The Lead Data Scientist Handbook

From Insight to AI Orchestration

Project: Customer Behavior Analysis for Subscription Retention
Dataset: Alibaba User Behavior (RL)

Your Three Phases of Work

1. **Phase 1 (Now – Feb 9):** The Storyteller. EDA, feature selection, and the "Problem Setting" presentation.
2. **Phase 2 (Feb 10 – Mar 2):** The Architect. Building the RL Agent that combines your team's models.
3. **Phase 3 (Mar 3 – Mar 9):** The Closer. Final report, slides, and defense.

Status: CONFIDENTIAL TEAM DOCUMENT
Date: February 2026

1. The Mission: Why You Matter

The Engineers build the car. **You determine where we drive.**

We are using the **Alibaba User Behavior Dataset**. On the surface, it looks like a shop. Your job is to convince the professor that this is actually a **Subscription Retention Problem**.

1.1 The ”Creative Problem Setting” (The Story)

You must sell this narrative in your presentation:

- **The Old World:** Subscription = Monthly Contract (Netflix). Churn happens once a month.
- **The New World:** Subscription = Attention (TikTok, Alibaba). Churn happens *every second.*
- **Our Thesis:** A user's ”Session” is a micro-subscription. As long as they keep scrolling (Action), they are ”Retained.” If they stop (Terminal State), they have ”Churned.”
- **Value Estimation:** We don't just want them to stay; we want to predict the *Total Session Revenue* (LTV).

2. Getting Your Hands on Data (Phase 1)

The Data Engineer has already processed the raw logs (6GB) into clean Parquet files (small) stored in the `processed-data` bucket in MinIO. You do not need to process raw text.

2.1 Step 1: Install Libraries

Run this in your terminal to get the analysis tools:

```
pip install pandas numpy seaborn matplotlib minio pyarrow fastparquet
```

2.2 Step 2: Connect to the Data Lake

Create a new notebook `notebooks/eda_analysis.ipynb`. Use this code to pull the clean data:

```
import pandas as pd
from minio import Minio
from io import BytesIO

# Connect to Local Data Lake
client = Minio(
    "localhost:9000",
    access_key="minioadmin",  # Or whatever the engineer set
    secret_key="minioadmin",
    secure=False
)

# Download a few chunks to analyze (we don't need all 1.5M rows for EDA)
BUCKET = "processed-data"
objects = client.list_objects(BUCKET)
files = [obj.object_name for obj in objects if obj.object_name.endswith('.parquet')
    ][:5]

dfs = []
```

```
for f in files:
    response = client.get_object(BUCKET, f)
    dfs.append(pd.read_parquet(BytesIO(response.read())))
    response.close()
    response.release_conn()

df = pd.concat(dfs)
print(f"Loaded {len(df)} rows for analysis.")
print(df.head())
```

3. The Golden Features (Requirement: 5+)

You need to select at least 5 features to satisfy the professor. Here is the strategy you will present.

3.1 1. User State: Purchase Power (Continuous)

- **Definition:** A score (0-10) indicating how rich/active the user is.
- **Hypothesis:** Higher purchase power users churn *less* and generate *higher* value.
- **Action:** Plot a Boxplot: `Purchase Power` vs `Total Reward`.

3.2 2. Temporal Context: Hour of Day (Cyclical)

- **Definition:** 0 to 23.
- **Hypothesis:** Users browsing at 2 AM might churn faster (doom scrolling) than users at 8 PM (shopping time).
- **Action:** Plot a Line Chart: `Hour` vs `Churn Rate`.

3.3 3. Behavioral Signal: Engagement Score (Derived)

- **Definition:** The sum of Clicks + Add-to-Carts on the current page.
- **Hypothesis:** High engagement on Page T predicts retention on Page $T + 1$.
- **Action:** Correlation Matrix.

3.4 4. Sequence Depth: Page ID (Integer)

- **Definition:** How deep are they in the feed? (Page 0, 1, 2...).
- **Hypothesis:** ”The Fatigue Effect.” Churn probability increases as Page ID increases.

3.5 5. Financial Signal: Current Page Value (Float)

- **Definition:** Total Yuan spent on the current page.
- **Hypothesis:** Spending money resets the ”Churn Clock.”

4. Generating Your Slides Evidence

Use this code to generate the 3 mandatory plots for your presentation.

4.1 Plot 1: The Churn Distribution (Class Imbalance)

Prove to the professor that we need accurate metrics (F1-score) because Churn is rare (or frequent).

```python
import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(6,4))
sns.countplot(x='done', data=df)
plt.title("Distribution of Churn Events (0=Stay, 1=Leave)")
plt.savefig("churn_dist.png")
```

4.2 Plot 2: The Correlation Heatmap (Feature Importance)

This is the most important chart. It proves which features matter.

```python
plt.figure(figsize=(10,8))
# Select numeric columns
cols = ['state_purchase_power', 'action_page_id', 'reward', 'done']
sns.heatmap(df[cols].corr(), annot=True, cmap='coolwarm')
plt.title("Feature Correlation Matrix")
plt.savefig("heatmap.png")
```

4.3 Plot 3: The ”Value” Analysis (Regression Target)

Show the distribution of Money (Reward).

```python
plt.figure(figsize=(10,4))
sns.histplot(df[df['reward'] > 0]['reward'], bins=50, kde=True)
plt.title("Distribution of Non-Zero Rewards (Session Value)")
plt.xlabel("Reward Value (Yuan)")
plt.savefig("value_dist.png")
```

5. Phase 1 Deliverables (Feb 9)

5.1 1. The Presentation

You need to create 5-7 slides covering:

1. **Problem Statement:** ”Redefining Retention in the Attention Economy.”
2. **Data Overview:** Alibaba RL Dataset (1.5M interactions).
3. **Feature Strategy:** Show the 5 features defined in Chapter 3.
4. **EDA Insights:** Show the Heatmap and Churn Distribution.
5. **Proposed Solution:** ”We will use a Hybrid Model: Regression for Value, Classification for Churn.”

5.2 2. The Hand-off to Engineers

Once you confirm the features, send this message to the group chat:

> ”Team, I have finished the EDA. **Confirmed Features for Training:**

- `state_purchase_power`
- `action_page_id`

Targets:

- `reward` (For Regression)
- `done` (For Classification)

I have verified the data in `processed-data` is clean. You may proceed with model training."

6. Phase 2: The Reinforcement Learning Agent (Feb 10 – Mar 2)

While the engineers build the individual models, **you build the Brain**. You must create the environment that uses their models to simulate a user session.

6.1 Your Task: Build 'src/lead_ds/alibaba_env.py'

You will write a Python class (inheriting from OpenAI Gym) that simulates the user.

Logic of the Environment:

1. **Step 1:** The Agent (System) shows a page.
2. **Step 2:** The Environment calls the **Classification Model** (Engineer B) to ask: "Did the user leave?"
3. **Step 3:** The Environment calls the **Regression Model** (Engineer A) to ask: "How much did they spend?"
4. **Step 4:** The Environment returns the Reward and Next State.

6.2 Why this gets extra points:

Most students just predict "Churn." You are creating a **Simulator** that optimizes for Value. This is "Advanced AI Engineering."

7. Phase 3: Final Delivery (Mar 3 – Mar 9)

Your final job is to synthesize everything into the Project Report.

7.1 The Final Report Structure

- **Introduction:** Your "Subscription" justification (from Phase 1).
- **Methodology:**
  - Engineering: Airflow/MinIO (Credit the Data Engineer).
  - Algorithms: Explain the "From Scratch" Gradient Descent (Credit the ML Engineers).
- **Results:** Compare the "Scratch" models vs Scikit-Learn (Benchmarking).
- **Conclusion:** How this system could save Alibaba millions in retained revenue.

You are the glue that holds the technical and business sides together. Good luck.