АО «АЛМАТИНСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет инженерных и информационных технологий

Кафедра компьютерной инженерии

ПРОЕКТИРОВАНИЕ СИСТЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

ЛЕКЦИЯ № 11

Объяснимый и ответственный искусственный интеллект

Теоретико-практический конспект

Составитель: Алтыбай А.

Кафедра компьютерной инженерии

Алматы, 2026

Аннотация и назначение лекции

Лекция 11 раскрывает тему «Объяснимый и ответственный искусственный интеллект» в контексте разработки промышленной AI-системы. Материал опирается на тематический план дисциплины и соединяет определения, принципы проектирования, математические оценки, проектные артефакты, практический код и аргументированный анализ рисков. Вопросы построены вокруг решений, которые предстоит принимать инженеру, а не только вокруг названий алгоритмов. [1, 2]

План занятия

1. Explainability и interpretability
2. Global и local explanations
3. Пермутационная важность
4. SHAP
5. Bias и fairness
6. Источники смещения
7. Privacy by design
8. Модельные документы
9. Практические сценарии, контроль качества и проектные задания.

Ожидаемые результаты обучения

По завершении студент способен:

1. объяснить и аргументированно применить принцип «Explainability и interpretability» к инженерному решению;
2. объяснить и аргументированно применить принцип «Global и local explanations» к инженерному решению;

3. объяснить и аргументированно применить принцип «Пермутационная важность» к инженерному решению;
4. объяснить и аргументированно применить принцип «SHAP» к инженерному решению;
5. объяснить и аргументированно применить принцип «Bias и fairness» к инженерному решению;
6. объяснить и аргументированно применить принцип «Источники смещения» к инженерному решению;
7. реализовать упрощённый проверяемый прототип и сформулировать критерии его приемки.
8. указать угрозы достоверности результатов и предложить проверки до эксплуатации.

Сквозной кейс. В течение лекции рассматриваем систему ранжирования поступающих заявок. Существенные условия: смещение по подгруппам, необъяснимые решения. Основная проверяемая характеристика: TPR gap и процедуру обжалования. Числа в расчётных примерах учебные и не являются результатами реальных измерений.

1. Базовые теоретические положения из конспекта курса

Ниже представлены исходные положения учебно-методического конспекта, связанные с данной темой. Следующие разделы раскрывают их с инженерными деталями и проверяемыми примерами. [1, 2] Цель лекции. рассмотреть explainability, fairness, privacy, governance и риск-ориентированный подход к ответственному использованию AI.

Ключевые понятия. interpretability, explainability, SHAP, feature importance, fairness, bias, privacy, model card, data sheet, governance, risk management.

1.1 Интерпретируемость и объяснимость

Интерпретируемая модель имеет понятную внутреннюю структуру, например небольшое дерево решений. Explainability описывает методы, которые помогают объяснить поведение сложной модели после обучения. Важно не путать объяснение с доказательством причинности.

Локальное объяснение отвечает, почему модель выдала конкретный прогноз; глобальное — какие закономерности характерны для модели в целом.

1.2 Feature importance и SHAP

Feature importance может быть встроенной (например, у дерева) или полученной через permutation. SHAP использует идеи значений Шепли и оценивает вклад признаков в отклонение конкретного прогноза от базового уровня.

Объяснения нужно проверять на устойчивость и использовать осторожно при коррелированных признаках. Пользовательское объяснение также должно соответствовать роли: инженеру нужен технический анализ, конечному пользователю — понятная причина и способ оспаривания решения.

1.3 Bias и fairness

Смещение может появляться на этапе сбора данных, разметки, выбора цели, модели или deployment. Fairness-метрики зависят от контекста: demographic parity, equal opportunity, equalized odds и другие критерии могут конфликтовать.

Поэтому Responsible AI не сводится к одному числу. Требуется описать защищаемые группы, возможный вред, правовой/организационный контекст и процедуру human oversight.

1.4 Privacy и минимизация данных

AI-система должна собирать только необходимые данные, контролировать доступ и исключать случайное раскрытие чувствительной информации. Для LLM-систем особенно важно не отправлять конфиденциальные данные внешнему провайдеру без разрешения и соответствующей политики.

Меры могут включать псевдонимизацию, шифрование, role-based access, retention policy и аудит.

1.5 Governance и документация

Governance определяет, кто утверждает модель, кто отвечает за данные, какие проверки обязательны перед production и когда модель следует остановить. Полезные документы: model card, datasheet, risk register, evaluation report, change log и incident response plan.

Риск-ориентированный подход означает, что глубина проверки должна соответствовать потенциальному ущербу.

1.6 Выводы

▷ Explainability помогает анализировать поведение, но не доказывает причинность.

▷ Fairness зависит от контекста и не сводится к одной универсальной метрике.

▷ Privacy-by-design следует применять уже при сборе данных.

▷ Governance задает ответственность и правила допуска AI-системы к эксплуатации.

1.7 Контрольные вопросы

▷ В чем различие интерпретируемости и объяснимости?

▷ Что показывает SHAP-значение?

- Почему fairness-метрики могут конфликтовать?
- Какие документы полезны для AI governance?
- Что означает принцип минимизации данных?

2. Систематическое изложение темы и инженерные решения

2.1 Explainability и interpretability

Некоторые модели внутренне прозрачны, другие требуют пост-хок объяснения. Объяснение отдельного прогноза не является доказательством причинности. [2]

Проверяемый пример. Сравните коэффициенты логистической регрессии и локальное объяснение для градиентного бустинга.

Вопрос для обсуждения. Как проверить выполнение такого требования: Сравните коэффициенты логистической регрессии и локальное объяснение для градиентного бустинга.

2.2 Global и local explanations

Global объяснение описывает поведение модели в целом; local относится к конкретному примеру. Эти понятия могут давать разные ответы. [3]

Проверяемый пример. Покажите, как важность возраста глобально отличается от его вклада для одного клиента.

Вопрос для обсуждения. Как проверить выполнение такого требования: Покажите, как важность возраста глобально отличается от его вклада для одного клиента.

2.3 Пермутационная важность

Перемешивание значения признака оценивает изменение качества; коррелированные признаки затрудняют интерпретацию. Перестановки выполняют на validation/test. [4]

Проверяемый пример. Сравните снижение F1 после перестановки двух сильно коррелирующих признаков.

Вопрос для обсуждения. Как проверить выполнение такого требования: Сравните снижение F1 после перестановки двух сильно коррелирующих признаков.

2.4 SHAP

SHAP распределяет вклад относительно выбранного базового значения и связан с идеями Шепли. Сумма локальных вкладов отражает отклонение модельного выхода от baseline в выбранном представлении. [5]

Проверяемый пример. Объясните, почему нельзя интерпретировать знак SHAP как причинное влияние фактора.

Вопрос для обсуждения. Как проверить выполнение такого требования: Объясните, почему нельзя интерпретировать знак SHAP как причинное влияние фактора.

2.5 Bias и fairness

Различные групповые метрики могут конфликтовать между собой; выбор понятия справедливости является нормативным решением. Для малых групп оценки нестабильны. [6]

Проверяемый пример. Рассчитайте difference in TPR между двумя подгруппами и обсудите статистическую неопределенность.

Вопрос для обсуждения. Как проверить выполнение такого требования:

Рассчитайте difference in TPR между двумя подгруппами и обсудите статистическую неопределенность.

2.6 Источники смещения

Selection, measurement, label и deployment bias возникают на разных стадиях. Коррекция модели не заменяет исправление несправедливого процесса сбора. [1]

Проверяемый пример. Исследуйте, почему заявки без доступа к интернету отсутствуют в обучающих данных.

Вопрос для обсуждения. Как проверить выполнение такого требования: Исследуйте, почему заявки без доступа к интернету отсутствуют в обучающих данных.

2.7 Privacy by design

Минимизация данных, контроль доступа, retention и ограничение целей применяются до обучения. Удаление колонки имени не гарантирует анонимности. [2]

Проверяемый пример. Проверьте возможность повторной идентификации клиента по сочетанию косвенных признаков.

Вопрос для обсуждения. Как проверить выполнение такого требования: Проверьте возможность повторной идентификации клиента по сочетанию косвенных признаков.

2.8 Модельные документы

Model cards и datasheets описывают назначение, данные, оценки и ограничения, обеспечивая возможность аудита. Документация должна быть привязана к версиям. [3]

Проверяемый пример. Составьте карточку для скоринговой модели с таблицей метрик по группам.

Вопрос для обсуждения. Как проверить выполнение такого требования: Составьте карточку для скоринговой модели с таблицей метрик по группам.

2.9 Устойчивость

Модель может давать разные ответы при небольших изменениях входа, смене домена или наличии шума. Robustness-тесты должны соответствовать реальным угрозам. [4]

Проверяемый пример. Проверьте влияние опечаток и смены регистра на классификатор обращений.

Вопрос для обсуждения. Как проверить выполнение такого требования: Проверьте влияние опечаток и смены регистра на классификатор обращений.

2.10 Human oversight

Автоматизация должна иметь понятные правила проверки, обжалования и эскалации. Особое внимание требуется решениям с существенными последствиями. [5]

Проверяемый пример. Опишите процесс отмены ошибочной блокировки аккаунта.

Вопрос для обсуждения. Как проверить выполнение такого требования: Опишите процесс отмены ошибочной блокировки аккаунта.

2.11 Risk assessment

Оценка риска связывает вероятность, тяжесть ущерба и контролирующие меры. Нулевой риск недостижим, остаточный риск документируют и принимают уполномоченные лица. [6]

Проверяемый пример. Создайте реестр из пяти рисков и назначьте owners и способы мониторинга.

Вопрос для обсуждения. Как проверить выполнение такого требования: Создайте реестр из пяти рисков и назначьте owners и способы мониторинга.

2.12 Governance and audit

Управление включает политику данных, модельных изменений, журнал решений и независимую проверку. Объяснимость — лишь часть подотчетной AI-системы. [1]

Проверяемый пример. Свяжите отдельное решение API с конкретной model card, version, input schema и датой аудита.

Вопрос для обсуждения. Как проверить выполнение такого требования: Свяжите отдельное решение API с конкретной model card, version, input schema и датой аудита.

3. Математические инструменты и количественные критерии

Для перехода от неформального решения к проверяемой постановке введём следующие величины.

$$\text{DPGap} = \left| P(\hat{Y} = 1 \mid A = 0) - P(\hat{Y} = 1 \mid A = 1) \right| \tag{1}$$

$$\text{TPRGap} = \left| P(\hat{Y} = 1 \mid Y = 1, A = 0) - P(\hat{Y} = 1 \mid Y = 1, A = 1) \right| \tag{2}$$

Эти показатели описывают разные критерии справедливости. Они могут быть несовместимы, и не являются универсальной юридической оценкой.

При выборе числового критерия нужно отличать обучающую функцию потерь, продуктовую ценность и эксплуатационные ограничения. Первую можно оптимизировать дифференцированием, вторая определяется процессом принятия решения, а третьи проверяются нагрузочными испытаниями, политиками безопасности и мониторингом.

3.1 Пошаговый расчёт на учебных данных

Рассмотрим две конфигурации одной системы. Вариант A имеет условную оценку качества $q_A = 0.86$ и задержку $t_A = 90$ мс; вариант B — $q_B = 0.89$ и $t_B = 280$ мс. Предположим, что требование заказчика задаёт $q \geq 0.85$ и $p95 \leq 150$ мс. Тогда A удовлетворяет обоим условиям, тогда как B нарушает второе. Следовательно, выбор B только потому, что его offline-метрика выше, не является обоснованным. Числа иллюстративны; реальный вывод требует измерений при одинаковой нагрузке и одинаковом наборе данных.

$$\text{accept}(A) = \mathbf{1}\{0.86 \geq 0.85 \wedge 90 \leq 150\} = 1,$$
$$\text{accept}(B) = \mathbf{1}\{0.89 \geq 0.85 \wedge 280 \leq 150\} = 0.$$

3.2 Как интерпретировать число, а не только вычислить его

Сначала следует проверить единицы измерения, состав выборки и смысл агрегирования. Например, p95 на спокойном стенде нельзя переносить на рабочую нагрузку с иным числом одновременных запросов. Затем исследуют чувствительность к изменениям входных данных и неопределённость оценок. Наконец, фиксируют последствия порога: какую ошибку он предотвращает, а какую может увеличить. Этот подход делает количественный критерий частью инженерной аргументации, а не украшением отчёта. [1, 2]

4. Практическое программирование на Python

Пример предназначен для демонстрации ключевого инженерного принципа. Он не претендует на обработку реального production-трафика. Запускать код следует в изолированном окружении и фиксировать версии библиотек, если они используются. Вывод зависит от указанных входных данных; для стохастических алгоритмов он может меняться.

4.1 Основной воспроизводимый пример

```python
import numpy as np
from sklearn.metrics import confusion_matrix

y_true = np.array([0, 0, 1, 1, 1, 0, 1, 0])
y_pred = np.array([0, 1, 1, 0, 1, 0, 1, 0])
group = np.array([0, 0, 0, 0, 1, 1, 1, 1])
for g in (0, 1):
    ix = group == g
    cm = confusion_matrix(y_true[ix], y_pred[ix], labels=[0, 1])
    tn, fp, fn, tp = cm.ravel()
    tpr = tp / (tp + fn) if tp + fn else float("nan")
    print("group:", g, "TPR:", tpr)
# Inspect counts and uncertainty before using fairness conclusions.
```

Проверка результатов. Сначала убедитесь, что код завершился без исключений; проверьте формы массивов, схему результата и отсутствие недопустимых значений. Если используются внешние данные, включите в отчёт способ их получения и идентификатор версии. Измените один входной параметр и объясните, почему наблюдаемый результат изменился или остался прежним.

4.2 Компактная проверка контракта или метрики

```python
def safe_explanation(note, model_version):
    return {"explanation": note, "model_version": model_version,
            "warning": "Non-causal local explanation"}
```

```
print(safe_explanation("feature contributions", "v1"))
```

При самостоятельном повторении предложите минимум один негативный тест: некорректное значение, пустой вход, несовместимую версию схемы либо нарушение SLO. Переход от успешного демонстрационного запуска к надежной системе требует именно таких проверок.

5. Разбор трёх инженерных ситуаций

5.1 Ситуация 1: Explainability и interpretability

Заказчик использует систему ранжирования поступающих заявок. Новое требование заставляет рассмотреть принцип «Explainability и interpretability». Основная неопределённость состоит в том, как одновременно обеспечить TPR gap и процедуру обжалования, не допустить необъяснимые решения и сохранить возможность изменения модели.

Постановка. Некоторые модели внутренне прозрачны, другие требуют пост-хок объяснения. Объяснение отдельного прогноза не является доказательством причинности.

Действие инженера. Сравните коэффициенты логистической регрессии и локальное объяснение для градиентного бустинга.

Альтернативы. Сравните три подхода: (A) простое правило без ML, (B) модульное ML-решение с отдельной проверкой входных данных и (C) интегрированный сервис со сложной моделью. Для каждого укажите потребность в данных, ожидаемые инженерные риски, возможность тестирования и минимальную инфраструктуру. Таблица ниже задаёт форму отчёта, а не готовый рейтинг.

Вариант	Данные и зависимости	Проверка	Основной риск
Правила	Фиксированные условия	Набор тест-кейсов	Ограниченная гибкость
ML-модуль	Версионированные признаки	Offline и API tests	Drift и skew
Сложный сервис	Несколько компонентов	End-to-end и нагрузка	Каскадный отказ

Проверка решения. Подготовьте две позитивные и две негативные пробы; задайте метрику успеха, базовый сценарий и критерий остановки внедрения. Обсудите, какие допущения останутся непроверенными даже после удачного теста.

5.2 Ситуация 2: Источники смещения

Заказчик использует систему ранжирования поступающих заявок. Новое требование заставляет рассмотреть принцип «Источники смещения». Основная неопределённость состоит в том, как одновременно обеспечить TPR gap и процедуру обжалования, не допустить необъяснимые решения и сохранить возможность изменения модели.

Постановка. Selection, measurement, label и deployment bias возникают на разных стадиях. Коррекция модели не заменяет исправление несправедливого процесса сбора.

Действие инженера. Исследуйте, почему заявки без доступа к интернету отсутствуют в обучающих данных.

Альтернативы. Сравните три подхода: (A) простое правило без ML, (B) модульное ML-решение с отдельной проверкой входных данных и (C) интегрированный сервис со сложной моделью. Для каждого укажите потребность в данных, ожидаемые инженерные риски, возможность тестирования и минимальную инфраструктуру. Таблица ниже задаёт форму отчёта, а не готовый рейтинг.

Вариант	Данные и зависимости	Проверка	Основной риск
Правила	Фиксированные условия	Набор тест-кейсов	Ограниченная гибкость
ML-модуль	Версионированные признаки	Offline и API tests	Drift и skew
Сложный сервис	Несколько компонентов	End-to-end и нагрузка	Каскадный отказ

Проверка решения. Подготовьте две позитивные и две негативные пробы; задайте метрику успеха, базовый сценарий и критерий остановки внедрения. Обсудите, какие допущения останутся непроверенными даже после удачного теста.

5.3 Ситуация 3: Risk assessment

Заказчик использует систему ранжирования поступающих заявок. Новое требование заставляет рассмотреть принцип «Risk assessment». Основная неопределённость состоит в том, как одновременно обеспечить TPR gap и процедуру обжалования, не допустить необъяснимые решения и сохранить возможность изменения модели.

Постановка. Оценка риска связывает вероятность, тяжесть ущерба и контролирующие меры. Нулевой риск недостижим, остаточный риск документируют и принимают уполномоченные лица.

Действие инженера. Создайте реестр из пяти рисков и назначьте owners и способы мониторинга.

Альтернативы. Сравните три подхода: (A) простое правило без ML, (B) модульное ML-решение с отдельной проверкой входных данных и (C) интегрированный сервис со сложной моделью. Для каждого укажите потребность в данных, ожидаемые инженерные риски, возможность тестирования и минимальную инфраструктуру. Таблица ниже задаёт форму отчёта, а не готовый рейтинг.

Вариант	Данные и зависимости	Проверка	Основной риск
Правила	Фиксированные условия	Набор тест-кейсов	Ограниченная гибкость
ML-модуль	Версионированные признаки	Offline и API tests	Drift и skew
Сложный сервис	Несколько компонентов	End-to-end и нагрузка	Каскадный отказ

Проверка решения. Подготовьте две позитивные и две негативные пробы; задайте метрику успеха, базовый сценарий и критерий остановки внедрения. Обсудите, какие допущения останутся непроверенными даже после удачного теста.

6. Самостоятельная работа: индивидуальные проектные задания

Для каждого задания студент создаёт отдельный небольшой артефакт: схему, набор тестов, код либо таблицу сравнения, прикладывает наблюдаемый результат и объясняет ограничения.

1. **Explainability и interpretability.** Сравните коэффициенты логистической регрессии и локальное объяснение для градиентного бустинга. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

2. **Global и local explanations.** Покажите, как важность возраста глобально отличается от его вклада для одного клиента. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

3. **Пермутационная важность.** Сравните снижение F1 после перестановки двух сильно коррелирующих признаков. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

4. **SHAP.** Объясните, почему нельзя интерпретировать знак SHAP как причинное влияние фактора. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

5. **Bias и fairness.** Рассчитайте difference in TPR между двумя подгруппами и обсудите статистическую неопределенность. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

6. **Источники смещения.** Исследуйте, почему заявки без доступа к интернету отсутствуют в обучающих данных. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

7. **Privacy by design.** Проверьте возможность повторной идентификации клиента по сочетанию косвенных признаков. Сформулируйте проверя-

емое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

8. **Модельные документы.** Составьте карточку для скоринговой модели с таблицей метрик по группам. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

9. **Устойчивость.** Проверьте влияние опечаток и смены регистра на классификатор обращений. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

10. **Human oversight.** Опишите процесс отмены ошибочной блокировки аккаунта. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

11. **Risk assessment.** Создайте реестр из пяти рисков и назначьте owners и способы мониторинга. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

12. **Governance and audit.** Свяжите отдельное решение API с конкретной model card, version, input schema и датой аудита. Сформулируйте проверяемое требование, продемонстрируйте альтернативу и приложите протокол хотя бы одного эксперимента.

7. Вопросы для текущего и итогового контроля

1. В чём назначение и ограничения принципа «Explainability и interpretability»?

2. Как проверить на практике утверждение: «Сравните коэффициенты логистической регрессии и локальное объяснение для градиентного бустинга.»?

3. В чём назначение и ограничения принципа «Global и local explanations»?

4. Как проверить на практике утверждение: «Покажите, как важность возраста глобально отличается от его вклада для одного клиента.»?
5. В чём назначение и ограничения принципа «Пермутационная важность»?
6. Как проверить на практике утверждение: «Сравните снижение F1 после перестановки двух сильно коррелирующих признаков.»?
7. В чём назначение и ограничения принципа «SHAP»?
8. Как проверить на практике утверждение: «Объясните, почему нельзя интерпретировать знак SHAP как причинное влияние фактора.»?
9. В чём назначение и ограничения принципа «Bias и fairness»?
10. Как проверить на практике утверждение: «Рассчитайте difference in TPR между двумя подгруппами и обсудите статистическую неопределенность.»?
11. В чём назначение и ограничения принципа «Источники смещения»?
12. Как проверить на практике утверждение: «Исследуйте, почему заявки без доступа к интернету отсутствуют в обучающих данных.»?
13. В чём назначение и ограничения принципа «Privacy by design»?
14. Как проверить на практике утверждение: «Проверьте возможность повторной идентификации клиента по сочетанию косвенных признаков.»?
15. В чём назначение и ограничения принципа «Модельные документы»?
16. Как проверить на практике утверждение: «Составьте карточку для скоринговой модели с таблицей метрик по группам.»?
17. В чём назначение и ограничения принципа «Устойчивость»?
18. Как проверить на практике утверждение: «Проверьте влияние опечаток и смены регистра на классификатор обращений.»?
19. В чём назначение и ограничения принципа «Human oversight»?
20. Как проверить на практике утверждение: «Опишите процесс отмены ошибочной блокировки аккаунта.»?

21. В чём назначение и ограничения принципа «Risk assessment»?
22. Как проверить на практике утверждение: «Создайте реестр из пяти рисков и назначьте owners и способы мониторинга.»?
23. В чём назначение и ограничения принципа «Governance and audit»?
24. Как проверить на практике утверждение: «Свяжите отдельное решение API с конкретной model card, version, input schema и датой аудита.»?
25. Почему высокое offline-качество не гарантирует надежность промышленной AI-системы?
26. Какие артефакты нужны для воспроизводимости и анализа инцидента?

8. Глоссарий ключевых понятий

Explainability и interpretability
Некоторые модели внутренне прозрачны, другие требуют пост-хок объяснения.

Global и local explanations
Global объяснение описывает поведение модели в целом; local относится к конкретному примеру.

Пермутационная важность
Перемешивание значения признака оценивает изменение качества; коррелированные признаки затрудняют интерпретацию.

SHAP
SHAP распределяет вклад относительно выбранного базового значения и связан с идеями Шепли.

Bias и fairness
Различные групповые метрики могут конфликтовать между собой; выбор понятия справедливости является нормативным решением.

Источники смещения
Selection, measurement, label и deployment bias возникают на разных стадиях.

Privacy by design
Минимизация данных, контроль доступа, retention и ограничение целей применяются до обучения.

Модельные документы
Model cards и datasheets описывают назначение, данные, оценки и ограничения, обеспечивая возможность аудита.

Устойчивость
Модель может давать разные ответы при небольших изменениях входа, смене домена или наличии шума.

Human oversight
Автоматизация должна иметь понятные правила проверки, обжалования и эскалации.

Risk assessment

Оценка риска связывает вероятность, тяжесть ущерба и контролирующие меры.

Governance and audit

Управление включает политику данных, модельных изменений, журнал решений и независимую проверку.

9. Итоги и связь со следующей темой

Представленные методы имеют смысл как части целостного управляемого жизненного цикла. Ключевой результат инженерного проектирования состоит не в максимальном числе компонентов или сложности модели, а в убедительной связи требований, проектных решений, измерений и механизмов сопровождения. Дальнейшее изучение курса развивает эти связи на следующем этапе жизненного цикла.

Список литературы

[1] NIST. AI Risk Management Framework (AI RMF 1.0). NIST AI 100-1, 2023. https://doi.org/10.6028/NIST.AI.100-1.

[2] Lundberg S. M., Lee S.-I. A Unified Approach to Interpreting Model Predictions. NeurIPS, 2017. https://arxiv.org/abs/1705.07874.

[3] Mitchell M. et al. Model Cards for Model Reporting. FAT*, 2019. https://arxiv.org/abs/1810.03993.

[4] Murphy K. P. Probabilistic Machine Learning: An Introduction. MIT Press, 2022. https://probml.github.io/pml-book/.

[5] Huyen C. Designing Machine Learning Systems. O'Reilly Media, 2022. https://www.oreilly.com/library/view/designing-machine-learning/9781098107956/.

[6] Sculley D. et al. Hidden Technical Debt in Machine Learning Systems. NeurIPS, 2015. https://research.google/pubs/hidden-technical-debt-in-machine-learning-systems/.