PHASE 2 – PREDICTIVE MODELING TASKS

Objective

Use the cleaned and validated RetailSmart datasets to develop, train, and evaluate predictive models that generate actionable business insights such as **customer churn prediction**, **customer lifetime value estimation**, and **marketing response modeling**.

---

1. Data Preparation and Integration

Goal: Combine relevant data sources to form a unified modeling dataset.

Tasks:

- Load cleaned data (data_cleaned/ files).
- Merge **customers**, **sales**, **marketing**, and **products** using keys customer_id and product_id.
- Aggregate transaction data to the customer level (e.g., total spend, avg order value, last purchase date).
- Include marketing variables such as average spend and conversion rate per channel.
- Validate row counts and null values in the merged dataset.
- Save as model_input.csv.

---

2. Feature Engineering

Goal: Create meaningful predictive variables.

Tasks:

- Derive **RFM features** (Recency, Frequency, Monetary).
- Create temporal features such as *days_since_last_order*, *tenure*, *month_of_last_purchase*.
- Encode categorical variables (payment_type, channel, category_english, state).
- Create customer-level metrics like *average order value*, *marketing engagement score*, *number of campaigns received*.
- Normalize or scale numerical features if needed.

---

3. Target Definition and Label Creation

Goal: Define what we’re predicting.

Options:

- **Churn Prediction:** Use churn_flag from customers.csv as the target.
- **CLV Prediction (optional):** Predict total_spent or Monetary as a continuous target.

- **Response Modeling (optional):** Predict conversions or response_rate from marketing data.

Tasks:

- Select the modeling objective.
- Confirm target column (churn_flag for classification or total_spent for regression).
- Balance the dataset if class imbalance exists (SMOTE or stratified sampling).

---

4. Train–Test Split and Baseline Model

Goal: Build and evaluate baseline predictive models.

Tasks:

- Split the dataset into training and test sets (e.g., 70 / 30 split).
- Train a **logistic regression** model for churn prediction (baseline).
- Evaluate performance using accuracy, precision, recall, F1, ROC-AUC.
- Document baseline results.

---

5. Advanced Models and Hyperparameter Tuning

Goal: Improve model performance using ensemble and tree-based methods.

Tasks:

- Train decision tree, random forest, gradient boosting (XGBoost / LightGBM) models.
- Tune hyperparameters using GridSearchCV or RandomizedSearchCV.
- Compare performance metrics with the baseline model.
- Select the best performing model.

---

6. Model Interpretation and Insights

Goal: Translate model outputs into business understanding.

Tasks:

- Identify top predictive features (feature importance plot).
- Interpret how each variable impacts churn or revenue.
- For churn models, explain profiles of likely-to-churn customers.
- For CLV or response models, interpret value drivers.

---

7. Model Evaluation on Test Data

Goal: Confirm model generalization.

Tasks:

- Evaluate on unseen test data.
- Generate confusion matrix and classification report.
- For regression models, compute RMSE, MAE, R^2.
- Compare train vs. test performance to detect overfitting.

---

8. Model Preservation and Documentation

Goal: Prepare the model for later deployment or reuse in MLOps (Phase 3).

Tasks:

- Save the final model (.pkl file using joblib / pickle).
- Export preprocessing pipeline and feature metadata.
- Document key insights, model parameters, and evaluation results.
- Store outputs under models/ and reports/ folders.