**Part 1: Data Cube Operations**

A **data cube** is a multidimensional data structure used in online analytical processing (OLAP) to store and analyze large amounts of data. In this case, the cube has three dimensions: **Course**, **Student**, and **Time**, with the measure being **aggregate marks**.

**i) Roll up**

**Roll up** is an aggregation operation that summarizes data by climbing up a hierarchy or reducing dimensions. For example, rolling up on the **Time** dimension could change the view from **monthly** to **quarterly** or **yearly marks**. Rolling up on the **Course** dimension could aggregate marks from individual courses to **department-level** averages.

**ii) Drill down**

**Drill down** is the reverse of roll up. It provides more detail by navigating down a hierarchy or adding dimensions. For example, drilling down on **Student** could show individual marks for a student for each course rather than their overall average. Drilling down on the **Time** dimension could break down a yearly mark into **quarterly** or **monthly** marks.

**iii) Slice**

**Slice** is a selection operation that creates a sub-cube by selecting a single value for one of the dimensions. For example, slicing the cube for a specific **Course** like "Data Mining" would create a 2D table showing the marks for all students at all times for only that course.

**iv) Dice**

**Dice** is a selection operation that creates a sub-cube by selecting multiple values for multiple dimensions. For example, dicing the cube could involve selecting marks for specific **Students** (e.g., student A, student B), for a specific range of **Time** (e.g., semester 1), and for specific **Courses** (e.g., Math, Physics).

**Part 2: Star vs. Snowflake Schema & Star Schema Design**

**Star vs. Snowflake Schema**

| Feature | Star Schema | Snowflake Schema |
| --- | --- | --- |
| **Structure** | A central **fact table** surrounded by a single layer of **dimension tables**. | A central **fact table** with normalized **dimension tables**, which may have their own sub-dimension tables. |
| **Normalization** | Denormalized. All attributes for a dimension are in one table. | Normalized. Dimensions are broken down into multiple tables to eliminate redundancy. |
| **Query Performance** | Generally faster due to fewer joins required for queries. | Slower due to the need for more joins to link the various dimension tables. |
| **Storage** | Uses more storage space due to denormalization and data redundancy. | Uses less storage space due to the normalized structure. |
| **Complexity** | Simpler to design and understand. | More complex to design and maintain due to multiple joins and tables. |

Export to Sheets

**Star Schema for Company Sales**

This schema links a central **Sales Fact Table** to three denormalized dimension tables: **Location**, **Item**, and **Time**.

* **Sales Fact Table:** Contains foreign keys linking to the dimension tables and the measures for analysis.
  + Location\_key (Foreign Key)
  + Item\_key (Foreign Key)
  + Time\_key (Foreign Key)
  + Sales\_amount (Measure)
  + Quantity\_sold (Measure)
  + Cost (Measure)
* **Location Dimension Table:**
  + Location\_key (Primary Key)
  + City
  + State
  + Country
* **Item Dimension Table:**
  + Item\_key (Primary Key)
  + Item\_name
  + Brand
  + Category
* **Time Dimension Table:**
  + Time\_key (Primary Key)
  + Day
  + Month
  + Quarter
  + Year

**Part 3: Data Mining as a Knowledge Discovery Process (KDD)**

Data mining is a crucial step within the broader **Knowledge Discovery in Databases (KDD)** process. The key steps involved are:

1. **Data Cleaning:** This initial step involves handling noise, missing values, and inconsistent data. It ensures the data is in a usable format for analysis.
2. **Data Integration:** Data from multiple, diverse sources is combined into a single, coherent data store, such as a data warehouse. This step addresses schema conflicts and data redundancy.
3. **Data Selection:** Relevant data for the analysis task is retrieved from the data warehouse. This can involve filtering data to focus on a specific time period or set of attributes.
4. **Data Transformation:** The selected data is converted into a format suitable for mining. This includes aggregation, normalization, and feature construction (creating new attributes).
5. **Data Mining:** This is the core step where intelligent methods are applied to extract patterns and models from the data. This is where algorithms for classification, clustering, association, and regression are used.
6. **Pattern Evaluation:** The discovered patterns are evaluated to identify interesting, meaningful, and useful knowledge. This involves using metrics like interestingness measures.
7. **Knowledge Presentation:** The final step involves presenting the discovered knowledge to the user in a clear and understandable format, often through visualizations or reports.

**Part 4: Data Preprocessing Steps**

**Data preprocessing** is a vital set of techniques used to transform raw data into a clean and understandable format. The main steps are:

1. **Data Cleaning:** This involves:
   * **Handling Missing Values:** Filling in or ignoring data points where values are absent (e.g., using mean, median, or a constant value).
   * **Smoothing Noisy Data:** Removing or reducing random error or variance in the data (e.g., using binning, regression, or clustering).
   * **Identifying Outliers:** Detecting and addressing data points that are significantly different from the rest of the data.
   * **Resolving Inconsistencies:** Correcting discrepancies in data codes or names.
2. **Data Integration:** This is the process of combining data from various sources into a unified repository. Challenges include:
   * **Schema Integration:** Matching and merging schemas from different sources.
   * **Handling Redundancy:** Identifying and removing redundant data.
   * **Detecting and Resolving Data Conflicts:** Ensuring consistent values for the same real-world entity.
3. **Data Transformation:** This step converts data into a more suitable format for mining. Key techniques include:
   * **Normalization:** Scaling data values to a specified range (e.g., [0, 1]) to prevent attributes with large ranges from dominating the analysis.
   * **Aggregation:** Summarizing data (e.g., calculating monthly sales from daily sales records).
   * **Feature Construction:** Creating new attributes from existing ones to improve the performance of the mining algorithm.
4. **Data Reduction:** This aims to obtain a reduced representation of the data set that is much smaller in volume but still produces the same or similar analytical results. Techniques include:
   * **Dimensionality Reduction:** Reducing the number of attributes or features (e.g., using PCA).
   * **Numerosity Reduction:** Reducing the number of data points (e.g., through sampling or data cube aggregation).
   * **Data Compression:** Encoding data to reduce its size.

**Part 5: K-Means Clustering**

**Initial Setup**

* **Points:** A1(2,10),A2(2,5),A3(8,4),B1(5,8),B2(7,5),B3(6,4),C1(1,2),C2(4,9).
* **Initial Cluster Centers:**
  + Cluster 1 (C1\_center): A1(2,10)
  + Cluster 2 (C2\_center): B1(5,8)
  + Cluster 3 (C3\_center): C1(1,2)

**a) Three Cluster Centers After the First Round**

**Step 1: Assign each point to the closest cluster center.**

We calculate the Euclidean distance d=(x2​−x1​)2+(y2​−y1​)2​ from each point to the three initial centers.

| Point | Dist to A1(2,10) | Dist to B1(5,8) | Dist to C1(1,2) | Assigned Cluster |
| --- | --- | --- | --- | --- |
| A1(2,10) | 0 | 32+22​=3.61 | (−1)2+(−8)2​=8.06 | **Cluster 1** |
| A2(2,5) | 02+(−5)2​=5 | 32+32​=4.24 | (−1)2+(−3)2​=3.16 | **Cluster 3** |
| A3(8,4) | 62+(−6)2​=8.49 | (−3)2+42​=5 | 72+22​=7.28 | **Cluster 2** |
| B1(5,8) | 32+(−2)2​=3.61 | 0 | (−4)2+(−6)2​=7.21 | **Cluster 2** |
| B2(7,5) | 52+(−5)2​=7.07 | (−2)2+32​=3.61 | 62+32​=6.71 | **Cluster 2** |
| B3(6,4) | 42+(−6)2​=7.21 | (−1)2+42​=4.12 | 52+22​=5.39 | **Cluster 2** |
| C1(1,2) | (−1)2+(−8)2​=8.06 | (−4)2+(−6)2​=7.21 | 0 | **Cluster 3** |
| C2(4,9) | 22+(−1)2​=2.24 | (−1)2+12​=1.41 | 32+72​=7.62 | **Cluster 2** |

Export to Sheets

* **Cluster 1:** {A1(2,10)}
* **Cluster 2:** {A3(8,4), B1(5,8), B2(7,5), B3(6,4), C2(4,9)}
* **Cluster 3:** {A2(2,5), C1(1,2)}

**Step 2: Recalculate the cluster centers (mean of all points in the cluster).**

* **New Cluster 1 Center:** (2/1,10/1)=(2,10)
* **New Cluster 2 Center:** ((8+5+7+6+4)/5,(4+8+5+4+9)/5)=(30/5,30/5)=(6,6)
* **New Cluster 3 Center:** ((2+1)/2,(5+2)/2)=(3/2,7/2)=(1.5,3.5)

The three cluster centers after the first round of execution are **(2, 10), (6, 6), and (1.5, 3.5)**.

**b) The Final Three Clusters**

**Round 2: Assign points to the new centers and recalculate.**

New Centers: C1'=(2,10), C2'=(6,6), C3'=(1.5,3.5)

| Point | Dist to C1'(2,10) | Dist to C2'(6,6) | Dist to C3'(1.5,3.5) | Assigned Cluster |
| --- | --- | --- | --- | --- |
| A1(2,10) | 0 | 42+42​=5.66 | 0.52+6.52​=6.52 | **Cluster 1** |
| A2(2,5) | 02+(−5)2​=5 | 42+12​=4.12 | 0.52+1.52​=1.58 | **Cluster 3** |
| A3(8,4) | 62+(−6)2​=8.49 | (−2)2+22​=2.83 | 6.52+0.52​=6.52 | **Cluster 2** |
| B1(5,8) | 32+(−2)2​=3.61 | 12+(−2)2​=2.24 | 3.52+4.52​=5.70 | **Cluster 2** |
| B2(7,5) | 52+(−5)2​=7.07 | (−1)2+12​=1.41 | 5.52+1.52​=5.70 | **Cluster 2** |
| B3(6,4) | 42+(−6)2​=7.21 | 02+22​=2 | 4.52+0.52​=4.53 | **Cluster 2** |
| C1(1,2) | (−1)2+(−8)2​=8.06 | (−5)2+(−4)2​=6.40 | 0.52+1.52​=1.58 | **Cluster 3** |
| C2(4,9) | 22+(−1)2​=2.24 | 22+32​=3.61 | 2.52+5.52​=6.04 | **Cluster 1** |

Export to Sheets

* **Cluster 1:** {A1(2,10), C2(4,9)}
* **Cluster 2:** {A3(8,4), B1(5,8), B2(7,5), B3(6,4)}
* **Cluster 3:** {A2(2,5), C1(1,2)}

**Recalculate centers:**

* **New Cluster 1 Center:** ((2+4)/2,(10+9)/2)=(3,9.5)
* **New Cluster 2 Center:** ((8+5+7+6)/4,(4+8+5+4)/4)=(26/4,21/4)=(6.5,5.25)
* **New Cluster 3 Center:** ((2+1)/2,(5+2)/2)=(1.5,3.5)

The centers changed. We repeat the process.

**Round 3: Assign points to the new centers and recalculate.**

New Centers: C1''=(3,9.5), C2''=(6.5,5.25), C3''=(1.5,3.5)

| Point | Dist to C1''(3,9.5) | Dist to C2''(6.5,5.25) | Dist to C3''(1.5,3.5) | Assigned Cluster |
| --- | --- | --- | --- | --- |
| A1(2,10) | (−1)2+0.52​=1.12 | 4.52+4.752​=6.54 | 0.52+6.52​=6.52 | **Cluster 1** |
| A2(2,5) | 12+4.52​=4.61 | 4.52+0.252​=4.5 | 0.52+1.52​=1.58 | **Cluster 3** |
| A3(8,4) | 52+5.52​=7.44 | (−1.5)2+1.252​=1.95 | 6.52+0.52​=6.52 | **Cluster 2** |
| B1(5,8) | 22+1.52​=2.5 | (−1.5)2+2.752​=3.14 | 3.52+4.52​=5.7 | **Cluster 1** |
| B2(7,5) | 42+4.52​=6.02 | (−0.5)2+0.252​=0.56 | 5.52+1.52​=5.7 | **Cluster 2** |
| B3(6,4) | 32+5.52​=6.26 | (−0.5)2+1.252​=1.35 | 4.52+0.52​=4.53 | **Cluster 2** |
| C1(1,2) | 22+7.52​=7.76 | 5.52+3.252​=6.38 | (−0.5)2+(−1.5)2​=1.58 | **Cluster 3** |
| C2(4,9) | 12+0.52​=1.12 | 2.52+3.752​=4.51 | 2.52+5.52​=6.04 | **Cluster 1** |

Export to Sheets

* **Cluster 1:** {A1(2,10), B1(5,8), C2(4,9)}
* **Cluster 2:** {A3(8,4), B2(7,5), B3(6,4)}
* **Cluster 3:** {A2(2,5), C1(1,2)}

**Recalculate centers:**

* **New Cluster 1 Center:** ((2+5+4)/3,(10+8+9)/3)=(11/3,27/3)=(3.67,9)
* **New Cluster 2 Center:** ((8+7+6)/3,(4+5+4)/3)=(21/3,13/3)=(7,4.33)
* **New Cluster 3 Center:** ((2+1)/2,(5+2)/2)=(1.5,3.5)

The cluster assignments remain the same in the next iteration. Therefore, the algorithm has converged.

The final three clusters are:

* **Cluster 1:** **{A1(2,10), B1(5,8), C2(4,9)}**
* **Cluster 2:** **{A3(8,4), B2(7,5), B3(6,4)}**
* **Cluster 3:** **{A2(2,5), C1(1,2)}**