ML

30 сентября 2025 г.

Содержание

1 Ссылка на роадмапу

РОАДМАПА НАХОДИТСЯ ПО ССЫЛКЕ

2 Данные, EDA и метрики

2.1 EDA и визуализация

Описательные статистики: среднее, медиана, квантили, дисперсия, асимметрия, эксцесс.

Визуализация: гистограммы, KDE, box/violin plots, scatter/hexbin, парные графики.

Корреляции и ассоциации: Пирсон, Спирмен, Кендалл; для категорий — Cramér's V, Theil's U.

Анализ пропусков и выбросов; оценка утечек признаков (data leakage).

Профилирование датасетов и репорты качества.

2.2 Предобработка и фичи

Очистка: обработка пропусков (simple/mICE), дубликатов, опечаток.

Масштабирование/нормализация: Standard/MinMax/Robust; преобразования (Box–Cox, Yeo–Johnson).

Кодирование категорий: one-hot, target/mean encoding, ordinal, hashing.

Обработка выбросов: IQR/percentile clipping, *robust* модели.

Выбор признаков: фильтрационные методы (ANOVA, χ^2), wrapper/embedded (RFE, L1).

Создание признаков: лаги, агрегаты по группам, datetime/текст/гео-инженерия.

2.3 Оценка и валидация

Разбиения: Train/Validation/Test; k-fold, stratified, group, time-series split.

Метрики регрессии: RMSE, MAE, R^2, MAPE/sMAPE, pinball loss (квантильная регрессия).

Метрики классификации: accuracy, precision/recall/F1, ROC–AUC, PR–AUC, log-loss, Brier score.

Метрики ранжирования: NDCG-k, MAP-k, Recall-k, MRR.

Калибровка вероятностей: Platt scaling, isotonic regression.

Устойчивость: повторяемость, доверительные интервалы (bootstrap), пермутационные тесты.

3 Классический ML

3.1 Supervised

Линейные модели: линейная/логистическая регрессия, регуляризация (Ridge/Lasso/ElasticNet).

SVM: линейный и ядровый (RBF, polynomial), выбор C, γ.

Деревья решений и ансамбли: CART, Random Forest, ExtraTrees.

Градиентный бустинг: XGBoost, LightGBM, CatBoost (в т.ч. для категориальных).

K-NN для базовых задач, Naive Bayes для текстов.

Трюки: обработка дисбаланса (class weights, SMOTE), ранняя остановка, подбор целевых трансформаций.

3.2 Unsupervised

Кластеризация: K-Means/K-Medoids, иерархическая (Ward/complete), DBSCAN/HDBSCAN, Gaussian Mixture Models.

Снижение размерности: PCA/ICA, t-SNE/UMAP, автоэнкодеры (линию с DL).

Оценка кластеров: silhouette, Davies–Bouldin, Calinski–Harabasz; стабильность кластеров.

Поиск выбросов: Isolation Forest, LOF, One-Class SVM.

3.3 Time Series (база)

Базовые подходы и эвристики: Naive, Seasonal Naive, скользящее среднее; backtesting со скользящим/расширяющимся окном, rolling origin.

Декомпозиция рядов: классическая и STL; выделение тренда, сезонности и остатков.

Стационарность и преобразования: ADF/KPSS, дифференцирование, лог/Box–Cox/Yeo–Johnson.

ACF/PACF и выбор порядков; диагностика остатков (белый шум, Ljung–Box), интервалы прогноза.

Экспоненциальное сглаживание (ETS): простое, Хольта, Хольта–Винтерса.

Классические модели: AR/MA/ARMA, ARIMA/SARIMA, ARIMAX/SARIMAX.

Мультивариантные: VAR/VARMAX (для взаимосвязанных рядов).

Прототипирование: Prophet (регрессоры, праздники, смена тренда).

Стратегии многошагового прогноза: recursive vs. direct vs. multi-output; walk-forward валидация.

Метрики: RMSE/MAE/MAPE/sMAPE, MASE, pinball loss для вероятностного прогноза.

3.4 Подбор и регуляризация

Поиск: Grid/Random search, Байесовская оптимизация (TPE, Gaussian Process), Hyperband/BOHB, Optuna sampler'ы.

Энсемблирование: bagging, boosting, stacking/blending; калибровка.

Регуляризация: L1/L2, ранняя остановка, dropout/weight decay (перекличка с DL).

3.5 Эксперименты

A/B/n-тестирование: рандомизация, стратификация, power analysis.

Статистические тесты: t-test/ANOVA, χ^2, Манна–Уитни/Уилкоксона.

Непараметрика: bootstrap, пермутационные тесты.

Подводные камни: p-hacking, множественные сравнения, сезонность/каннибализация.

3.6 Интерпретация

Важности: impurity/permutation importance.

Локальная/глобальная: LIME, SHAP (Tree/Kernel/Deep), ALE/ICE, PDP.

Контрафактуальные объяснения, частичная зависимость от признаков.

4 Глубокое обучение

4.1 DL основа

Автодифференцирование и backprop; инициализации (Xavier/He).

Оптимизация: SGD/momentum, RMSProp, Adam/AdamW, LAMB; клиппинг градиента.

Регуляризация и нормализации: dropout, batch/layer/group norm.

Режимы обучения: mixed precision, lr-schedulers (step, cosine, OneCycle), ранняя остановка.

Фреймворки: PyTorch, TensorFlow/Keras; экосистема (Lightning, timm, torchvision/torchtext/torc

4.2 Компьютерное зрение

Архитектуры: LeNet/VGG, ResNet/EfficientNet/ConvNeXt, Vision Transformers (ViT/DeiT/Swin).

Задачи: классификация, детекция (Faster R-CNN, RetinaNet, YOLO), сегментация (U-Net, DeepLab), keypoints/pose.

Аугментации: flips/crops, color jitter, CutOut, MixUp/CutMix, RandAugment.

Лоссы/метрики: CE, focal, dice/IoU; mAP, mIoU.

Transfer learning, fine-tuning, distillation.

4.3 NLP

Токенизация: WordPiece/BPE, sentencepiece; нормализация/стемминг/лемматизация.

Эмбеддинги: word2vec/GloVe/fastText; контекстные (ELMo, BERT family).

Модели: RNN/LSTM/GRU, Attention, Transformers (BERT/RoBERTa/DeBERTa, GPT, T5/LongT5).

Задачи: классификация, NER, QA, суммаризация, машинный перевод, retrieval.

Трюки: fine-tuning, adapters/LoRA/QLoRA, prompt-инжиниринг, оценка (BLEU/ROUGE/F1).

4.4 Generative / LLM

Глубокие генеративные модели: VAE, GAN (основы), диффузионные модели (DDPM/DDIM, Stable Diffusion).

LLM-практика: инструкции/запросы, few-shot/контекстуализация, RAG (ретривер, индексация, чанкинг, перезапрос).

Тонкая настройка: полное дообучение, PEFT/LoRA, DPO/RLHF (высокоуровневые принципы), безопасность/оценка токсичности.

Оптимизация инференса: квантование, sparsity, компиляция (ONNX Runtime, TensorRT).

4.5 Графовые сети

Базовые идеи: message passing, агрегаторы, over-smoothing.

Архитектуры: GCN, GraphSAGE, GAT, GIN.

Задачи: классификация вершин/рёбер, линк-предикшн, классификация графов.

Фреймворки: PyTorch Geometric, DGL.

4.6 DL для табличных

Архитектуры: Wide&Deep, DeepFM, TabNet, NODE, FT-Transformer/TabTransformer.

Когда применять: большие данные/мультимодальные фичи/совместно с GBM; сравнение с CatBoost/LightGBM.

5 Направления

5.1 Рекомендательные системы

Базовые подходы: user/item-based CF, матричная факторизация (ALS, BPR).

Ранжирование: pointwise/pairwise/listwise; LambdaMART/XGBoost/LightGBM.

Нейронные: two-tower/DSSM, sequence models (GRU4Rec), DIN/DIEN, deep retrieval + re-ranking.

Метрики: Recall/NDCG/MAP@k, diversity/coverage, calibration.

Практика: холодный старт, анти-спам, отклик/контрафакты, bandits для explore/exploit.

5.2 Причинно-следственный анализ

Модели потенциальных исходов: ATE/CATE; S-/T-/X-learner, uplift trees.

Графовый подход: DAG, do-calculus, критерий d-separation.

Идентификация: matching/propensity score, IPW, IV (инструментальные переменные), DiD.

В полях: влияние сезонности, интерференция, зависимости времени.

5.3 Байесовский анализ

Байесовский вывод: априорные/апостериорные, сопряжённые пары.

MCMC: Metropolis–Hastings, Gibbs, HMC/NUTS; диагностика сходимости.

Вариационный вывод: mean-field, BBVI; стох. вариационный вывод.

Гауссовские процессы: ядра (RBF/Matern), регрессия/классификация.

Инструменты: PyMC, NumPyro, Stan.

5.4 Reinforcement Learning

Бандиты и исследование: ε-жадный, UCB1/UCT, градиентный бандит, Томпсоновское сэмплирование; метрики (средняя награда, доля оптимального действия).

Value-based и TD: уравнения Беллмана; TD(0), TD(λ), SARSA/Expected SARSA, Q-learning; аппроксимация функций ценности.

DQN и улучшения: experience replay, target network; Double DQN, Dueling, Prioritized Replay (PER), NoisyNets, n-step returns, distributional (C51/QR-DQN), Rainbow.

Policy gradient & actor–critic: REINFORCE, baseline/advantage; A2C/A3C, PPO/TRPO, GAE, энтропийная регуляризация.

Непрерывные действия: DDPG, TD3 (target policy smoothing, delayed updates), SAC (энтропийный контроль).

Модельно-ориентированное и имитационное: Dyna-Q, MBRL/планирование, MPC; имитационное обучение (behavior cloning, DAgger).

Практика и стабильность: нормализация наблюдений/награды, клиппинг градиента, выбор сидов, оценка по среднему возврату, sample efficiency.

Экосистема: Gym/Gymnasium, Stable Baselines3, RLlib, CleanRL; вёрпперы, логирование и визуализация обучения.

5.5 Time Series (продвинутые)

Пространство состояний и фильтрация: Калман (KF/UKF/EKF), LDS, HMM; сглаживание/фильтрация, оценка неопределённости.

Мультивариантные и коинтеграция: VAR/VARMAX, VECM; тесты Йохансена; межсерийные связи и иерархические модели (MinT reconciliation).

Прототипирование и регрессоры: Prophet (регрессоры, праздники, разрывы тренда), внешние факторы/события.

DL-подходы: Seq2Seq/Attention, N-BEATS, Temporal CNN/TCN, Temporal Fusion Transformer (TFT), Informer; глобальные модели для множеств рядов.

Аномалии и изменения: change-point detection (CUSUM, Bayesian Online), аномалии (изол. лес, автоэнкодеры), доверительные и предсказательные интервалы, калибровка.

Вероятностный прогноз: квантильная регрессия, pinball loss, конформное предсказание.

Специальные случаи: редкий спрос (Croston/TSB), интермиттирующие ряды, календарные/праздничные эффекты.

6 MLOps

6.1 DataOps и качество данных

Версионирование: DVC/LakeFS, контроль схем (pydantic/cerberus).

Валидация: Great Expectations, Deequ; профилирование и тесты данных.

Каталоги/линейки: DataHub/Amundsen; lineage и доступы.

6.2 Эксперименты и трекинг

Трекинг: MLflow, Weights&Biases, Neptune; логирование метрик/артефактов.

Репродуцируемость: фиксирование seed, зависимостей (conda/poetry), Docker.

Автоматизация отчётов и сравнение запусков.

6.3 AutoML & HPO

Пакеты AutoML: auto-sklearn, AutoGluon, FLAML, H2O AutoML, TPOT, AutoKeras.

HPO-фреймворки: Optuna, Ray Tune, NNI, Hyperopt; Hyperband/BOHB, Population Based Training.

NAS/мета-обучение: поиск архитектур, warm-start из метаданных.

Практика: определение целевой метрики/ограничений (latency/cost), контроль утечек, справедливость.

6.4 Пайплайны

ML-пайплайны: sklearn Pipelines/ColumnTransformer, feature engineering как шаги.

Оркестрация: Airflow, Prefect, Dagster; задачи, зависимости, ретраи.

Тестирование: unit/интеграционные, data tests; CI для пайплайнов.

6.5 Деплой и сервинг

Сервинг: FastAPI/gRPC, BentoML, TorchServe, TF Serving, Triton Inference Server.

Пакетирование: Docker/OCI; развёртывание на K8s (HPA, Istio), serverless варианты.

Стратегии: batch/online/streaming; canary/shadow/A/B; кэширование и фичевая консистентность.

Производительность: ONNX, TorchScript, TensorRT; мониторинг латентности/ресурсов.

6.6 Мониторинг

Дрифт данных/концепции: PSI/JS/KS, коридоры метрик, алерты.

Мониторинг качества: регрессионный контроль, пост-лейблинг, активное обучение.

Наблюдаемость: логи/метрики/трейсинг; дашборды.

Модельные карточки, аудит/этика, управление рисками.

6.7 Feature Store

Консистентность online/offline, point-in-time корректность.

Инструменты: Feast (open-source) и аналоги; материализация, слои хранения.

Схемы: каталог, контроль качества признаков.

6.8 Большие данные

Фреймворки: Apache Spark (SQL/MLlib/Structured Streaming), Dask, Ray.

Хранилища и форматы: Parquet/ORC/Delta; lakehouse-паттерны.

Потоки: Kafka/Pulsar; микробатчи/стриминг.

6.9 Облака

Объектное хранилище и IAM: AWS S3/STS, GCP GCS/IAM, Azure Blob/AAD.

ML-платформы: SageMaker, Vertex AI, Azure ML; реестры моделей/артефактов.

Стоимость/безопасность: квоты, бюджетирование, приватность данных, комплаенс.

7 Пет-проекты

7.1 Кредитный скоринг

1. Кредитный скоринг Стоит ли давать кредит— довольно популярная задача и отличный выбор для новчиков, чтобы самостоятельно проделать все этапы. Сначала берем любой датасет на kaggle по запросу Credit Scoring. Проводим EDA, генерируем гипотезы, фичи, готовим данные для модели и делаем бейзлайн: логистическая регрессия. Затем уже можно попробовать случайный лес, градиентный бустинг, KNN или еще что по вкусу— сравниваем метрики. И на последок не забываем проанализировать результаты и культурно презентовать. Можно провести AB тест на смой первой модели. Все варианты решения и реализации можно найти в интернетах: GitHub, Хабр. Очень полезным будет посмотреть всякие выступления на конференциях по этой теме для вдохновения, да и это очень поможет на мл кейсах.

7.2 Наивный Байесовский классификатор (НБК)

2. Наивный Байесовский классификатор (НБК) Для конкретики будем классифицировать письма на спам. Опять же обработаем данные: удаляем числа, знаки препинания, стоп-слова, стемминги, лемматизацию. Объединяем все методы предварительной обработки и создаём словарь слов и счётчик каждого слова в наборе данных для обучения: 1. Вычисляем вероятность для каждого слова в тексте и отфильтровываем слова со значением вероятности меньше порогового. Такие слова будут нерелевантными. 2. Для каждого слова в словаре создаём вероятность, что это слово окажется в спаме. Определяем условную вероятность для использования её в НБК. 3. Вычисляем прогнозируемый результат с помощью условных вероятностей. НБК реализовать не сложно. Куда интересней погрузиться во всю теорию, которая за этим стоит, в вероятностные модели. К тому же, кейс фильтрации спама и подобного часто встречается на собесах.

7.3 MLOps

3. MLOps Можно наладить какой-то минимальный прод для проектов: например телеграм бот или FastAPI. Можно еще автоматизировать пайплайн с помощь AirFlow и попробовать запустить инфраструктуру не только локально, но и облаке. Конечно нужно будет поизучать Docker, Cuber, Hadoop, Spark, HDFS, Kafka. Но на самом деле ничего трудного— после нашего курса дата инженера будете делать такие вещи по щелчку пальцев.

7.4 Ранжирование и матчинг

4. Ранжирование и матчинг Для начала лучше пробежаться глазами по статье и посмотреть, что пишут в интернетах. Можно выделить три подхода к задаче: поточечный, попарный, списочный. Советую начать с первого как самого простого. Для конкретики будем предсказать оценку релевантности для запросов тестового датасета. Здесь можно кстати поучиться парсить web-страниц и собирать сырые данные, размечать их с помощью какого-нибудь Яндекс-Толока. Делаем регрессию, а затем Random Forest Regressor, XGBoost, lightGBM, CatBoost. Совсем продвинутые могут

попробовать языковые модели в духе FastText, Word2Vec, DSSM и более сложные: BERT, можно даже попробовать архитектуру трансформеров.

7.5 Рекомендашки

5. Рекомендашки Очень популярный кейс на собесах. Для начала лучше пробежаться глазами по этому разделу и посмотреть, что пишут в интернетах. Затем начинаем реализовывать самое простое как бейзлайн, например, content-based рекомендации, KNN. Дальше можно попробовать факторизации матрицы рейтингов по svd разложению или по более эффективной als архитектуре и функции ошибок bpr. Затем можно попробовать W2V подход, чтобы использовать последовательность взаимодействий пользователя для построения рекомендации следующего предмета. Для знатоков DL можно попробовать DSSM, SasRec/Bert4Rec, MultVAE, Merlin или графовые нейронки: GCN-подобные архитектуры. Также стоит попробовать обучение с подкреплением: многоруких бандитов. Ну и конечно рекомендательные системы можно попробовать рассмотреть как задачу ранжирования.