reduce bias.

Report and Recommendations:

- Document your methodology.
- Clearly illustrate your proposed banner layout and explain how it leverages the lift concept to maximize potential CTR. Describe how your layout increases engagement the maximum possible fashion compared to a naive layout.
- Discuss **two additional imitations** of your analysis and suggest at **least two improvements** other than suggested in the above hints.

Submission

Your submission should include:

- A well-structured report (Word/PDF) detailing your analysis, findings, and recommendations.
- All Excel files/ code used for data processing and MBA.