PROJECT PROPOSAL

GROUP MEMBERS:
Rohith Reddy, Yamini Muthyala, Shashank Reddy, Sakshi Mukkirwar, Swati

PROJECT NAME:
Cloud Analytics and data warehouse implementation for
VEHICLE INSURANCE

Abstract:

This project aims to help an insurance company predict whether its health insurance customers from the past year will also be interested in vehicle insurance provided by the company. The data includes demographics such as gender, age, and region code type, as well as information about the customers' vehicles and insurance policies. By building a predictive model, the company can optimize its communication strategy and increase revenue. To achieve this, we will build an OLTP database using a Star Schema Model with fact tables and dimension tables, using cloud databases such as **Amazon Redshift** and **MySQL**. We will extract, transform and load the data using Python scripts and orchestrate the ETL pipeline using **Apache Airflow**. The data will then be analyzed using SQL queries and visualized through business intelligence tools such as **Microsoft Power BI** or **Tableau**. Interesting queries such as the number of events in each country, cumulative deaths report, and top 10 analysis in time and region based on **KPI metrics** will be performed. This project showcases how cloud-based technologies and machine learning algorithms can be used to optimize business strategies and increase revenue.

Motivation:

This project can provide numerous benefits to the insurance company. By building a predictive model to identify potential customers for vehicle insurance, the company can optimize its communication strategy and increase revenue. The data analysis and visualization can also provide valuable insights into the company's operations, customer behavior, and potential areas for growth. Additionally, working with cloud-based technologies such as Amazon Redshift and MySQL can enhance the company's data management capabilities and improve the efficiency of its processes. This project is not only challenging but also highly relevant in today's data-driven business environment, offering an opportunity to develop valuable skills and make a significant impact on the company's success.

Use Cases:

Here are some potential use cases for the insurance company using the results of this project:

1. **Targeted Marketing**: The predictive model can help the company identify potential customers who are most likely to purchase vehicle insurance, based on their demographic and policy information. This can enable the company to create targeted marketing campaigns that focus on these potential customers and increase the effectiveness of their outreach efforts.

2. **Revenue Optimization**: By identifying customers who are most likely to purchase vehicle insurance, the company can optimize its revenue by focusing its sales efforts on these customers. This can also help the company tailor its insurance products to meet the specific needs and preferences of its customers, increasing their satisfaction and loyalty.

3. **Risk Management**: The data analysis and visualization can provide insights into the company's risk management strategies and potential areas for improvement. By analyzing patterns in claims data and identifying high-risk policyholders, the company can develop more effective risk management strategies and reduce its exposure to losses.

4. **Operational Efficiency**: By using cloud-based technologies such as Amazon Redshift and MySQL, the company can streamline its data management processes and improve operational efficiency. This can include automating data extraction, transformation, and loading (ETL) processes, as well as providing real-time access to data for decision-making and reporting.

Overall, the results of this project can provide significant benefits to the insurance company, including increased revenue, improved customer satisfaction, and enhanced operational efficiency.

Methodology:

Steps to perform data analytics for the insurance dataset using Amazon Redshift and Python:

1. **Data exploration**: Explore the data to understand the structure of the dataset and the relationships between its attributes. This will help identify any data quality issues, such as missing or inconsistent values.
2. **Data cleaning and preparation**: Clean and prepare the data for analysis by addressing any data quality issues, removing any irrelevant or redundant attributes, and transforming the data as needed.
3. **Data transfer to AWS Redshif**t: Transfer the cleaned and prepared data to Amazon Redshift, a cloud-based data warehouse that can store and analyze large volumes of structured data.
4. **Data modeling**: Design and build a star schema model for the data warehouse, which includes fact tables and dimension tables. This will help organize the data in a way that supports efficient querying and analysis.
5. **ETL pipeline development:** Develop an ETL (extract, transform, load) pipeline in Python using the AWS SDK to automate the data transfer and transformation process. This pipeline should read data from the source, apply any necessary transformations, and load the data into the data warehouse.
6. **Data analysis**: Once the data is loaded into the data warehouse, perform data analysis using SQL queries to gain insights into the data. For example, you can analyze the distribution of policies across different regions, age groups, or genders, or identify trends in policy premiums over time.
7. **Visualization and reporting**: Use a business intelligence tool such as Microsoft Power BI or Tableau to create visualizations and reports that summarize the insights gained from the data analysis. These visualizations can help stakeholders understand key trends and make informed decisions based on the data.

In terms of specific steps for transferring data to AWS and automating DAGs using Python:

1. To transfer data to AWS Redshift, you can use the AWS SDK for Python to programmatically load data from various sources such as Amazon S3 or MySQL database.
2. To automate DAGs, you can use a workflow management tool such as Apache Airflow or AWS Step Functions. These tools allow you to define DAGs using Python code and schedule them to run on a recurring basis. For example, you can use Airflow to define a DAG that runs every night to transfer data from MySQL to Redshift, transform the data, and update reports.

References:

1)Insurance Dataset:https://www.kaggle.com/code/yashvi/vehicle-insurance-eda-and-boosting-models/input

2)"Insurance Analytics: The Key to a Profitable Future" by McKinsey & Company:
https://www.mckinsey.com/industries/financial-services/our-insights/insurance-analytics-the-key-to-a-profitable-future

3)”Data Analytics in the Insurance Industry: Realizing the Benefits" by PwC: https://www.pwc.com/us/en/library/data-analytics-insurance-industry.html

4)”Insurance Analytics - A Guide to Understanding the Value of Analytics in Insurance" by Insurance Nexus: https://www.insurancenexus.com/analytics/insurance-analytics-guide-understanding-value-analytics-insurance

5)"Data analytics for automobile insurance pricing: an empirical analysis" by G. Bolton and M. L. Moriarty. This paper analyzes the use of data analytics for automobile insurance pricing and provides empirical evidence of the benefits of using data analytics for insurance pricing