**Group 8 - Amazon Redshift**

**Vinesh Narendra Sangoi (U55816884)**

**Leelamahesh Chakravarthy Kodi (U28275523)**

**Neha Satpute (U19160274)**

**Payal Samanta (U76228941)**

# **What is Amazon Redshift?**

Amazon Redshift is a powerful, fully managed data warehousing service provided by Amazon Web Services (AWS). Designed for high-performance analysis of vast datasets, Redshift leverages a columnar storage architecture and parallel processing to deliver exceptional query speeds. With seamless integration into the AWS ecosystem, Redshift enables organizations to scale their data infrastructure on-demand, accommodating the evolving demands of analytical workloads. Offering features like automated backups, robust security measures, and compatibility with popular BI tools, Redshift empowers users to glean valuable insights from their data efficiently. Its ability to handle complex queries, support concurrency, and interact with data lakes through Redshift Spectrum makes it a go-to solution for businesses seeking a cloud-based, scalable, and cost-effective data warehousing solution.

# **A Detailed Description of The KDD Nuggets Referenced Data.**

We would like to disclose that the dataset used in our analysis has been obtained from Kaggle, a prominent platform for sharing and discovering datasets. The links to the specific datasets utilized in this study are as follows:

**Adidas Sales Dataset**

<https://www.kaggle.com/datasets/heemalichaudhari/adidas-sales-dataset>

**Loan sanction**

<https://www.kaggle.com/datasets/rishikeshkonapure/home-loan-approval>

# **A detailed description of the Product (Transactional or Analytical). You must describe why the design of the Product makes it Transactional or Analytical.**

**Adidas sales dataset:**

This dataset is inherently analytical, providing a comprehensive view of retail sales performance which can be leveraged for strategic business decisions. The combination of retailer information, geographic data (region, state, city), and product details allows for multi-dimensional analysis of sales trends. Retailer ID and product ID serve as unique identifiers, enabling tracking of sales down to the individual item sold by each retailer.

The inclusion of invoice dates allows for time series analysis to understand sales dynamics over time, identify seasonal trends, and measure the impact of promotions or external events. Price per unit and units sold offer a detailed look at sales volume and revenue generation, which, when combined with total sales and operating profit, enable profitability analysis at various granularities—from individual products to entire regions.

Operating margin data provides insights into cost management effectiveness, while sales method data can reveal consumer purchasing preferences and the performance of different sales channels. Furthermore, the geographical identifiers (region\_id, state\_id, city\_id) facilitate regional market analysis, helping businesses optimize distribution and marketing strategies.

Overall, this dataset is a valuable analytical tool for tracking sales performance, optimizing supply chain operations, tailoring marketing efforts, and ultimately driving strategic growth initiatives.

**Loan sanction dataset**:

The dataset consisting of loan ID, gender, marital status, dependents, education, self-employment status, applicant and co-applicant income, loan amount, and loan tenure is highly analytical in nature. It provides a rich source for understanding and predicting loan repayment behaviors and assessing credit risk.

Firstly, the inclusion of demographic information like gender, marital status, and dependents allows for demographic segmentation and analysis, which can highlight patterns in loan uptake and repayment across different social groups. Education and self-employment status provide insight into the financial stability and earning potential of applicants, key factors in determining loan eligibility and sustainability.

Applicant and co-applicant income data are central to any credit risk model, offering a direct measure of the ability to repay. Loan amount and tenure data contribute to understanding the risk profile of the loan itself, including how manageable repayments will be over time given the applicant's financial situation.

Analyzing this dataset can help financial institutions tailor their loan products to specific customer segments, forecast future loan performance, and make informed lending decisions. Risk managers can develop predictive models to estimate the probability of default, while marketing teams can identify profitable niches and underserved demographics. The dataset is a valuable asset for building strategies to minimize defaults, optimize loan offerings, and enhance customer satisfaction.

# **A detailed description of the Product data structures.**

Amazon Redshift is a fully managed, petabyte-scale data warehouse service in the cloud. It is designed for large-scale data set storage and analysis and also for performing large-scale database migrations. The product data structures within Amazon Redshift are organized into tables and columns and can be categorized into a few key concepts:

**Tables and Columns:**

Redshift organizes data into tables. A table is a collection of data organized into rows and columns, similar to relational databases.

Tables can have a variety of column types including integers, decimals, varying character types, booleans, date/time types, and more.

**Data Distribution Styles:**

Redshift allows you to specify how data is distributed across nodes in the cluster. This can be KEY distribution (where rows are distributed according to the values in one column), ALL distribution (where the entire table is replicated on every node), or EVEN distribution (where rows are distributed evenly across all nodes).

**Sort Keys:**

Redshift uses sort keys to organize data within each block and can improve performance for certain types of queries. For example, if you frequently query data by date, it might make sense to set a date column as a sort key.

**Data Compression:**

Redshift automatically compresses columnar data, but you can also set compression encodings manually to optimize for your specific workload.

**Schema Design:**

A schema is a collection of tables. Redshift supports multiple schemas within each database and different authorization levels can be set for each schema.

**Primary Keys and Foreign Keys:**

Redshift supports primary keys and foreign keys; however, unlike other relational databases, these do not enforce uniqueness or referential integrity. They are used for query optimization purposes.

**Workload Management (WLM):**

WLM allows you to manage priorities within the Redshift data warehouse ensuring that fast-running queries do not get stuck behind long-running queries.

**Columnar Storage:**

Redshift stores data by columns, which allows for more efficient I/O and can significantly improve query performance as only the columns needed for a query are processed.

**Concurrency Scaling:**

Redshift offers concurrency scaling which automatically adds additional cluster capacity when needed to handle bursts in query activity.

**Snapshots and Continuous Backups:**

Redshift automatically takes snapshots of your data warehouse, enabling point-in-time recovery. It also continuously backs up your data to Amazon S3.

**Security:**

Redshift provides robust security measures, such as encryption in transit and at rest, VPC support, and the ability to manage permissions at a granular level.

**Materialized Views:**

Redshift supports materialized views which can store the result of a query and can be refreshed on demand, making it easier to perform complex calculations for frequently run queries.

Amazon Redshift's architecture is designed to work well with large-scale data processing and analysis workflows. It integrates well with various data loading and ETL (extract, transform, load) tools, BI (business intelligence) applications, and is a popular choice for companies looking to perform complex queries on large datasets.

# **A detailed description of the ETL process.**

1. **Create the staging, dimension, and fact table:**

* Create a staging table that matches the structure of the data you want to load.
* The staging table is used to hold the data before it's transformed and loaded into the fact table.
* Create dimensions and fact tables as well. Create an IAM role with the necessary permissions to allow redshift to access data in the S3 bucket.

A screen shot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

1. **Upload files in S3 bucket:**

* Create an S3 bucket and upload the required file which we want to load to the staging table.
* S3 is designed to store and retrieve any amount of data from anywhere on the web.
* It is widely used for data storage, backup, content distribution, and data archiving.

A screenshot of a computer

Description automatically generated

1. **Load data from S3 to Staging table:**

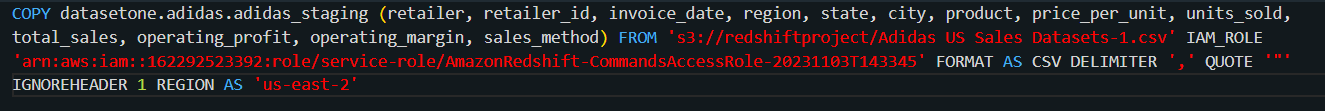
* Select “Load Data” to import data into the staging table.
* In the dialog box, browse S3 file location, select file format and choose delimiter character.
* In advance settings, you have options to select the required parameters for data conversion and handling null values.
* Click on next. Choose database, schema, table name and IAM role. Choose column names for one-to-one mapping (csv to staging table) and click on load data.
* A query with COPY command will be generated for bulk data loading. Run the query and data will be loaded from csv file to staging table.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated



1. **Load data into dimensions and update Staging table:**

* Using the staging table, we will insert data into dimensions using INSERT statements and auto generate id of all dimensions and update ids that are present in staging table using the dimensions.

A computer screen with white text

Description automatically generated

1. **Load data into Fact table:**

* Once the staging table is updated, use INSERT statement to move data into the final fact table. Necessary data transformations are also performed during this step.

A screen shot of a computer screen

Description automatically generated

1. **Business Question: State wise online sales distribution of Adidas in Northeast Region.**

