2024

Data Science Methodology

From System Design to Deployment

Alexander Guschin
Mikhail Rozhkov

OUTLINE

1. Problem Framing
2. Data and Feature Engineering
3. Modeling Techniques
4. Model Validation and Evaluation

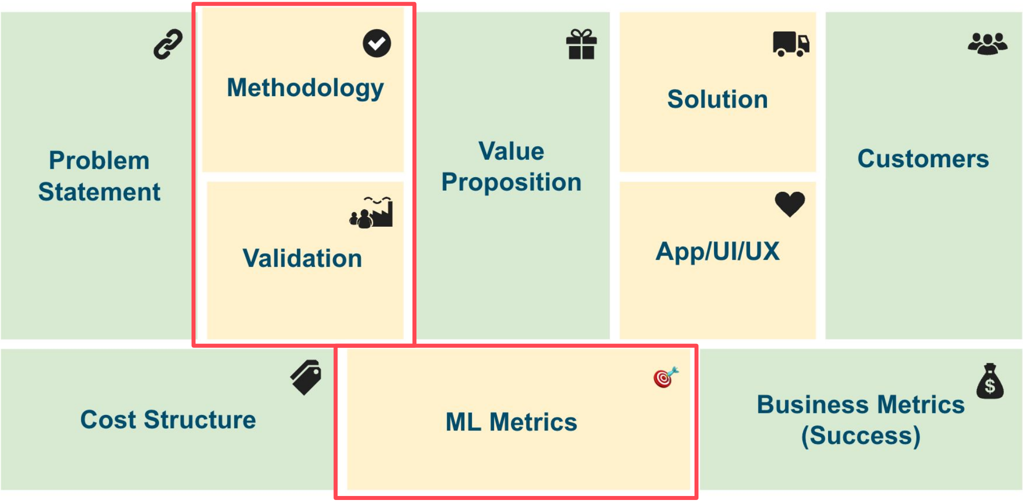
4.x – Data Science Methodology

Purpose

- Frame the ML problem

Guiding questions:

- How do we frame this as a machine learning problem? What ML metric should we optimize?



MODULE DETAILS

Goals

- Understand how to approach ML problems systematically
- Learn to make informed decisions about data and modeling techniques
- Develop skills to evaluate ML models effectively

Learning Outcomes

- Frame ML problems effectively
- Perform data and feature engineering
- Select appropriate modeling techniques
- Design robust evaluation frameworks

HANDS-ON ACTIVITIES

Exercise

- Create a clear understanding of ML task to solve to adhere to **Solution Design** created earlier
- Goal: Understand requirements for ML Pipeline

Output

1. Drafting the next part of a design document: **DS Methodology**
   a. Problem Framing and Approach
   b. Data and Feature Engineering
   c. Modeling Techniques and Algorithms
   d. Model Validation and Evaluation

Problem Framing and Approach

ML Product Design

> Guide: 4.1 - Problem Framing and Approach

4.1 – Problem Framing and Approach

How do we frame this as a machine learning problem?

Purpose

- To ensure the ML approach aligns with the business problem and leverages appropriate techniques.

Guiding questions:

- How do we frame this as a machine learning problem?
- What ML metric we should optimize?
- Why is this approach the most suitable?
- What is the simplest solution? Can we solve the problem without ML?
- What is a feasible baseline solution?

Case: NewPizza - long waiting time

How do we frame this as a machine learning problem?

Let’s recall business metrics we fixed for this task

Possible approaches:

- Regression?
- Classification?
- Unsupervised learning?

Given a business metrics to optimize, you can frame ML problem in different forms

Case: NewPizza - long waiting time

How do we frame this as a machine learning problem?

Let's recall business metrics we fixed for this task

Let's talk about baselines:

- Regression
- Classification

You can have no-ML baselines of various complexity

Case: shop queue detection

How do we frame this as a machine learning problem?

Possible approaches

- Object detection?
- Applying multi-modal LLM?
- Other ideas?

Your overall solution may consist from several ML problems/models

Case: shop queue detection

How do we frame this as a machine learning problem?

Let’s talk about baselines:

- Object detection
- Applying multi-modal LLM

Sometimes no-ML baseline for a chosen solution isn’t possible.
How can we get a baseline then?

Exercise: Problem Framing and Approach

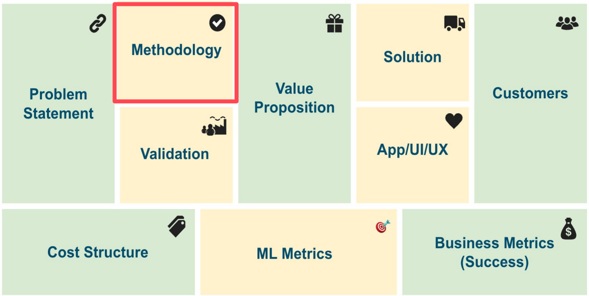
How does the solution look for our customers?

Group task:

- Brainstorm and complete the ML Product Design sections:
  - 4.1 - Problem Framing and Approach
- 5 min

Key points:

- ML problem type (e.g., classification, regression)
- Potential alternative approaches
- Baseline solution (without ML)



Practice

4.1 - Problem Framing and Approach

How do we frame this as a machine learning problem?

Overview:

- The business problem of inaccurate trip duration predictions is framed as a regression problem. The approach involves using historical trip data to train a model that predicts trip duration

Key points:

- Problem Type: Regression.
- Approach: Supervised learning with historical data.
- Baseline Solution: Current provider's predictions.

Example

ML Product Design: EasyRide Taxi

Problem Statement

- High MAPE > 30%.
- External provider's prediction service (we can't improve).
- Competitive market with accurate pricing as a differentiator.
- Critical to EasyRide's strategy of superior customer service.

Methodology

- Problem Type: Regression.
- Approach: Supervised learning.
- Metric: MAPE.
- Baseline: Current predictions.

Validation

- Validation Methodology: Shadow deployment, A/B testing.
- Pilot Scope: 1 week
- Success Criteria: Lower MAPE, higher booking rates.

Value Proposition

- Minimize revenue loss
- Improve customer retention
- Improve driver retention

Solution

- Target: trip duration in minutes
- Format: a float number
- Components: Taxi App, Data Ingestion, ML Solution, Backend

App/UI/UX

- UI: ETA, trip duration and cost/earnings.
- UX: Requesting a ride, viewing the predicted cost, booking the ride.

Customers

- Taxi app customers
- Taxi drivers

Cost Structure

- Initial Development: $214,083
- Annual Operations: $386,000
- Annual Benefit: $15,147,500
- ROI: First Year: 2,424% / Subsequent Years: 3,825%

Performance / ML Metrics

- Prediction accuracy (MAPE) < 15%
- Prediction latency < 100ms
- Business Metrics: Booking rate, Revenue Increase
- Timeline: Daily metric evaluation.

Business Metrics (Success)

- Daily Revenue Increase by $24,000
- Pricing Loss Reduction by $17,000
- Booking Rate Improvement by 6%
- Evaluate metrics daily, with quarterly reviews

Data and Feature Engineering

ML Product Design

› Guide: 4.2 - Data and Feature Engineering

Case: shop queue detection

What data do we need?

- Where can we find a dataset?
- How can we collect one?
- How can we label one?

Collecting and labeling can be expensive, but will deliver you high-quality data

- What is the size of the dataset you may need here?
- How to ensure you collected enough data?

4.2 – Data and Feature Engineering

What data do we need?

Purpose

- To ensure the ML system has access to high-quality, relevant data.

Guiding questions:

- What data will you use to train your model?
- What input data is needed during serving?
- How will we ensure data quality?
- How will you clean and prepare the data (e.g., excluding outliers) - consider important edge cases

Exercise: Data and Feature Engineering

What data do we need?

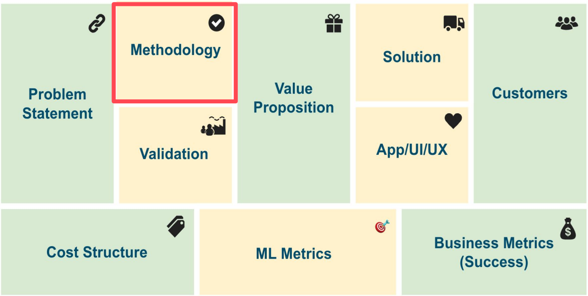
Group task:

- Brainstorm and complete the ML Product Design sections:
    - 4.1 - Problem Framing and Approach
- 5 min

Key points:

- Data sources and collection methods
- Data preprocessing and feature engineering
- Data quality assurance processes
- Data Labeling

Practice



4.2 – Data and Feature Engineering

What data do we need?

Overview:

- Data is sourced from the NY Taxi dataset, including features such as pickup and dropoff locations, trip distance, time of day, and day of the week.
- Data File format: Parquet

Key points:

- Data Sources: NY Taxi dataset (TLC Trip Record Data)

Data fields:

- **id** - a unique identifier for each trip
- **pickup_datetime** - date and time when the meter was engaged
- **dropoff_datetime** - date and time when the meter was disengaged
- **passenger_count** - the number of passengers in the vehicle (driver entered value)
- **pickup_longitude** - the longitude where the meter was engaged
- **pickup_latitude** - the latitude where the meter was engaged
- **dropoff_longitude** - the longitude where the meter was disengaged
- **dropoff_latitude** - the latitude where the meter was disengaged
- **trip_duration** - duration of the trip in seconds

Example

Modeling Techniques and Algorithms

ML Product Design

> Guide: 4.3 - Modeling Techniques and Algorithms

4.3 – Modeling Techniques and Algorithms

What it the best modelling approach?

Purpose

- To provide a clear understanding of the technical approach and its rationale

Guiding questions:

- Which ML algorithms are most suitable for our problem?
- How will we optimize model performance?
- What are the trade-offs between different modeling approaches?
- What feature engineering techniques you need to consider for selected ML model?

Case: NewPizza - long waiting time

What it the best modelling approach?

- Selected algorithms and rationale
  - Time-series models (frameworks?)
  - GBDT predicting the delta (frameworks?)
- Hyperparameter tuning strategy
  - Manual - try this first
  - Optuna - you still need to understand what hyperparameters matter

Case: shop queue detection

What it the best modelling approach?

- Selected algorithms and rationale
  - YOLO
  - Multi-modal LLM
- Model architecture details
  - ?
- Hyperparameter tuning strategy
  - ?

Exercise: 4.3 - Modeling Techniques

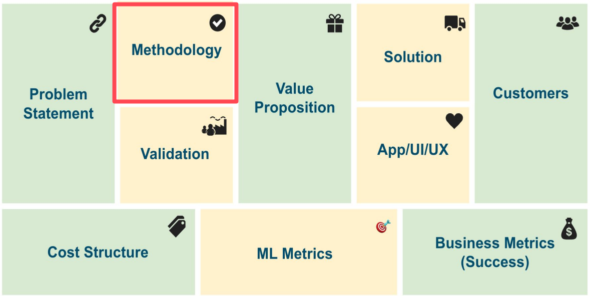
What it the best modelling approach?

Group task:

- Brainstorm and complete the ML Product Design sections:
  - 4.3 - Modeling Techniques and Algorithms
- 5 min

Key points:

- Selected algorithms and rationale
- Model architecture details (for DL)



Practice

4.3 – Modeling Techniques

What it the best modelling approach?

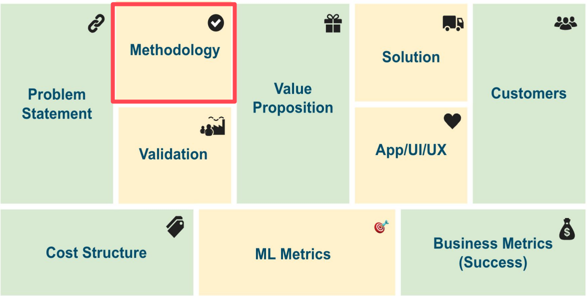
Overview:

- After evaluating various regression algorithms, Gradient Boosting Decision Trees (GBDT), specifically XGBoost, is chosen as the primary model for its balance of accuracy and computational efficiency in predicting trip durations.
- XGBoost is chosen for its:
  - Superior handling of non-linear relationships in geo-temporal data
  - Robustness to outliers common in urban traffic patterns
  - Ability to capture complex feature interactions
  - Scalability to large datasets typical in taxi operations
  - Balance between prediction accuracy and inference speed

Example

Key points:

- **Selected algorithm**: XGBoost
  Alternatives considered: Linear Regression, Random Forest



Model Validation and Evaluation Framework

ML Product Design

> Guide: 4.4 - Model Validation and Evaluation Framework

4.4 – Model Validation and Evaluation

How to ensure generalization and robustness?

Purpose

- To ensure the model's performance can be reliably measured and meets business requirements

Guiding questions:

- Which metrics do you need to calculate?
- How will we split our data to validate the model effectively?
- How will we ensure the evaluation process is unbiased and thorough?

Case: shop queue detection

How to ensure generalization and robustness?

What if we have just 100 samples?

- Cross-validation and data split:
  - K-fold
  - Holdout
  - or something else?

Case: NewPizza - long waiting time

How to ensure generalization and robustness?

- ML Metrics: Performance metrics specific to the validation and test phases. Evaluation metrics should be relevant to business metrics.
    - Classification:
        - Precision at Recall.
        - Other ideas?
    - Regression:
        - RMSE. Then what?

Exercise: Model Validation and Evaluation

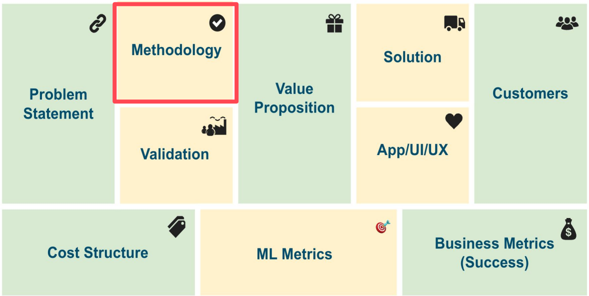
How to ensure generalization and robustness?

Group task:

- Brainstorm and complete the ML Product Design sections:
    - How to ensure generalization and robustness?
- 5 min

Key points:

- **Techniques**: Cross-validation (e.g., k-fold cross-validation), holdout validation, stratified sampling.
- **Data Splits**: Training set, validation set, and test set.
- **Metrics**: Performance metrics specific to the validation and test phases. Evaluation metrics should be relevant to business metrics.



Practice

4.4 – Model Validation and Evaluation

How to ensure generalization and robustness?

Overview:

- The model's performance is rigorously validated using a combination of time-based cross-validation and geospatial holdout validation, with evaluation metrics directly tied to business impact.
- Time-based Cross-validation:
  - Weekly data chunks
  - Rolling window approach
- Geospatial Holdout (suitable if we want to introduce new city/neighbourhood):
  - Reserve specific NYC neighborhoods
  - Test model generalization

Key points:

- MAPE: Target <15% (Current: 30%)
- RMSLE: Penalize underestimation
- Latency: <500ms response time
- Time-based split for cross-val

Example

Assignment

Start with ML System Design!

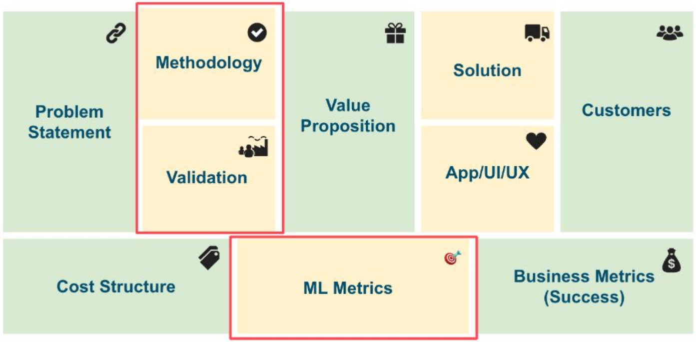
Prerequisites

- Draft of the ML Product Design: Business Understanding
- Draft of the ML Product Design: Solution

Practice: Methodology

If … else … LLM …

- Frame DS methodology for your project
- Follow the guide to describe each section
- Summarise Methodology blocks on the canvas
- Update Cost Structure & Solution (if needed)



Materials & Links

- Course Materials: Google Drive
- Practice - EasyRide Taxi - Day 3 - PUBLIC
- Guide - ML System Design - Canvas