

КАЗАХСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ АЛЬ-ФАРАБИ

А.Т. Агишев

МЕТОДЫ И ТЕХНОЛОГИИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

**Сборник лекций для студентов магистратуры,
обучающихся по образовательной программе
«7M06201 - Радиотехника, электроника и телекоммуникации»**

Алматы, 2025

Лекция 7. Алгоритмы машинного обучения

Цель лекции

Рассмотреть основные алгоритмы машинного обучения, их принципы работы, математические основы и области применения. Научить студентов различать алгоритмы по типу задач (классификация, регрессия, кластеризация) и оценивать их достоинства и ограничения при решении инженерных и телекоммуникационных задач.

Основные вопросы:

1. Классификация алгоритмов машинного обучения.
2. Алгоритмы обучения с учителем: линейные и нелинейные модели.
3. Алгоритмы обучения без учителя.
4. Методы ансамблевого обучения.
5. Нейронные сети как универсальные аппроксиматоры.
6. Критерии выбора алгоритма для конкретной задачи.
7. Применение алгоритмов в радиотехнике и телекоммуникациях.

Краткие тезисы:

1. **Классификация алгоритмов машинного обучения.** По Alpaydin (2020), все алгоритмы делятся по типу обучения и назначению:
 - **Supervised learning:** классификация и регрессия (на размеченных данных);
 - **Unsupervised learning:** кластеризация, уменьшение размерности, поиск аномалий;
 - **Reinforcement learning:** обучение агента действовать оптимально во времени.Каждый алгоритм реализует аппроксимацию зависимости $f: X \rightarrow Y$ различными способами - линейными, вероятностными или нейросетевыми.

2. **Линейные модели.**

- **Линейная регрессия:**

$$y = w_0 + \sum_{i=1}^n w_i x_i + \epsilon$$

Используется для прогнозирования непрерывных параметров.

- **Логистическая регрессия:**

$$P(y = 1 | x) = \frac{1}{1 + e^{-(w^T x + b)}}$$

Применяется в бинарной классификации сигналов и событий. Достоинства: простота, интерпретируемость. Недостатки: неспособность моделировать нелинейные зависимости.

3. **Метод k-ближайших соседей (k-NN).** Принцип: объект классифицируется по классам ближайших соседей в пространстве признаков.

$$y = \text{mode}\{y_i \mid x_i \in N_k(x)\}$$

Применяется при распознавании сигналов и изображений. Достоинство - простота и отсутствие этапа обучения. Недостатки - высокая вычислительная сложность при больших данных.

4. **Метод опорных векторов (SVM).** Основан на поиске гиперплоскости, максимально разделяющей классы:

$$\max_{\mathbf{w}, b} \frac{2}{\|\mathbf{w}\|} \text{ при } y_i(\mathbf{w}^T x_i + b) \geq 1$$

Используются **ядровые функции (kernel functions)** для нелинейных задач (полиномиальные, RBF). Преимущества: устойчивость и высокая точность на малых выборках. Применения: классификация модуляций, анализ спектров.

5. **Деревья решений (Decision Trees).** Построены на последовательном разбиении пространства признаков по критериям (информационный прирост, энтропия, индекс Джини). Пример критерия Джини:

$$G = 1 - \sum_k p_k^2$$

Достоинства: интерпретируемость и работа с категориальными данными. Недостатки: склонность к переобучению.

Ансамблевые методы:

- **Random Forest** - усреднение многих деревьев для снижения разброса.
- **Gradient Boosting (XGBoost, LightGBM)** - последовательное обучение деревьев с учетом ошибки предыдущих.

6. **Байесовские методы.** Основываются на формуле Байеса:

$$P(C \mid X) = \frac{P(X \mid C)P(C)}{P(X)}$$

Пример - **Наивный Байесовский классификатор (Naïve Bayes)**, предполагающий независимость признаков. Эффективен при работе с

текстами, сигналами и при малом объеме данных. Недостатки - сильное предположение о независимости признаков.

7. Алгоритмы без учителя.

- **k-Means:** минимизация внутрикластерного разброса:

$$J = \sum_{i=1}^k \sum_{x \in C_i} \|x - \mu_i\|^2$$

- **PCA (Principal Component Analysis):** уменьшение размерности данных через ортогональные преобразования.
- **Autoencoders:** нейронные сети, восстанавливающие входной сигнал через сжатое представление.

Применения: анализ спектров, кластеризация сигналов, выявление аномалий в трафике.

8. Нейронные сети. По Goodfellow et al. (2016), искусственная нейронная сеть - это нелинейная композиция функций, приближающая зависимость между входами и выходами. Базовое уравнение:

$$y = f(Wx + b)$$

где f - функция активации (ReLU, Sigmoid, Tanh). Современные архитектуры:

- MLP (многослойный перцептрон);
- CNN (сверточные сети) - для изображений и сигналов;
- RNN, LSTM - для временных рядов и речи. Применения: предсказание параметров каналов связи, фильтрация помех, автоматическая классификация спектров.

9. Критерии выбора алгоритма. При выборе алгоритма учитываются:

- размер и структура данных;
- тип задачи (классификация, регрессия, кластеризация);
- требуемая интерпретируемость;
- вычислительная сложность;
- устойчивость к шуму.

Пример: для анализа сигналов - CNN или SVM; для кластеризации сетевых данных - k-means; для прогнозирования нагрузки сети - Random Forest или LSTM.

10. Применение в радиотехнике и телекоммуникациях.

- Распознавание и классификация модуляций - SVM, CNN.
- Прогнозирование трафика и отказов - Random Forest, Gradient Boosting.
- Выявление аномалий в каналах - k-means, Autoencoder.
- Оптимизация частотного распределения - Reinforcement Learning.
- Анализ электромагнитных помех - PCA, нейронные сети.

Вопросы для контроля, изучаемого материал:

- 1) На какие основные группы делятся алгоритмы машинного обучения?
- 2) В чём различие между линейной и логистической регрессией?
- 3) Как работает метод опорных векторов и зачем нужны ядровые функции?
- 4) Каков принцип построения дерева решений и ансамблевых моделей?
- 5) В чём преимущества и недостатки байесовских алгоритмов?
- 6) Какие задачи решаются с помощью методов без учителя?
- 7) Какую роль играют нейронные сети в современной инженерной практике?
- 8) Как выбрать подходящий алгоритм для конкретной задачи?

Рекомендуемый список литературных источников:

1. Russell, S., Norvig, P. Artificial Intelligence: A Modern Approach. 4th Edition. Pearson, 2021.
2. Goodfellow, I., Bengio, Y., Courville, A. Deep Learning. MIT Press, 2016.
3. Alpaydin, E. Introduction to Machine Learning. 4th Edition. MIT Press, 2020.