Data Pipeline for Loan Default Prediction

Tan Yin Yun

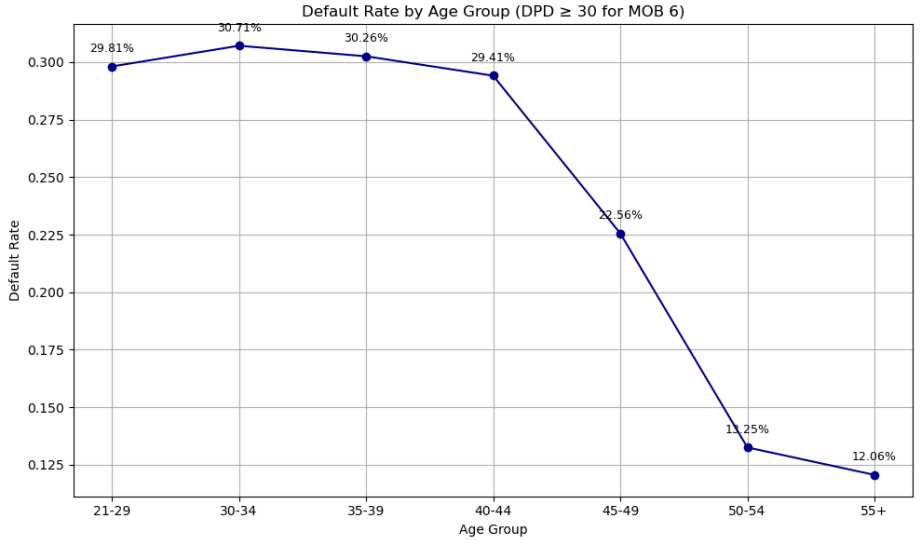
CS611 Machine Learning Engineering

Assignment 1

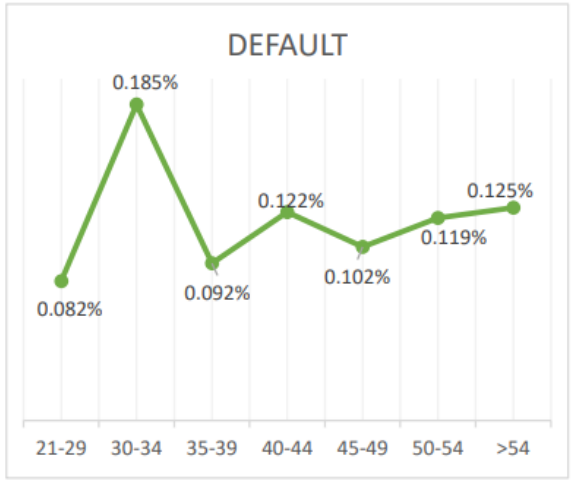
Business problem & objective

Business Problem: Default rates for cash loan product are way above Singapore's consumer averages by age group:

- Company XX: **12% - 30%**
- Singapore consumer average: **0.082% - 0.185%**



Source: Company XX LMS data (for loans approved in 2023)



Source: Consumer Credit Index Q1 2025

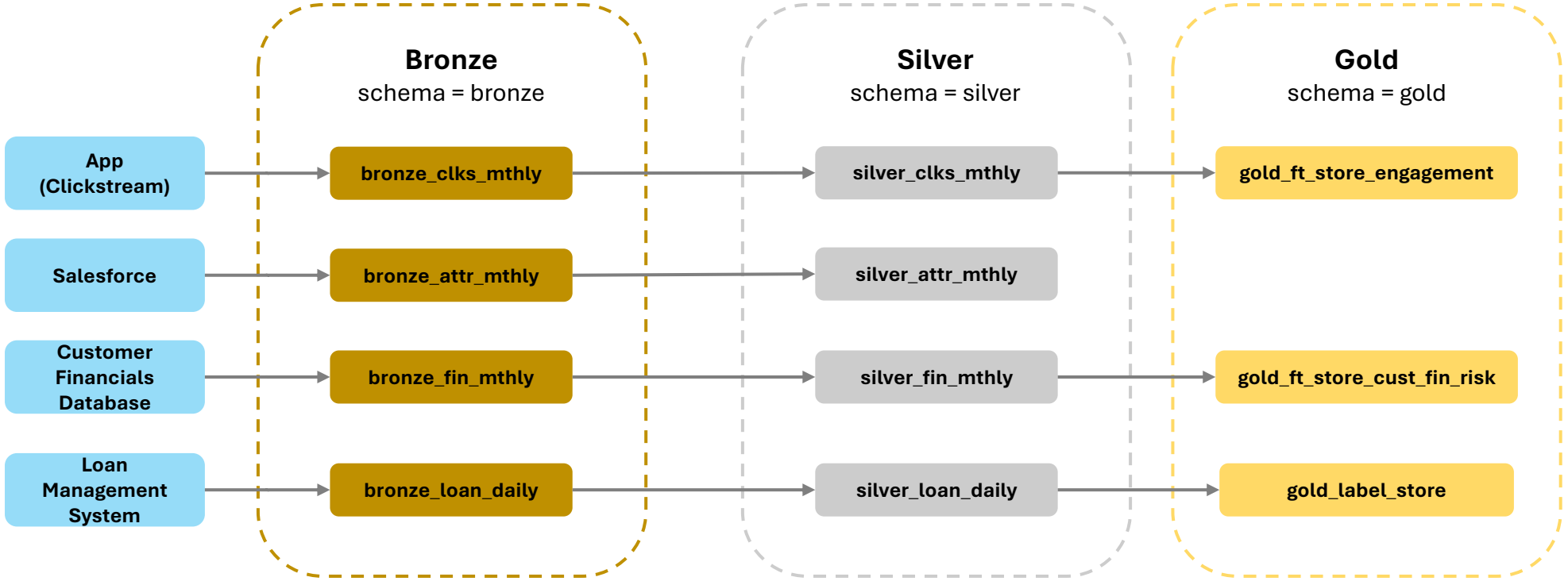
Impact: **20%** increase in cost of funds, resulting in **4% decline** in overall profitability in 2023

Objective: At loan application stage, predict (using machine learning) if customer will default as an approval criteria. This will reduce chance of accepting customers with high potential of default.

Scope (Phase 1): Build the **data pipeline** that will support the machine learning model

Proposed data pipeline using medallion architecture

Raw data from the company's systems (e.g. app, Salesforce, databases and systems) will be processed incrementally in 3 layers (bronze, silver, gold) to create high quality and structured data for diverse stakeholder usage in this project.



Aspect	Bronze	Silver	Gold
Processing at this layer	Raw data ingestion	Data cleaning and validation	Aggregation, filtering, dimensional modelling
Purpose	Permanent copy of raw data for archival, data lineage, auditability, re-processing	Merge, conform / standardize and clean the data for ease of use in multiple projects	Business reporting, filtering out the key metrics often used by business teams
Intended User	Data engineers, Compliance & Audit	Data engineers, analysts, data scientists	BI, business teams, ops and executives

Bronze tables: Raw data ingestion and partitioning by month

Insights from Exploratory Data Analysis

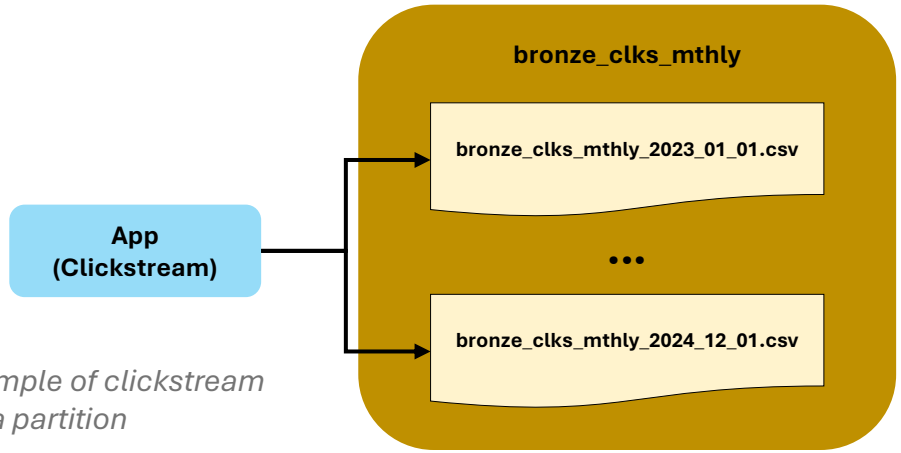
- All raw data is collected at the 1st day of each month (frequency: **monthly**), recorded with 'snapshot_date' field

	fe_1	fe_2	fe_3	fe_4	fe_5	fe_6	fe_7	fe_8	fe_9	fe_10	...	fe_13	fe_14	fe_15	fe_16	fe_17	fe_18	fe_19	fe_20	Customer_ID	snapshot_date
0	63	118	80	121	55	193	111	112	-101	83	...	-16	-81	-126	114	35	85	-73	76	CUS_0x1037	2023-01-01
1	-108	182	123	4	-56	27	25	-6	284	222	...	-14	-96	200	35	130	94	111	75	CUS_0x1069	2023-01-01
2	-13	8	87	166	214	-98	215	152	129	139	...	26	86	171	125	-130	354	17	302	CUS_0x114a	2023-01-01
3	-85	45	200	89	128	54	76	51	61	139	...	172	96	174	163	37	207	180	118	CUS_0x1184	2023-01-01
4	55	120	226	-86	253	97	107	68	103	126	...	76	43	183	159	-26	104	118	184	CUS_0x1297	2023-01-01

Example of clickstream data

Bronze Table Design

- To facilitate efficient querying the raw data, especially the latest months, all raw data is partitioned by month
- This also enables compression of older historical data in future



Example of clickstream data partition

Silver tables: Data cleaning & validation

Insights from Exploratory Data Analysis

- The raw data from the bronze tables were extremely dirty, probably due to lack of validation logic upstream
- Problems: non-numeric characters in numeric data, invalid and extreme values, missing data, typo errors

	fe_1	fe_2	fe_3	fe_4	fe_5	fe_6
0	63	118	80	121	55	193
1	-108	182	123	4	-56	27
2	-13	8	87	166	214	-98
3	85	45	200	89	128	54

	Customer_ID	Name	Age	SSN	Occupation
2	CUS_0x100b	Shirboni	19.0	#F%$D@*&8	Media_Manager
9	CUS_0x102e	Rhysn	26.0	#F%$D@*&8	Scientist
15	CUS_0x1044	Maki Shirakip	44.0	#F%$D@*&8	_____

Example of dirty data from different data sources

Silver Table Data Cleaning Logic

- Schema was enforced
- Missing values standardized to not a number (NaN)
- Table-specific validations (*right, bottom*) from business logic

clickstream	Data Type	Tasks
fe_1	Integer	Remove negative values
...		
fe_20	Integer	
Customer_ID	string	
snapshot_date	string	

Schema	Data Type	Tasks
Customer_ID	string	
Annual_Income	decimal	Replace negative values (where applicable) Outliers windsorized Categorical: Ensure enums conformed and encoded
...		
Payment_Behaviour	string	
Monthly_Balance	decimal	
snapshot_date	string	

attributes	Data Type	Tasks
Customer_ID	string	
Name	string	Remove disallowed punctuations, trailing spaces
Age	Integer	Replace invalid ages (<0, >100) as np.nan (2.5% data)
SSN	string	Replaced data not in SSS format with np.nan
Occupation	string	Replace _____ as np.nan (7% data)
snapshot_date	string	

Silver tables: Feature engineering to create commonly used business metrics

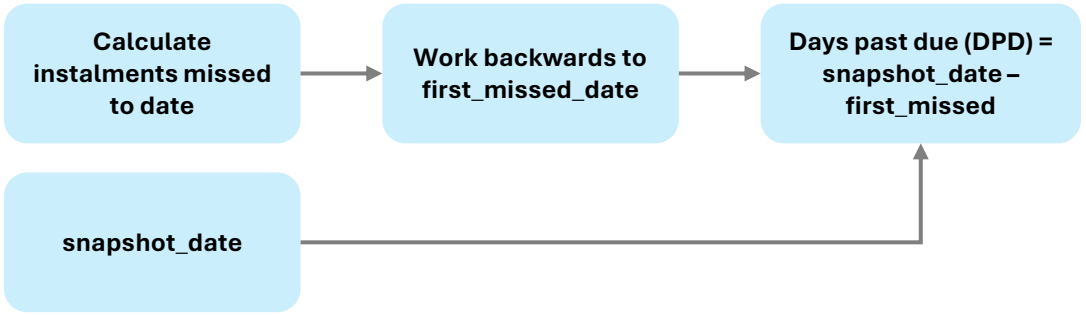
- Besides data cleaning, new features were engineered at the Silver Table stage, based on domain knowledge from business team and understanding of the cash loan approval process
- These features could be useful metrics for the machine learning model

Features	Description	Formula
Num_Fin_Pdts	Total no. of financial products the customer is using	Num_Bank_Accounts + Num_Credit_Card + Num_of_Loan
Loans_per_Credit_Item	No. of loans across the no. of credit products the customer is using, indicative of how he may be funding his loans	Num_of_Loan/ (Num_Bank_Accounts + Num_Credit_Card)
Debt_to_Salary	The proportion of debt to the customer's monthly salary, indicative in long term how long to pay off	Outstanding_Debt / Monthly Inhand Salary
EMI_to_Salary	Total loan paid out per month as a proportion of his monthly salary, indicative in short term whether his salary can cover his commitments	Total_EMI_per_month / Monthly Inhand Salary
Repayment_Ability	Amount of salary left after paying for loans	Monthly_Inhand_Salary - Total_EMI_per_month
Loan_Extent	Average delay (in days), normalized by loans. Indicates whether the customer is having issue across his commitments	Delay_from_due_date / Num_of_Loan

Gold label table: Ground truth based on business definition

To create the label store, the following steps were taken:

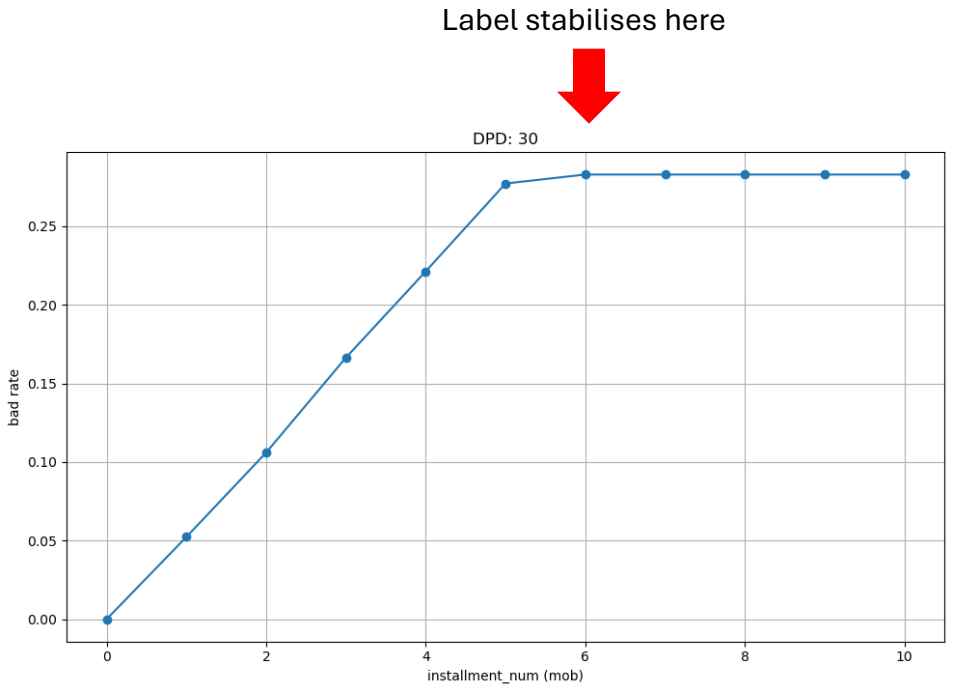
1. Computations were performed to check customer's DPD



2. To avoid labelling customers that had missed payment by mistake, causing potential false positives in the label, DPD threshold is **≥ 30 days** for customer to be considered a delinquent case
3. Exploratory data analysis found bad rate stabilized at **6 months on book (mob).** This would be a suitable juncture to generate the label

	loan_id	Customer_ID	loan_start_date	label	label_def	snapshot_date
6	CUS_0x1000_2023_05_01	CUS_0x1000	2023-05-01	1	30dpd_6mob	2023-11-01
39	CUS_0x1011_2023_11_01	CUS_0x1011	2023-11-01	0	30dpd_6mob	2024-05-01
50	CUS_0x1013_2023_12_01	CUS_0x1013	2023-12-01	0	30dpd_6mob	2024-06-01

Resultant gold label store table



Exploratory data analysis on bad rate vs mob

Gold feature tables: Domain-specific with critical metrics

Design Considerations

- Multiple gold tables are created by business domain, so that stakeholders can work with domain-specific data without unnecessary joins or filters.

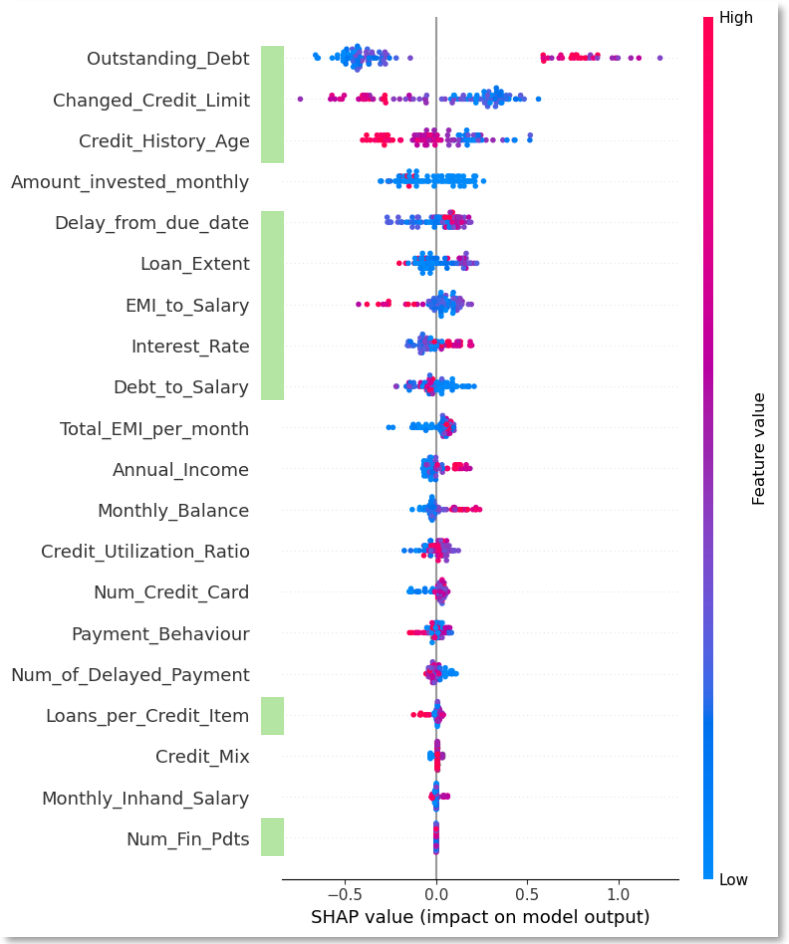
Gold Feature Tables

Given default definition as ≥ 30 days past due @ 6th month on book, 2 gold feature tables were created with the following logic:

- **engagement_tab**: Clickstream data (event-based) pivoted to contain last 6 months to understand customer behaviour leading up to default
- **cust_fin_risk_tab**: Financials data (event-based at point of application) filtered to key business metrics that were:
    1. selected by business team for reporting
    2. recommended by data science team as useful predictors based on a POC Machine Learning model

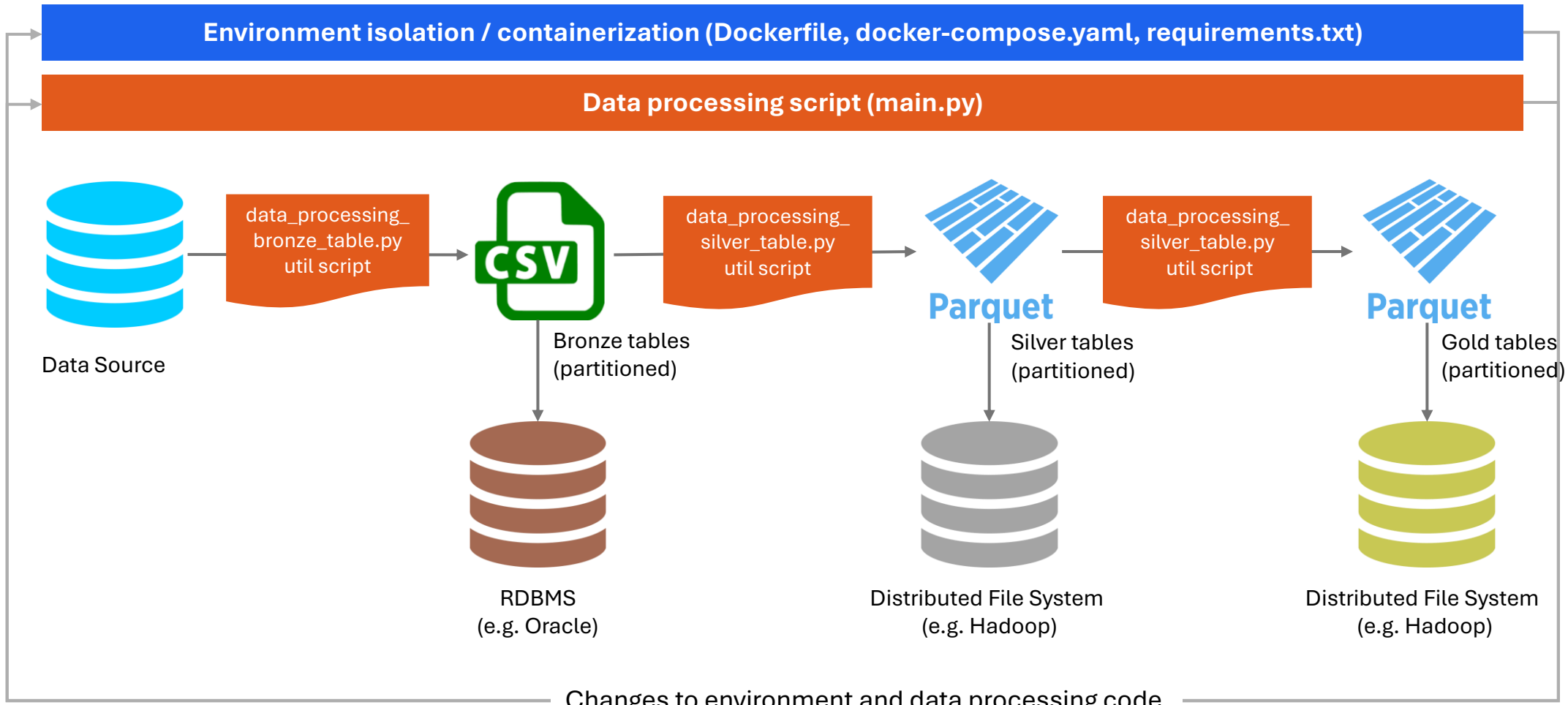
	Customer_ID	click_1m	click_2m	click_3m	click_4m	click_5m	click_6m	snapshot_date
0	CUS_0x1000	108	65	115	0	224	172	2024-06-01
1	CUS_0x100b	122	212	210	38	66	92	2024-06-01
2	CUS_0x1011	49	0	161	43	1	94	2024-06-01
3	CUS_0x1013	147	0	0	0	181	61	2024-06-01
4	CUS_0x1015	122	251	170	0	242	22	2024-06-01

engagement_tab:
Clickstream data gold table



cust_fin_risk_tab:
Feature importance based on POC ML model. Fields selected (green)

Proposed tech implementation of data pipeline



Tech Design Choices

- Docker: software portability across different envs
- PySpark: Handles large datasets efficiently
- Github: Version control and governance
- Parquet: Fast querying for multiple uses