On Time or Not?

Predicting Delivery Delays on Brazil’s largest e-commerce platform, Olist

Chentong Hao

October 24, 2025

Data Science Institute

https://github.com/chentonghao/Midterm

1 Problem Introduction

Have you ever ordered something online, checked the tracking page five times a day,and still wondered — *"Why hasn't my package arrived yet ???"*

Problem Definition:

Predict whether an e-commerce order will be delivered on time.

Why It Matters:

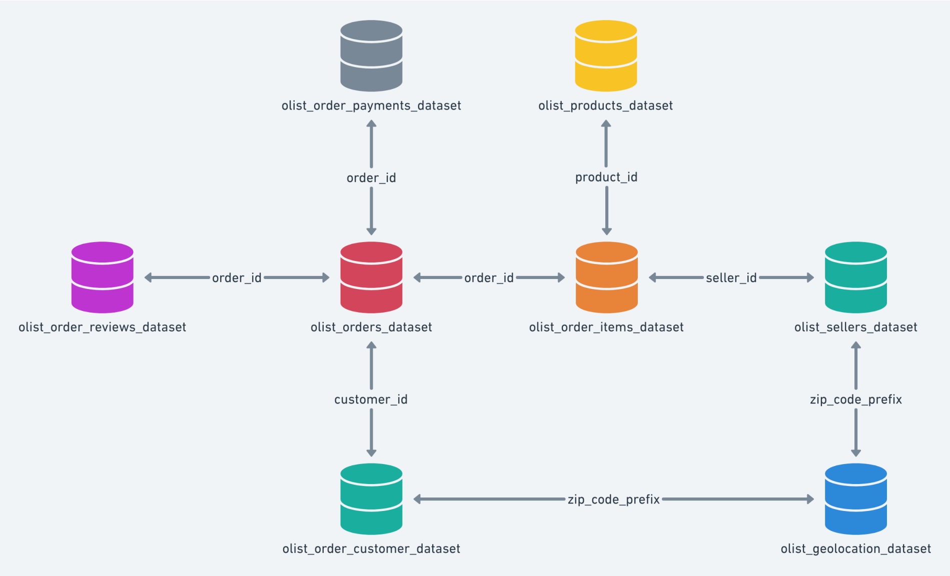
- **Delivery punctuality → key to customer satisfaction**
- **Late orders → harm seller reputation**
- **Accurate delay prediction → optimize logistics operations**

Type of Task: Binary Classification

- **Target: is_late (actual delivery date VS the estimated delivery date)**
- **1 → Delivered after estimated date**
- **0 → Delivered on time**

2 Dataset

- Source: Olist — Brazil's largest e-commerce platform
- Public release on Kaggle (Olist data-sharing initiative)
- Real transactional records from 2016–2018
- Each row = one real order



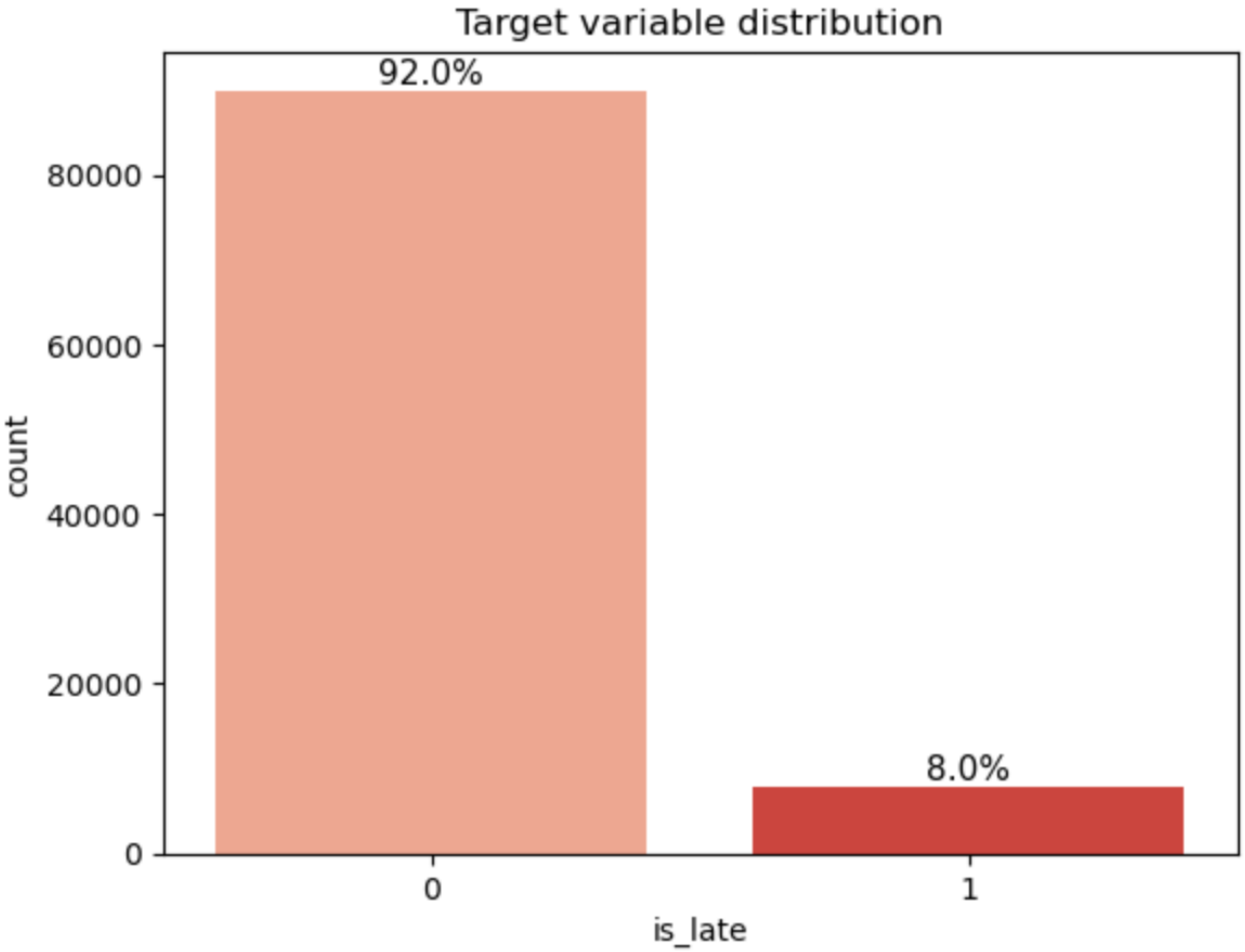
Three challenges:

- Large: over 100k orders and multiple linked tables.
- Non-iid : multiple orders come from the same customers

(each order is an independent sample in my task)

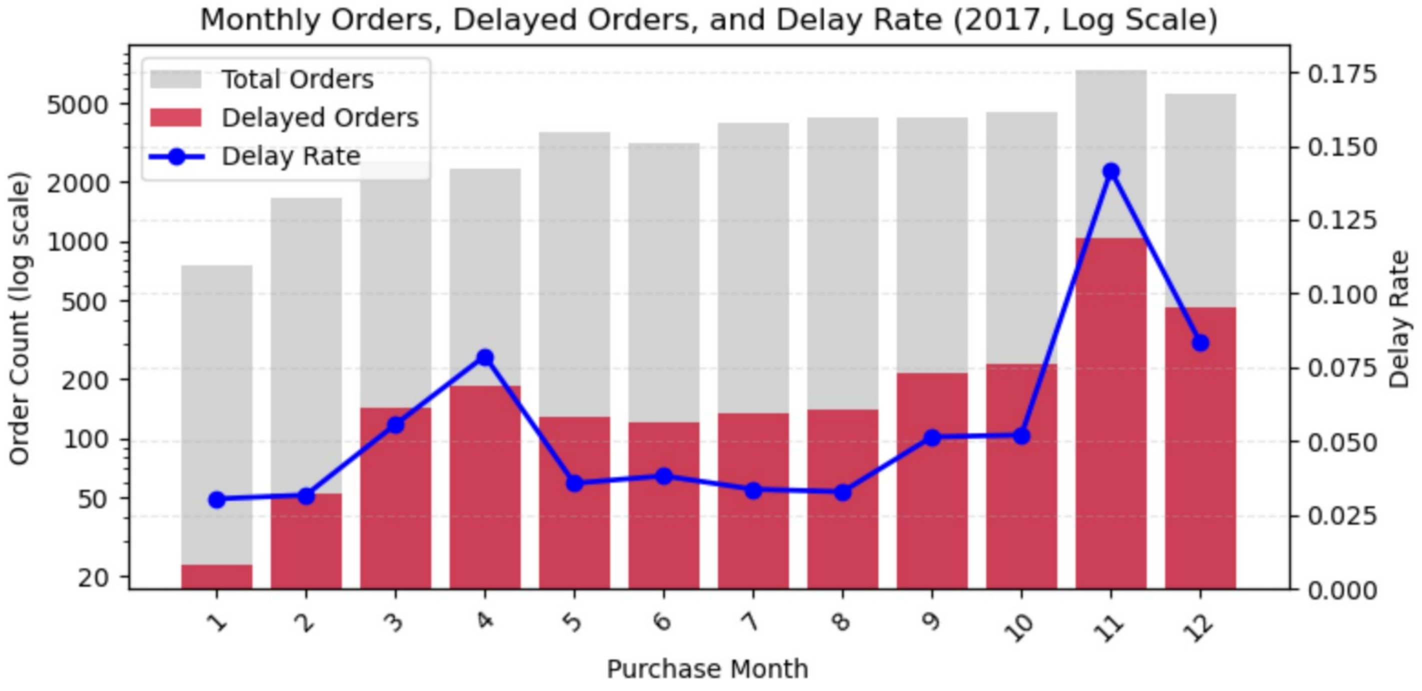
- Some tables contain missing values(<0.1%) .

3 EDA



A strong class imbalance

3 EDA

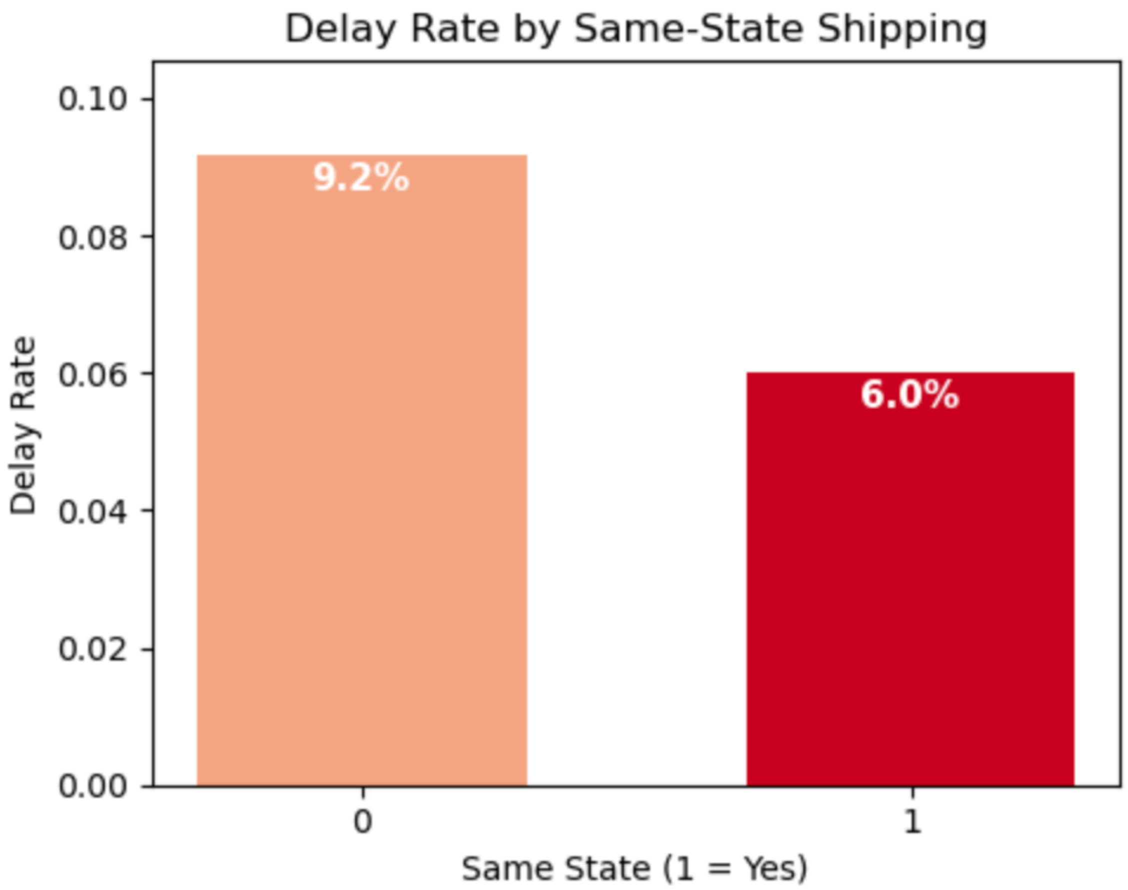


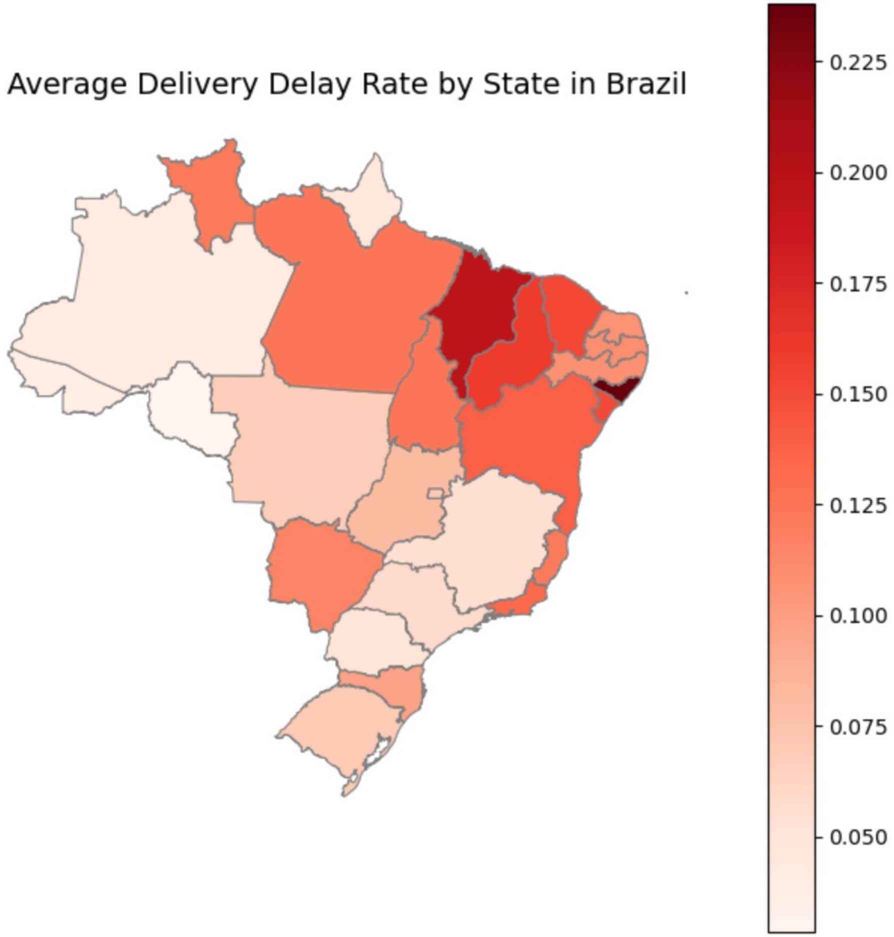
- **Delay peaks in November (Black Friday)**
- **Second spike in April (National public holidays)**
- **Holiday/Shopping Period cause?**

Holidays and Observances in Brazil in 2017

Apr 14	Friday	Good Friday	National Holiday
Apr 16	Sunday	Easter Sunday	Observance
Apr 21	Friday	Tiradentes Day	National Holiday
May 1	Monday	Labor Day	National Holiday

3 EDA





- **Cross-state orders → higher delay rate**
- **Same-state orders → fewer delays**
- **Shipping distance likely main factor**

4 Splitting and Preprocessing

Splitting:

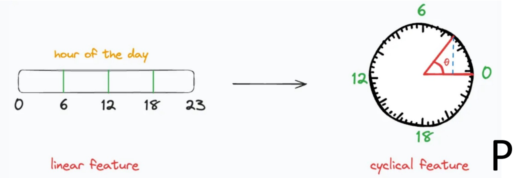
- Used a stratified train-test split to divide the dataset into 80% training + validation and 20% test.
- Within the 80% training + validation portion, applied 4-fold StratifiedKFold cross-validation

```
Train+Val size: (78255, 15), Test size: (19564, 15)

Fold 1
Train shape: (58691, 68), Val shape: (19564, 68), Test shape: (19564, 68)

Fold 2
Train shape: (58691, 68), Val shape: (19564, 68), Test shape: (19564, 68)

Fold 3
Train shape: (58691, 68), Val shape: (19564, 68), Test shape: (19564, 68)

Fold 4
Train shape: (58692, 67), Val shape: (19563, 67), Test shape: (19564, 67)
```

Feature Group	Features & Descriptions
Numeric (7)	• n_items - number of items per order • total_price - total product value • total_freight - total shipping cost • avg_price - average price per item • payment_value - actual payment amount • seller_avg_score - seller's average review score • purchase_to_estimated_days - promised delivery duration
Ordinal (1)	• payment_installments - number of payment installments
Categorical (4)	• main_payment_type - payment method • same_state -1 if buyer & seller are in same state • customer_state - customer's state code • seller_state - seller's state code
Cyclic (2)	• purchase_month - month of purchase (1-12) • purchase_dow - day of week (0-6)

Preprocessing:

- StandardScaler → numerical & linear ordinal features
- OneHotEncoder → categorical features (payment_type, state)
- Cyclic encoding → temporal features (month, day_of_week)



Picture from blog cyclical-feature-engineering

Q&A