**Bank Loan Eligibility Prediction in Large Scale Data Streams: A Big Data Approach**

Predicting bank loan eligibility in large-scale data streams is a complex task that requires a big data approach. Big data technologies and techniques enable the processing and analysis of vast amounts of data in real-time, allowing banks to make accurate and timely decisions regarding loan approvals.

By leveraging big data technologies, banks can process and analyze vast amounts of customer data in real-time, enabling them to make accurate loan eligibility predictions. This approach helps streamline the loan approval process, reduce manual effort, and improve the overall efficiency and accuracy of decision-making in the banking industry.

This is a proposal on the methodology and process that will be adapted to achieve this project:

1. **The Data**

Financial data of this nature can often be difficult to find this is because they contain private information of customers or heavily imbalanced. The data to be used in this project contains **981** historical loan application records of individuals. The data has the following features:

* Gender: Male or Female
* Married: Yes or No
* Dependents: Number of dependents
* Education: Graduate or Not Graduate
* Self-Employed: True or False
* Applicant Income: Yearly Income
* Co-applicant Income: Yearly Income
* Loan Amount
* Term: How long

1. **The Methodology:** The following methodologies would be used.

* **Supervised Learning**: Since we have historical loan application data with loan status as the target variable, supervised learning is a suitable methodology. It involves training a predictive model on labeled data to make predictions on new, unseen data.
* **Real-time Data Stream Processing**: Given the nature of the project, where loan applications arrive in real-time data streams, a real-time processing approach is required. AWS Kinesis Data Streams will be used to ingest and process the streaming data. AWS Lambda functions will perform feature extraction, transformation, and model predictions on each incoming record.
* **Visualization and Reporting**: Create visualizations, dashboards, and reports to effectively communicate the loan eligibility predictions and insights. Amazon QuikSight will be used for this purpose.

1. **The Roles of AWS Services:**

* **Kinesis**: AWS Kinesis Data Streams will be used to ingest and preprocess the streaming data.
* **Lambda**: AWS lambda functions would serve as a producer and consumer for our AWS Kinesis data stream. Our producer function would ingest data via an external API and our consumer function will process and run our pre-trained machine learning model to predict loan eligibility.
* **EFS**: AWS Elastic File System would act as a file system for our model artifacts.
* **API Gateway**: API gateway enables us collect data through a REST API endpoint.
* **QuikSight**: With Amazon QuikSight we can create real time visualizations of data we ingest into our data stream.
* **EC2**: An EC2 instance would enable us move model artifacts from our local pc directly into our EFS through mounting.
* **SNS**: Simple notification service will be used to send out real time email notifications of model results from loan applications.

1. **Process Breakdown:**

**Using Jupyter Notebooks (Offline)**

* Step 1: Ingest data files (.csv)
* Step 2: Pre-process and clean data.
* Step 3: Perform extensive exploratory data analysis to understand data and its features.
* Step 4: Build model (experiment several algorithms - logistic regression, decision trees, random forests and gradient boosting)
* Step 5: Save/Export model artifacts.

**Using Github**

* Step 1: Create a simple user interface for customers to apply for loans: using Streamlit UI
* Step 2: Upload on github

**Using AWS (Online)**

* Step 1: Create an EFS, mount on EC2, Upload model artifacts.
* Step 2: Create a Lambda function (producer)
* Step 3: Create a POST API using API Gateway, use it as source for producer.
* Step 4: Create a Kinesis data stream.
* Step 5: Create a Lambda function (consumer), use Kinesis data stream as source.
* Step 6: Create a Dynamo DB to save ingested data from Kinesis data stream.
* Step 7: Integrate SNS as destination for Lambda function (consumer).
* Step 8: Visualize data Ingested with Quiksight.