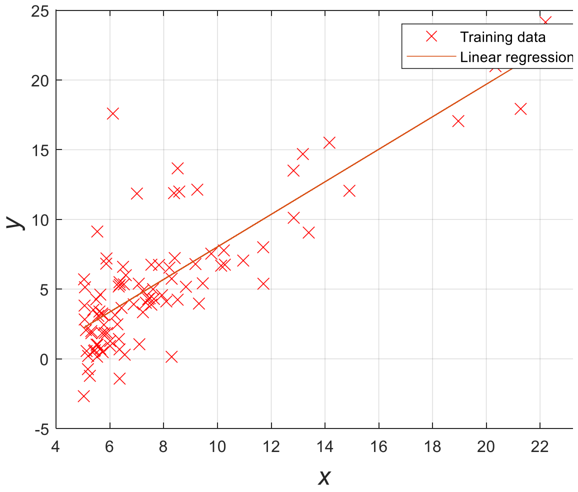
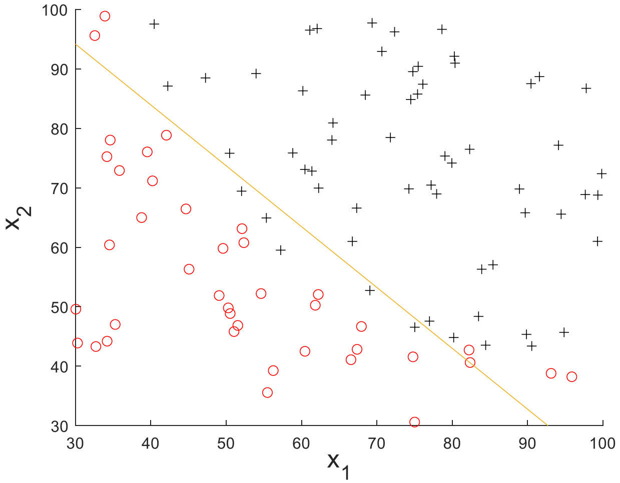
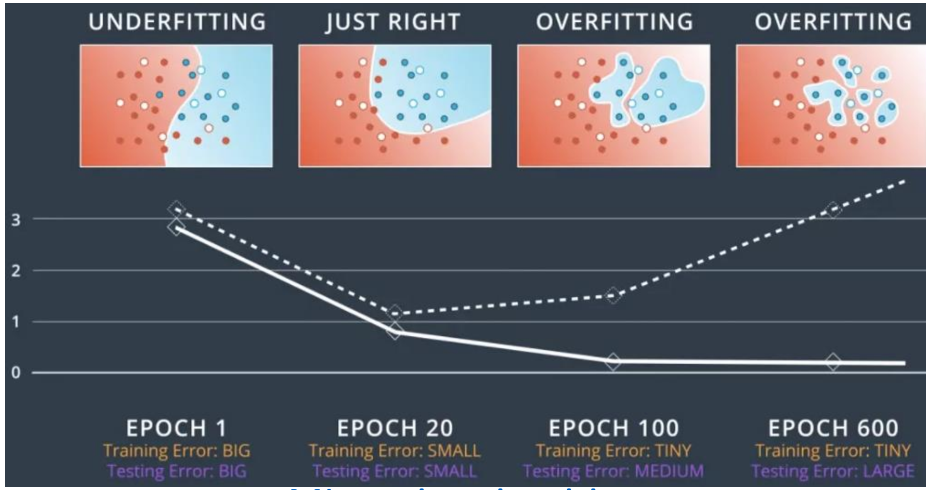
Лекция 1. Искусственный интеллект и основы машинного обучения (МО) / Artificial Intelligence (AI) and Machine Learning (ML) basics

1. **Область исследования и основные понятия / Terms**
2. **Линейная и полиномиальная регрессия / Linear and polynomial regression**
3. **Логистическая регрессия/ Logistic regression**
4. **Настройка МО/ ML settings**







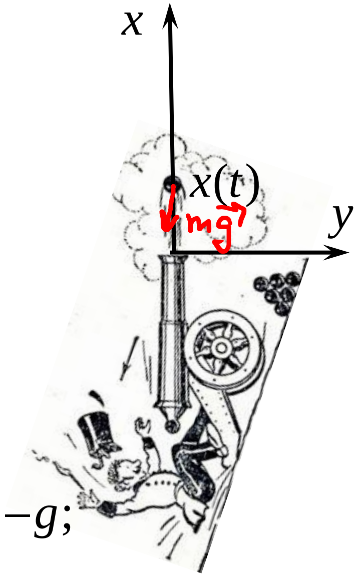
ML settings intuition

Область исследования и основные понятия / Terms

Математическое моделирование / Modeling

Детерминированное / Deterministic

Стохастическое / Stochastic



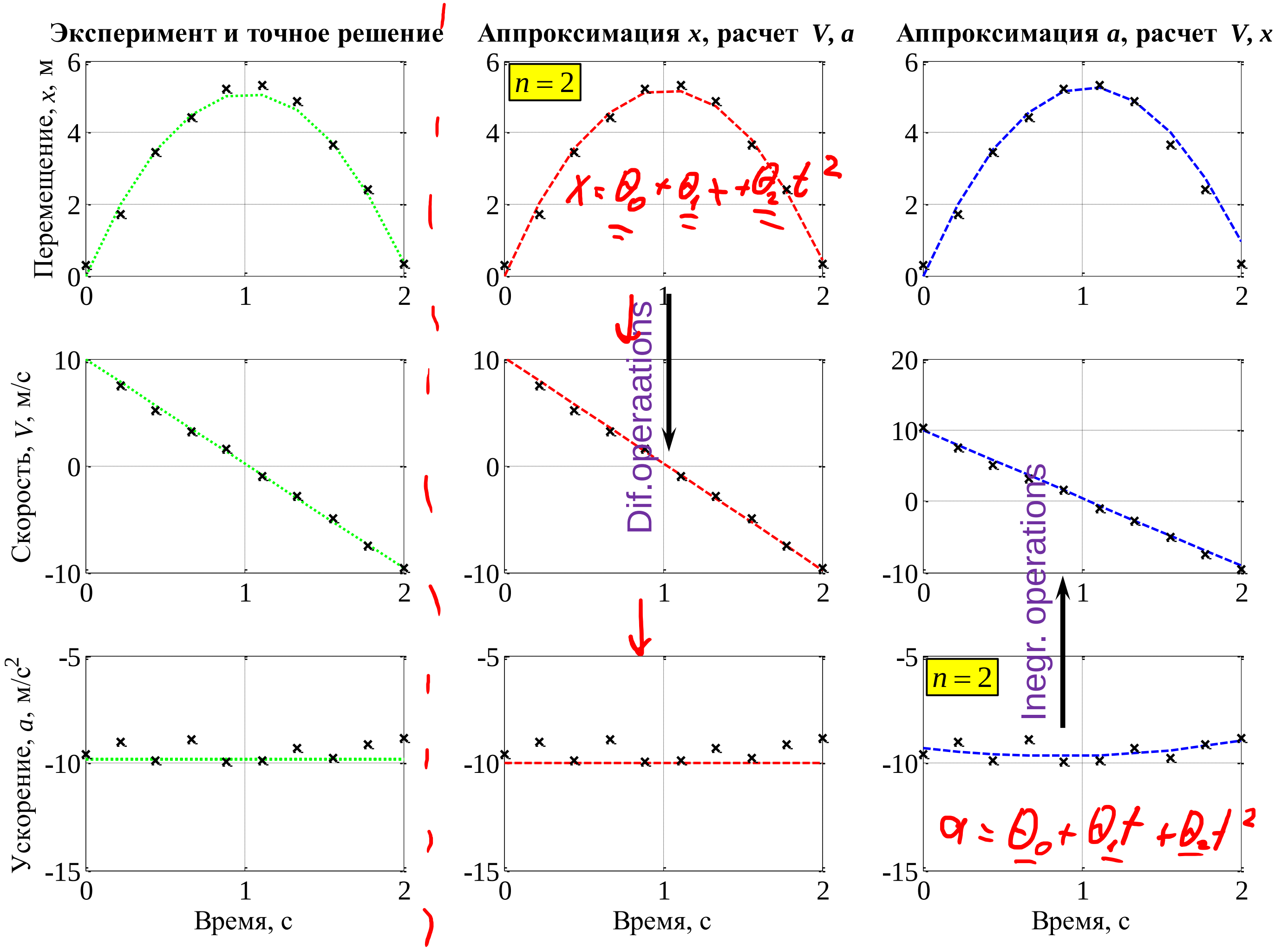
$$\frac{d^2x}{dt^2} = -g;$$

$$V(0) = V_0, \quad x(0) = x_0\,.$$

$$a = -g,$$

$$V = -gt + V_0,$$

$$x = -\frac{gt^2}{2} + V_0 t + x_0.$$

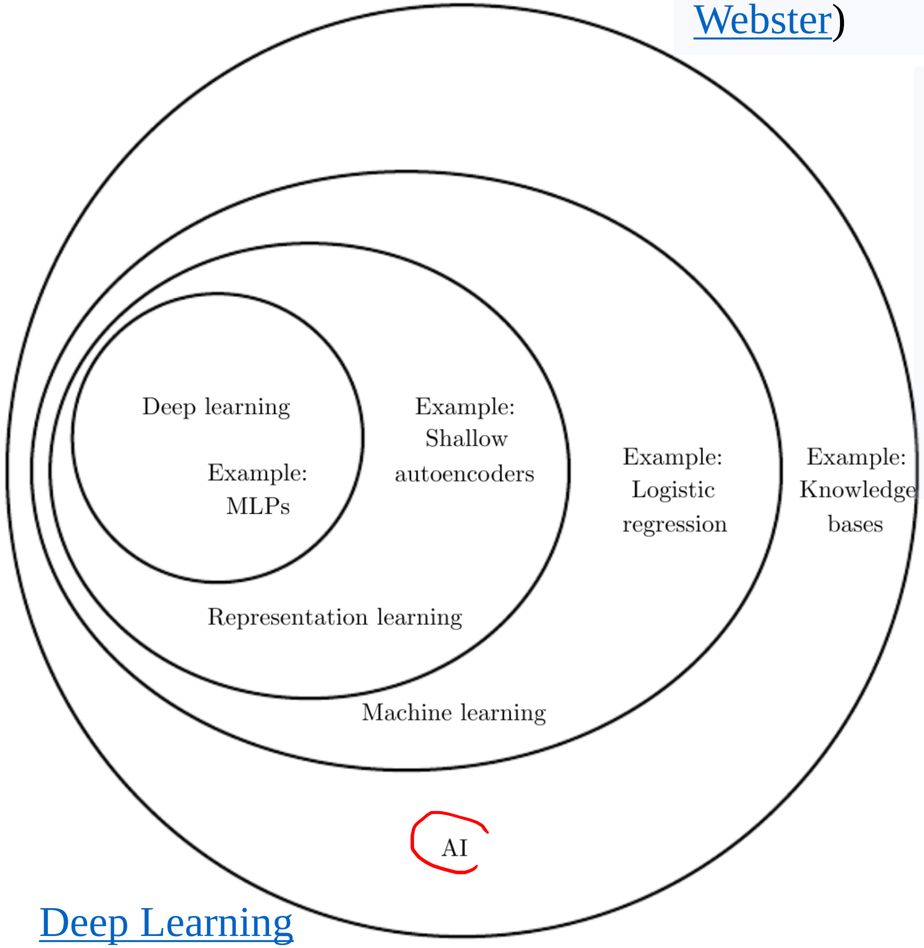


Область исследования и основные понятия / Terms

"When I use a word," Humpty Dumpty said in rather a scornful tone,
"it means just what I choose it to mean—neither more nor less."
Through the looking glass

Искусственный интеллект / Artificial Intelligence (AI):
область информатики, занимающаяся моделированием разумного поведения в компьютерах / a branch of computer science dealing with the simulation of intelligent behavior in computers (Merriam-Webster)

Машинное обучение / Machine Learning (ML):
область знаний, в которой компьютеры обучаются без явного программирования/ field of study that gives computers the ability to learn without being explicitly programmed (Arthur Samuel, 1959);
задача «З», в ходе решения которой программа обучается из опыта «О» и повышает меру качества «К» / well-posed learning problem: a computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E (Tom Mitchell, 1998)



Deep Learning

Идея / **Idea** + **Подход** / **Approach** + **Средства** / **Tools**

AI

ML

HNN
DATA

Область исследования и основные понятия / Terms

Машинное обучение / ML

Признаки / Features:
свойства объекта исследования,
выделяемые при обучении

① круг, квадрат.
Это квадрат!!!

% выделение границ;
% построен. контура;
% нахождение центра;
% нахожд. осей симм.;
% логический вывод.

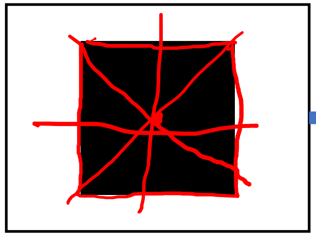
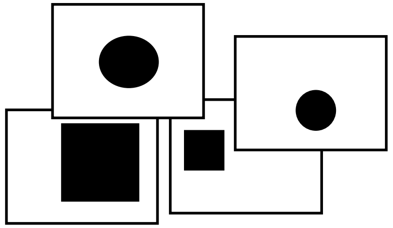


Fig.1- A figure

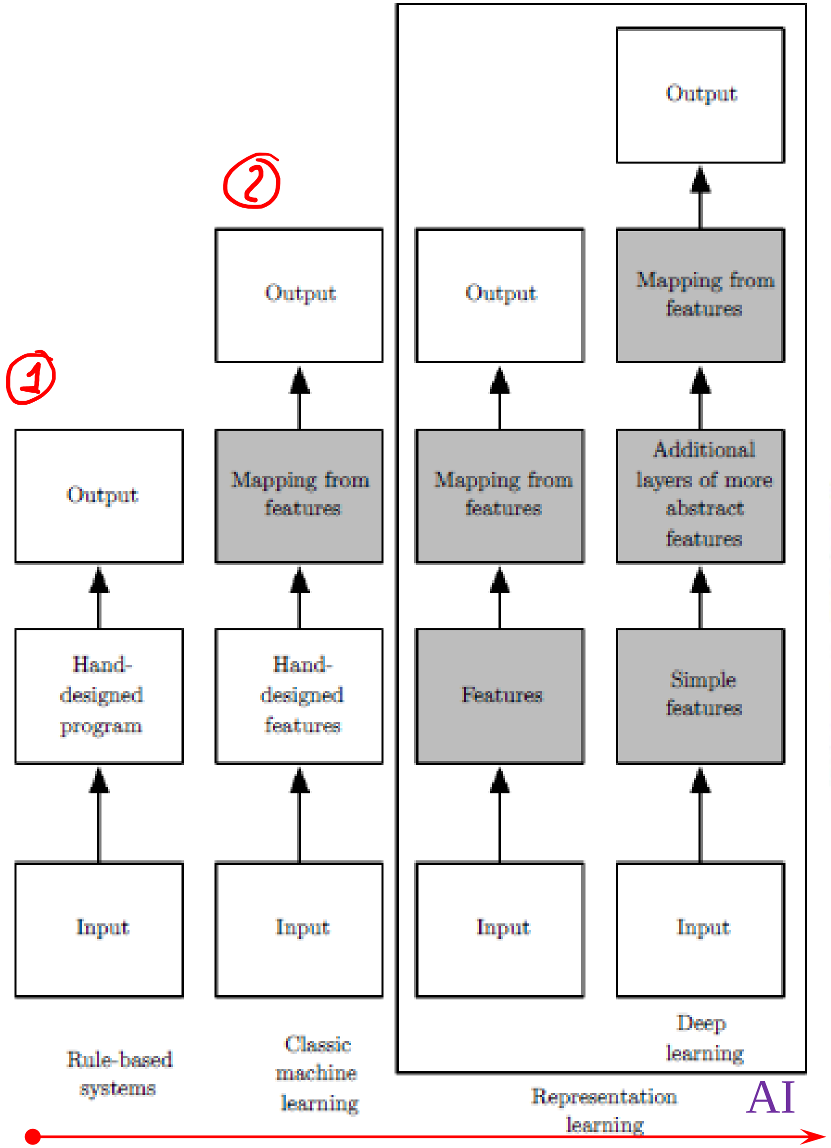
Это квадрат
(не круг)!!!

% обученная
программа

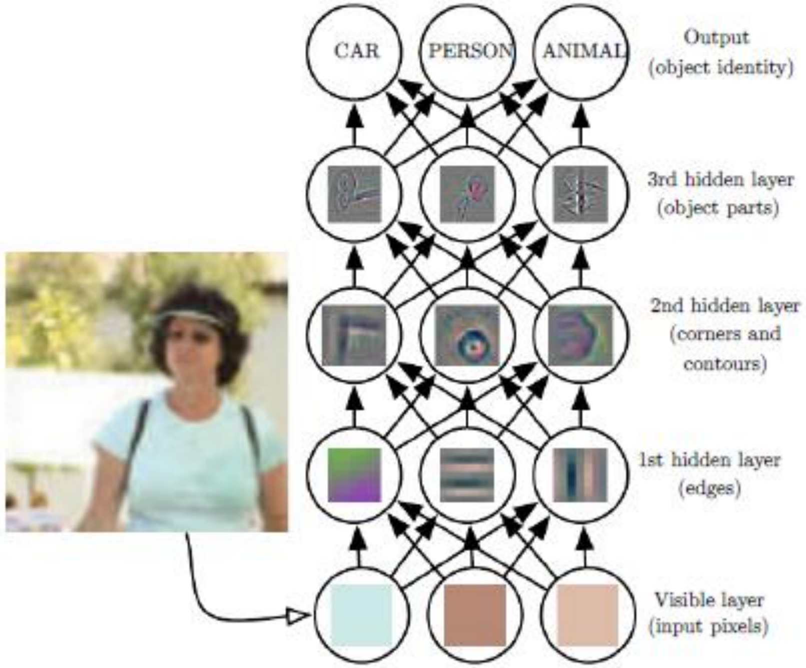
% обучение из
опыта «О» и
повышение меры
качества «К»



Data set



AI



Deep Learning

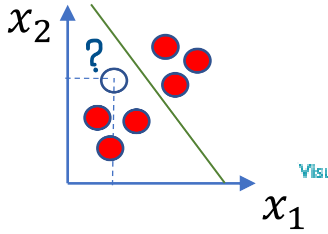
Область исследования и основные понятия / Terms

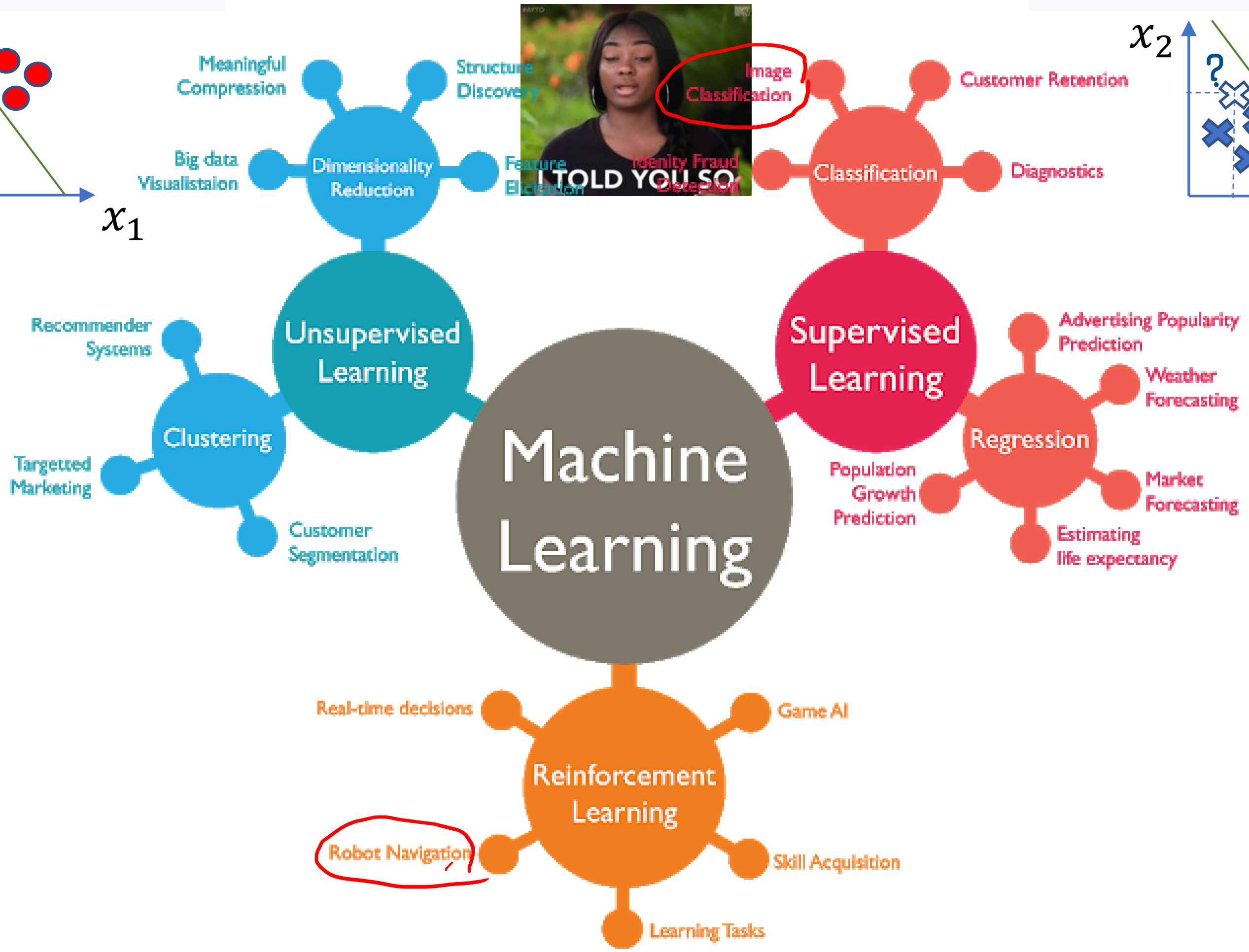
Машинное обучение / ML

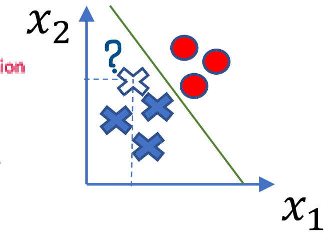
Обучение без учителя / Unsupervised learning

…Обучение с подкреплением / Reinforcement learning…

Обучение с учителем / Supervised learning







[Types of Machine Learning and Examples of Applications]

Зачем это нужно? Что нужно знать, чтобы начать? Где об этом почитать? / FAQs

Машинное обучение в жизни / ML applications

Дома, в компьютере, в тел.:
поисковые системы, голосовые команды, переводчики, спам-фильтры, игры, обработки фото, новости, реклама, поиск друзей, *создание семей …*

На улице:
прогноз погоды, навигация, распознавание лиц, номеров автомобилей, *беспилотный транспорт…*

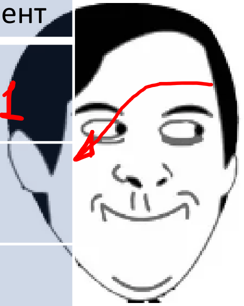
На работе, в обществ. местах:
автоматизация и управление, *роботизация*, ассортимент товаров, *диагностика, лечение…*

Зачем это нужно? Что нужно знать, чтобы начать? Где об этом почитать? / FAQs

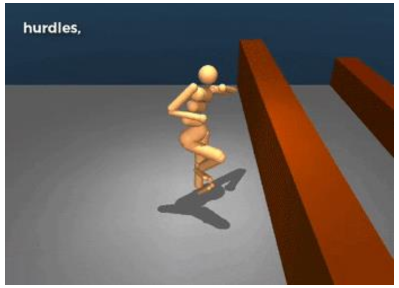
Основы машинного обучения / ML Basics

Линейная алгебра / Linear algebra + **Мат. анализ / Calculus** + **Теор. вер. / Probability theory**

Название	Геом. аналог	Обозначения (в N-мерном простр.)	Кол-во компонент
Скаляр	·	a	$N^0 = 1$
Вектор		$\vec{a}$ $\mathbf{a}$ A $[a_i]$	N^1
Тензор		$\overset{2}{T}_a = [a_{ij}]$	N^2
…		$\overset{n}{T} = [a_{\underbrace{i \dots j}_{n}}]$	N^n
М-ца-строка (столбец)		$A = ((a_i))$	N^1
Матрица		$A = ((a_{ij}))$	N^2
…		$A = ((a_{i \dots j}))$	N^n



$$\nabla a = \left[\left[\frac{\cancel{\partial} a}{\cancel{\partial} x_i}\right]\right].$$



Google Parkour

The **expectation** of some function $f(x)$ with respect to a probability distribution $p(x)$:

$$E(f(x)) = \sum_x p(x) f(x).$$

The conditional maximum likelihood estimator:

$$\Theta_{ML} = \operatorname{argmax} \sum_{i=1}^{m} \log\left(p\left(y^{(i)} | x^{(i)}; \Theta\right)\right).$$

Bellman Expectation Equation for State-Action Value Function (Q-Function):

$$q_\pi(s, a) = E_\pi[R_{t+1} + \gamma q_\pi(s_{t+1}, a_{t+1}) | s_t = s, a_t = a].$$

Полезные ссылки / Links

Онлайн курсы, обучающие ресурсы:

Neural Networks: серия видео с отличной визуализацией с канала «3Blue1Brown» на английском языке, плюс видео с русским дублирование на смежном канале;

Andrew Ng – Machine Learning: 11-недельный бесплатный интерактивный курс на английском языке по основам машинного обучения на платформе «Coursera»;

sim0nsays – Deep Learning на пальцах: бесплатный видео курс лекций на русском языке, плюс на канале есть ссылка на ресурс с полным курсом и заданиями;,

Kaggle: курсы, базы данных для машинного обучения (дата сеты), соревнования на английском языке.

Книги, статьи:

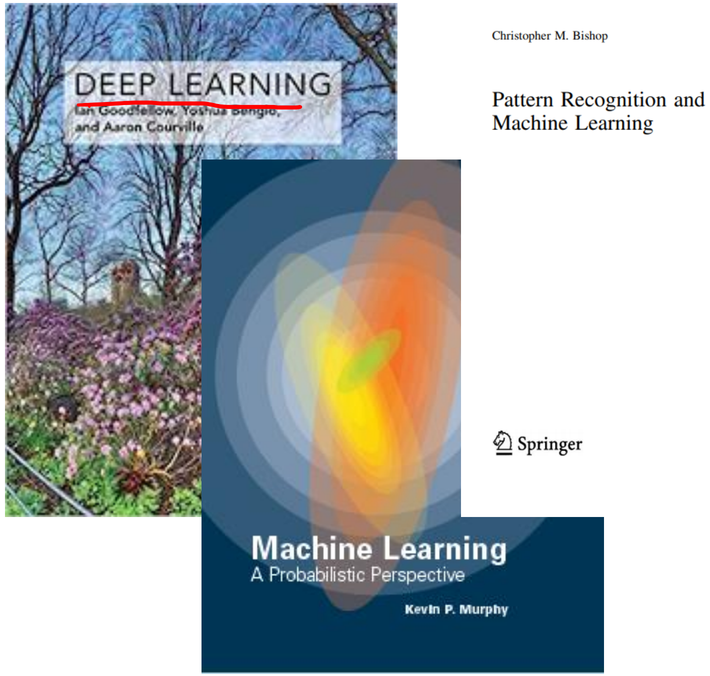
Deep Learning by Ian Goodfellow and Yoshua Bengio and Aaron Courville, 2016.

Pattern Recognition and Machine Learning by C.M. Bishop, 2006.

Machine Learning: A Probabilistic Perspective by K.P. Murphy, 2012.

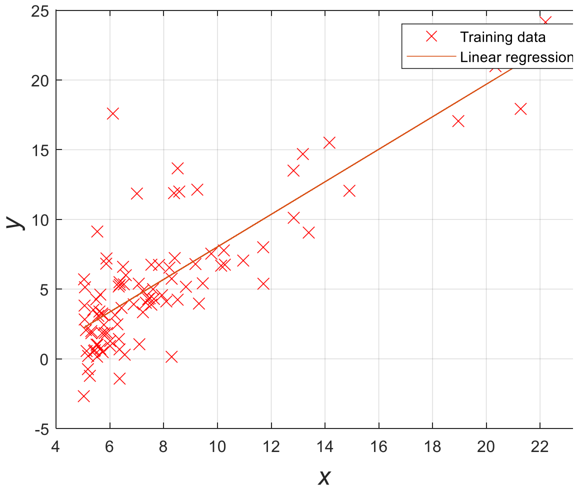
Elsevier , Springer : поисковые системы статей крупнейших издательств

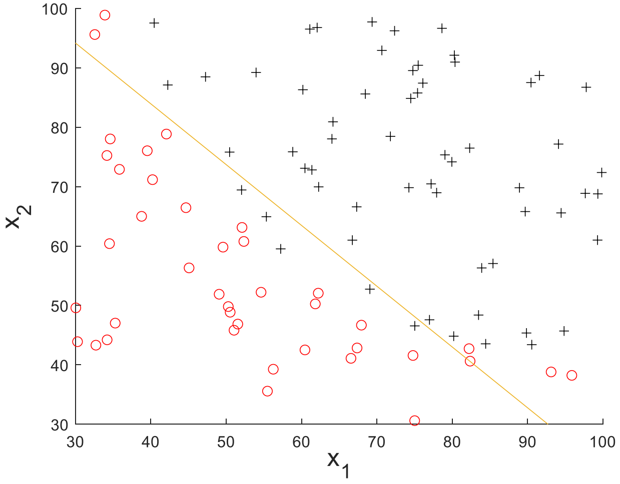
SJR , WoS : поисковые системы журналов, рейтинг журналов

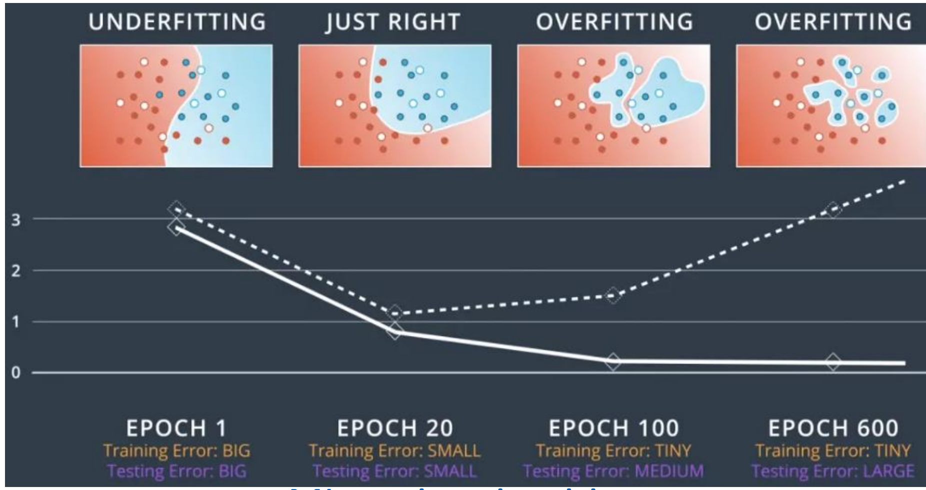


Лекция 1. Искусственный интеллект и основы машинного обучения (МО) / Artificial Intelligence (AI) and Machine Learning (ML) basics

1. Область исследования и основные понятия / Terms
2. Линейная и полиномиальная регрессия / Linear and polynomial regression
3. Логистическая регрессия/ Logistic regression
4. Настройка МО/ ML settings





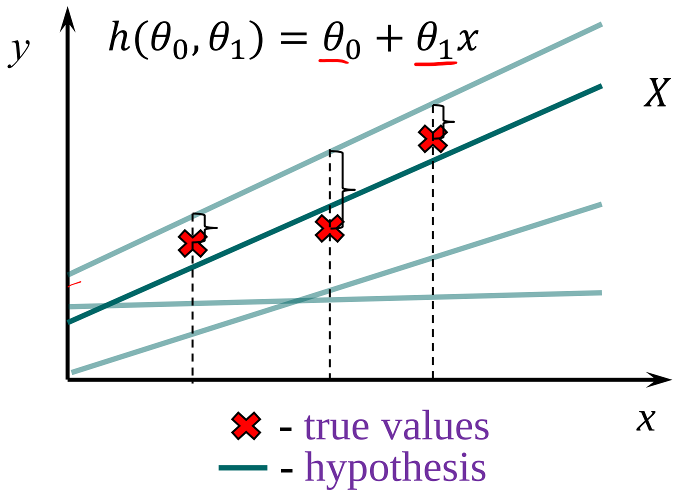


ML settings intuition

Все начиналось с линейной регрессии / Linear regression

Регрессионный (от лат. regressio – обратное движение, отход) **анализ:**
набор статистических действий для оценки связи зависимой переменной и одной или нескольких независимых переменных (Wikipedia)

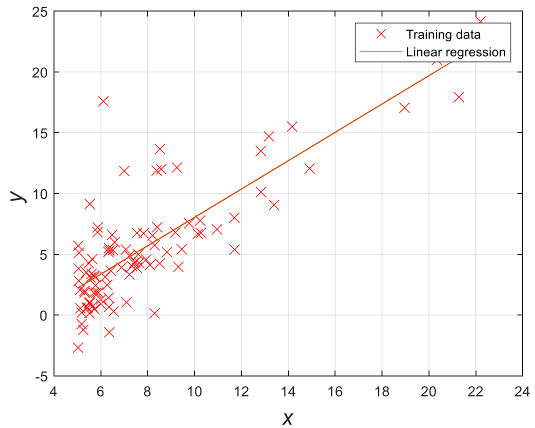
Основная идея линейной регрессии: поиск наилучшей функциональной зависимости множества $Y = \big((y_i)\big)$ от множества $X = \big((x_i)\big)$ (i=1…m) по критерию минимума некоторой функции качества:

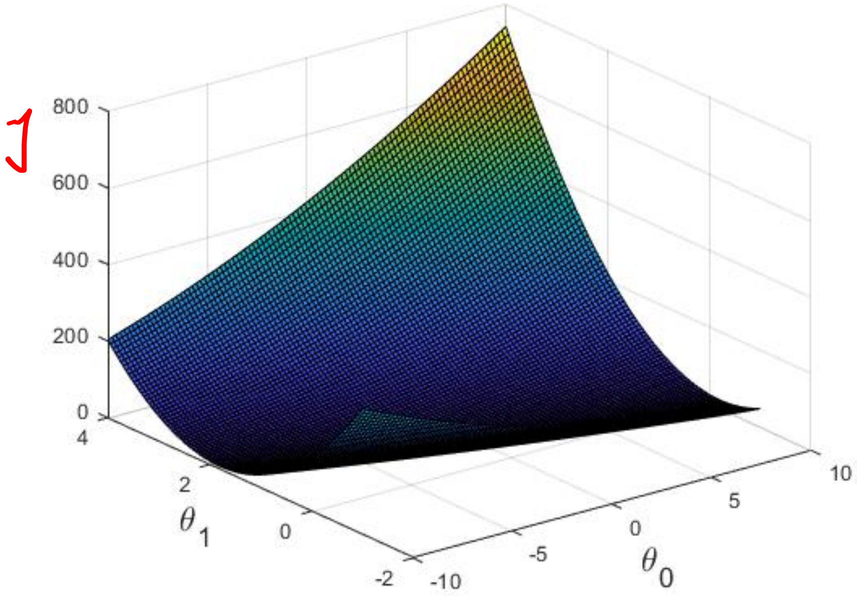


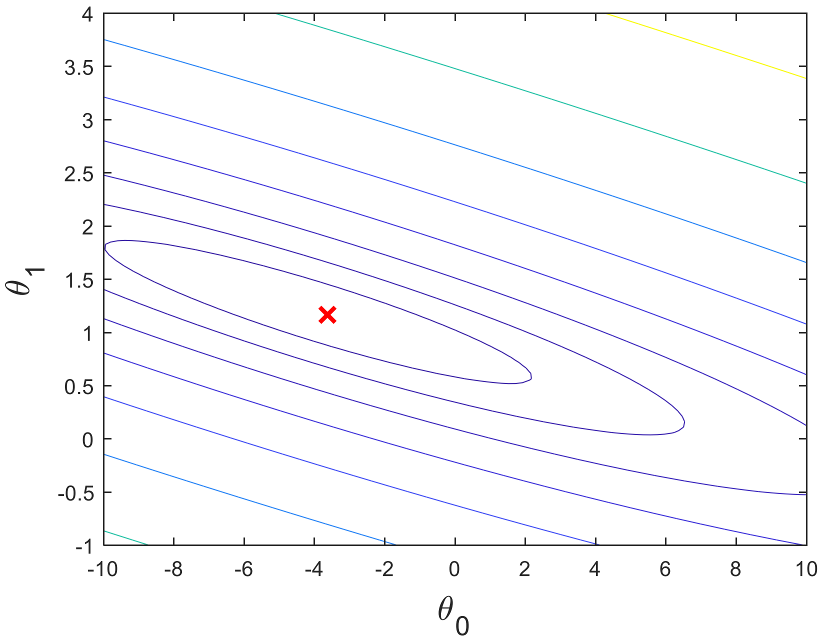
$$X = \left(\begin{pmatrix} x^{(1)} \\ \dots \\ x^{(m)} \end{pmatrix}\right);\ Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right);\ \Theta = \left(\begin{pmatrix} \theta_0 \\ \theta_1 \end{pmatrix}\right);$$

$$\boxed{J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h^{(i)} - y^{(i)}\right)^2 \Rightarrow \min.}$$

Cost function



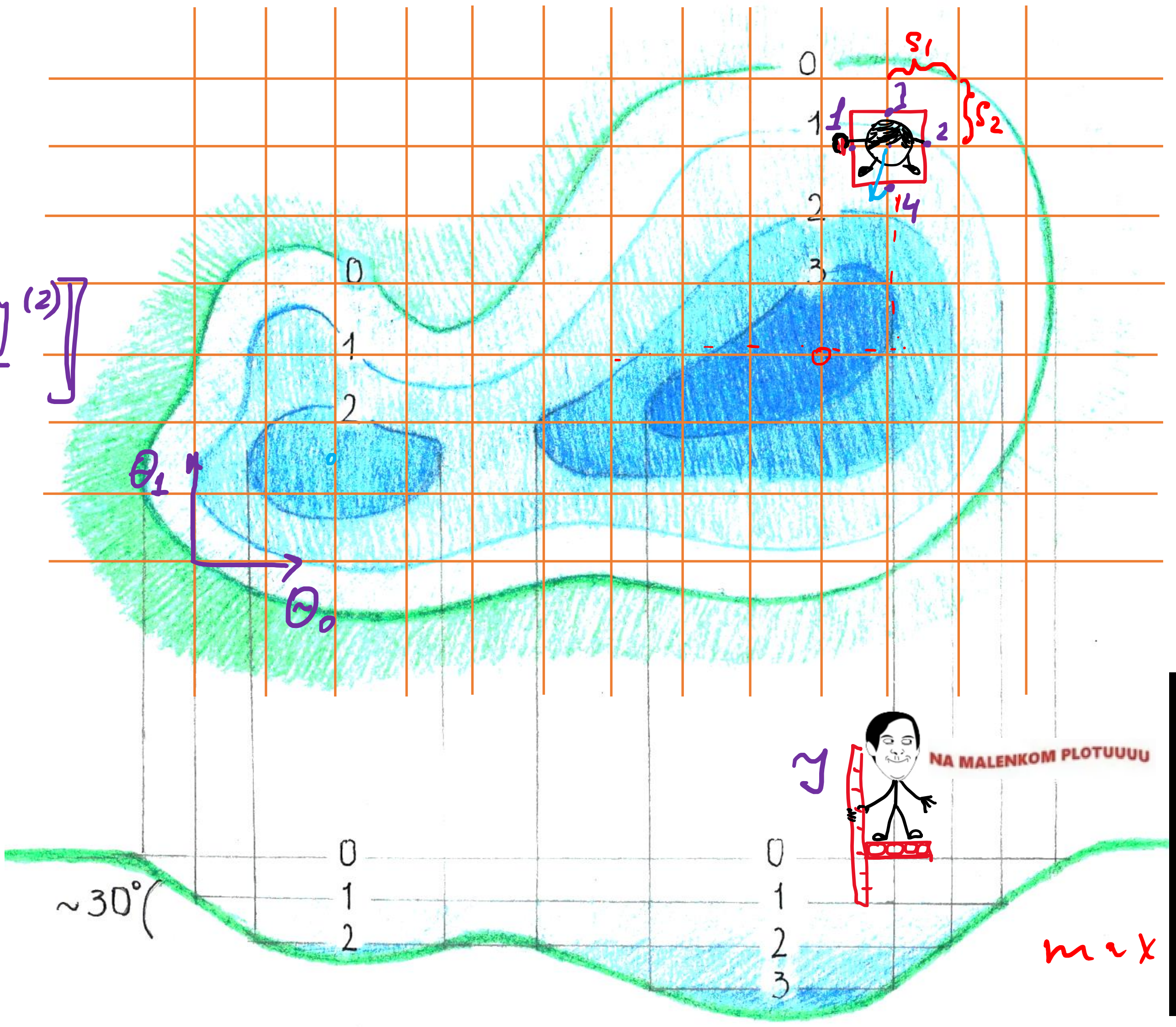




Все началось с линейной регрессии / Linear regression

Лирическое отступление. Метод градиентного спуска (подъема)

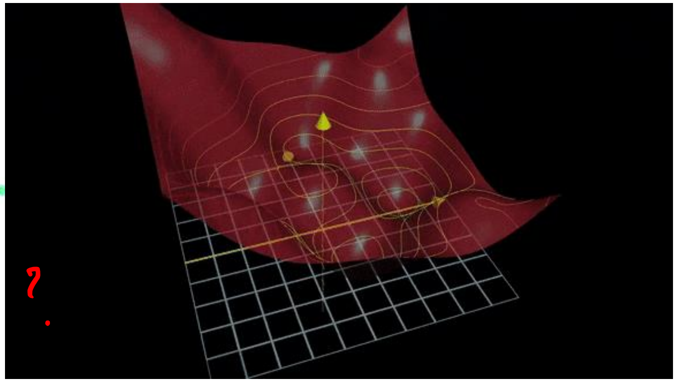
$$\vec{C} = \left[\frac{J^{(2)} - J^{(1)}}{S_1} \quad \frac{J^{(4)} - J^{(2)}}{S_2} \right]$$



$$S_1, S_2 \to 0$$

$$\vec{g} = \left[\frac{\partial J}{\partial \theta_0} \quad \frac{\partial J}{\partial \theta_1} \right]$$

max ?



Want to know more?
[DeepMind x UCL | Deep Learning Lectures | 5/12 | Optimization for Machine Learning]

[Пруд]

[Gradient descent intuition]

Все начиналось с линейной регрессии / Linear regression

Лирическое отступление 2. Действия над матрицами

$$A = ((a_{ij})) = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & a_{11} & a_{12} & a_{13} \\ 2 & a_{21} & a_{22} & a_{23} \\ 3 & a_{31} & a_{32} & a_{33} \end{array}$$

$\overbrace{i, j}^{2} = \underbrace{1 \dots 3}$

М-ца 2го ранга в 3мерном простр.

№ строки   № столбца

a_{ij}

1, 2 3

1	1
1	2
1	3
2	1
2	2
2	3
3	1
3	2
3	3

a_{ij}

$a_{ii} \leftrightarrow a_{11} + a_{22} + a_{33}$

1. Умножение матрицы на число:
$C = \alpha A, \quad c_{ij} = \alpha a_{ij}.$

2. Сложение матриц:
$C = A + B, \quad c_{ij} = a_{ij} + b_{ij}.$

3. Транспонирование матрицы
$A^T, \quad a^T_{ij} = a_{ji}.$

4. Обратная матрица
$A^{-1}, \quad AA^{-1} = I.$

$$\left(\left(\begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}\right)\right)$$

5. Умножение матриц:
$C = AB, \quad c_{ij} = a_{ik} \cdot b_{kj}.$

$$c_{11} = a_{11} b_{11} + a_{12} b_{21} + a_{13} b_{31}$$

$$\left(\left(\begin{matrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{21} \\ a_{31} & a_{32} & a_{33} \end{matrix}\right)\right) \left(\left(\begin{matrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{matrix}\right)\right)$$

5*. Скалярное произведение тензоров:
$c = T_A \cdot T_B, \quad c_{ij} = a_{ij} b_{ij}.$

$c_{11} = a_{11} b_{11} \quad c_{12} = a_{12} b_{12} \; \dots$

6. Дифференцирование и интегрирование полей матричных и тензорных величин:

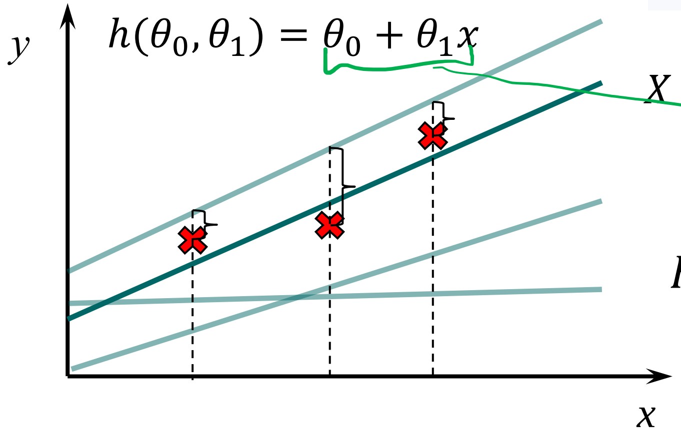
$$C = \frac{\partial A}{\partial x}, \quad c_{ij} = \frac{\partial a_{ij}}{\partial x}.$$

$$C = \int A\,dx, \quad c_{ij} = \int a_{ij}\,dx.$$

Все началось с линейной регрессии / Linear regression

Регрессионный (от лат. regressio – обратное движение, отход) **анализ:** набор статистических действий для оценки связи зависимой переменной и одной или нескольких независимых переменных (Wikipedia)

Основная идея линейной регрессии: поиск наилучшей функциональной зависимости множества $Y = \left(\left(y^{(i)}\right)\right)$ от множества $X = \left(\left(x^{(i)}\right)\right)$ (i=1…m) по критерию минимума некоторой функции качества:



$$X = \left(\begin{pmatrix} x^{(1)} \\ \dots \\ x^{(m)} \end{pmatrix}\right);\ Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right);\ \Theta = \left(\begin{pmatrix} \theta_0 \\ \theta_1 \end{pmatrix}\right);$$

$$\to X = \left(\begin{pmatrix} 1 & x^{(1)} \\ \vdots & \vdots \\ 1 & x^{(m)} \end{pmatrix}\right)\left(\begin{pmatrix} \theta_0 \\ \theta_1 \end{pmatrix}\right) = \left(\begin{pmatrix} \theta_0 + x^{(1)}\theta_1 \\ \vdots \\ \theta_0 + x^{(m)}\theta_1 \end{pmatrix}\right)$$

$$H = X\Theta, \quad \boxed{J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}\left(\underbrace{h^{(i)}}_{h(\theta_0,\theta_1)} - y^{(i)}\right)^2 \Rightarrow \min,}$$

$$\frac{\partial J}{\partial \theta_0} = \frac{1}{m}\sum_{i=1}^{m}\left(h^{(i)} - y^{(i)}\right)x^{(i,1)}$$

$$\frac{\partial J}{\partial \theta_1} = \frac{1}{m}\sum\left(h^{(i)} - y^{(i)}\right)x^{(i,2)}$$

$$\nabla J = \left[\frac{\partial J}{\partial \theta_0}\ \frac{\partial J}{\partial \theta_1}\right]$$

$$\boxed{\nabla J = \frac{1}{m}X^T(H - Y).}$$

Алгоритм поиска минимума функции качества.

1. Задать начальные значения компонент матрицы Θ случайным образом.
2. Рассчитать функцию гипотезы $H = X\Theta$ и вектор градиента функции качества: $\nabla J = \frac{1}{m}X^T(H - Y)$.
3. Зная текущие значения компонент (индекс «С») матрицы Θ найти новые значения компонент (индекс «Н»), двигаясь в направлении, противоположном направлению вектора градиента, с шагом, характеризуемым параметром скорости обучения α:

$\Theta^H = \Theta^C - \alpha\nabla J$, или в скалярной форме $\theta_0^H = \theta_0^C - \alpha\frac{\partial J}{\partial \theta_0}$, $\theta_1^H = \theta_1^C - \alpha\frac{\partial J}{\partial \theta_1}$.

4. Повторять пункты 2-3 до достижения минимума J по условию малости изменения значения функции на двух соседних итерациях или по условию достижения максимального количества итераций: $J^H - J^C < \delta$, #итер. > N_{max}.
5. Вывод результатов: Θ.

Все начиналось с линейной регрессии / Linear regression

Регрессия / Regression

Линейная / Linear

Нелинейная / Non-Linear

Одна переменная (признак) / One variable (feature)

Мн. переменных / Multiple variables (features)

$h(\theta_0, \theta_1) = \theta_0 + \theta_1 x$

$X = \left(\begin{pmatrix} x^{(1)} \\ \dots \\ x^{(m)} \end{pmatrix}\right); Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right).$

Training data (X, Y)

Learning Algorithm $\theta_0\ \theta_1$

x → h → h

Батч / Batch: на каждом шаге град. спуска используются все обучающие данные.

$h(\theta_0, \theta_j) = \theta_0 + \theta_k x_k, (j, k = 1 \dots n)$

$X = \left(\begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & & x_n^{(1)} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_1^{(m)} & x_2^{(m)} & & x_n^{(m)} \end{pmatrix}\right); Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right); \Theta = \left(\begin{pmatrix} \theta_0 \\ \dots \\ \theta_n \end{pmatrix}\right);$ ⇒

$\underset{m,1}{H} = \underset{[m,\,n+1]}{X}\underset{[n+1,\,1]}{\Theta},$

$$\boxed{J(\theta_0, \theta_j) = \frac{1}{2m}\sum_{i=1}^{m}\left(h^{(i)} - y^{(i)}\right)^2 \Rightarrow \min.}$$

Алгоритм поиска минимума функции качества.

1. Задать нач.зн. Θ случ. образом.
2. Рассч. $H = X\Theta$ и $\nabla J = \frac{1}{m} X^T (H - Y)$.
3. Найти $\Theta^H : \Theta^H = \Theta^C - \alpha \nabla J$.
4. Повторять пункты 2-3 до выполнения одного из условий: $J^H - J^C < \delta$, #итер. > N_{max}.
5. Вывод результатов: Θ.

Полиномиальная регрессия / Polynomial regression

Регрессия / Regression

- **Линейная /** Linear
  - **Одна переменная (признак) /** One variable (feature)
  - **Мн. переменных /** Multiple variables (features)
- **Нелинейная /** Non-Linear
  - **Полиномиальная /** Polynomial
  - **Логистическая /** Logistic

$$h(\theta_0, \theta_1) = \theta_0 + \theta_1 x$$

$$X = \left(\begin{pmatrix} x^{(1)} \\ \dots \\ x^{(m)} \end{pmatrix}\right).$$

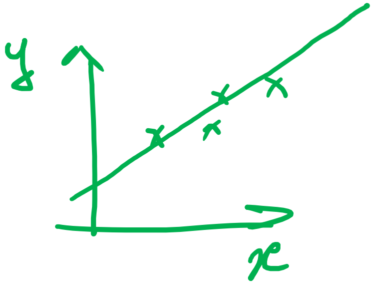
$$h(\theta_0, \theta_j) = \theta_0 + \underline{\theta_k x_k}, (j, k = 1 \dots n)$$

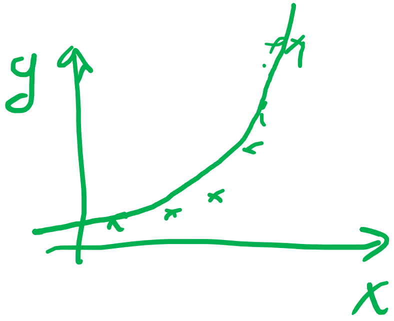
$$X = \left(\begin{pmatrix} x_1^{(1)} & x_2^{(1)} & & x_n^{(1)} \\ \dots & \dots & \dots & \dots \\ x_1^{(m)} & x_2^{(m)} & & x_n^{(m)} \end{pmatrix}\right).$$

$\longleftrightarrow$

$$h(\theta_0, \theta_j) = \theta_0 + \underline{\theta_k x^k}, (j, k = 1 \dots n)$$

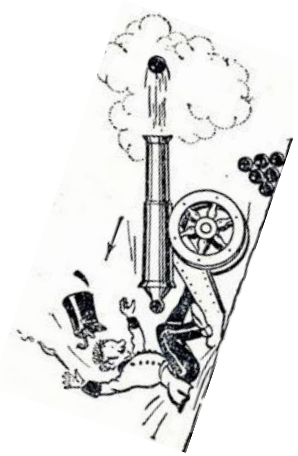
$x^i \to x_i$ $\quad$ $x^2 \to x_2$

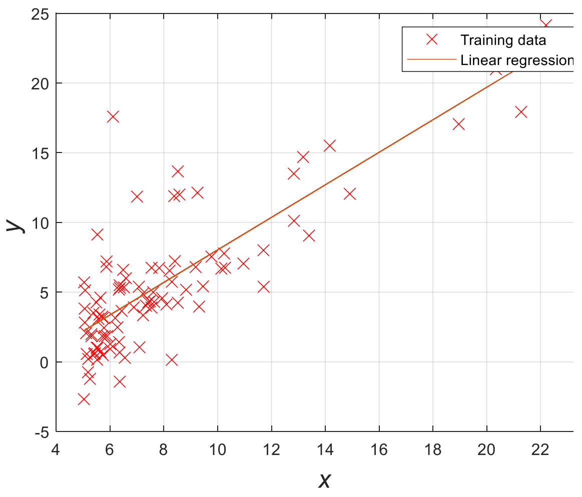


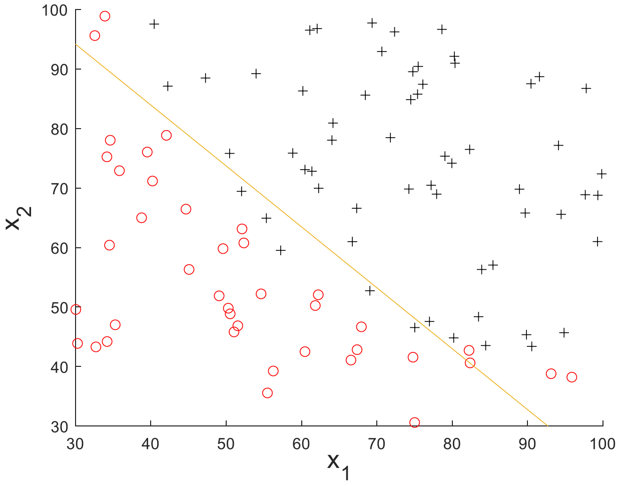


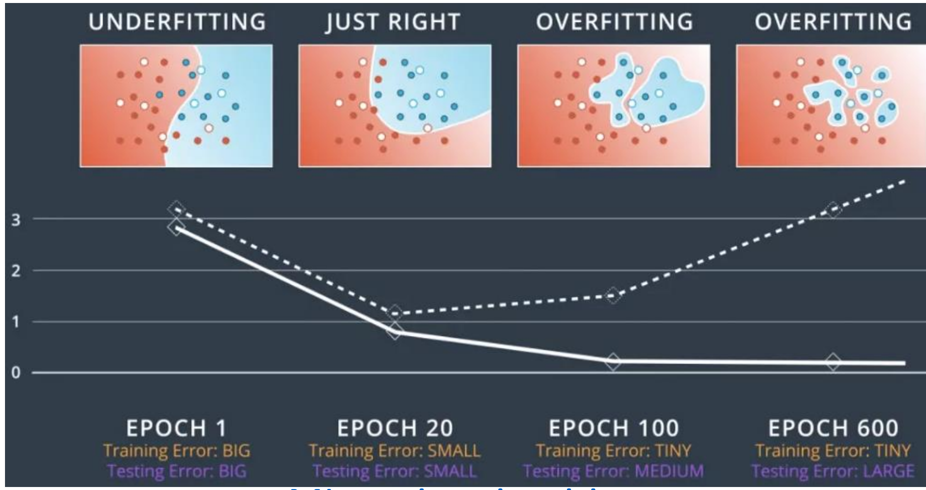
Лекция 1. Искусственный интеллект и основы машинного обучения (МО) / Artificial Intelligence (AI) and Machine Learning (ML) basics

1. **Область исследования и основные понятия / Terms**
2. **Линейная и полиномиальная регрессия / Linear and polynomial regression**
3. **Логистическая регрессия/ Logistic regression**
4. **Настройка МО/ ML settings**







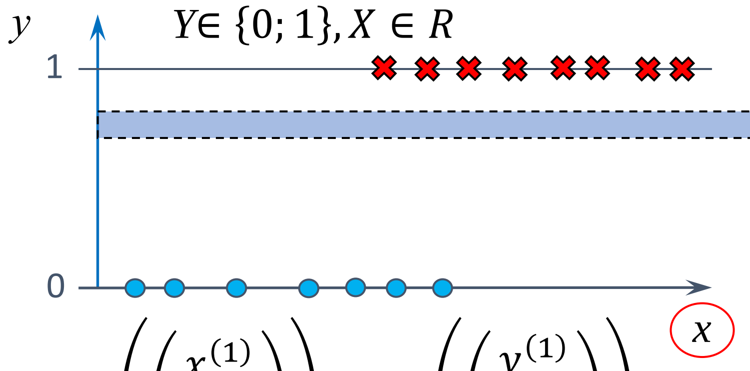


ML settings intuition

Логистическая регрессия и решение задач классификации / Logistic regression and classification

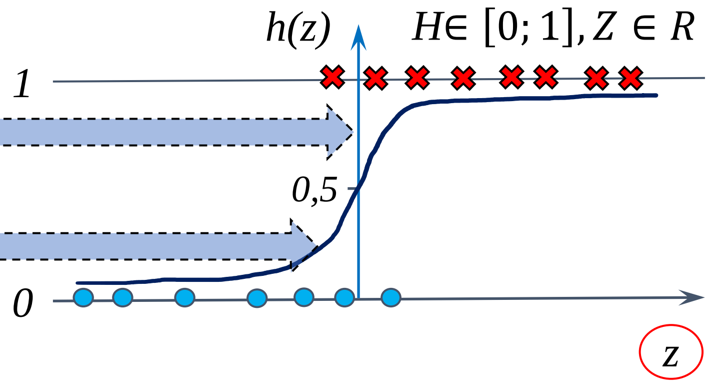
<u>Основная идея:</u> поиск наилучшей функциональной зависимости бинарного множества $Y = \left((y_i)\right)$ от множества $X = \left((x_i)\right)$ (i=1…m) по критерию минимума некоторой функции качества:

$$\boxed{J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}\ln(h^{(i)}) + (1-y^{(i)})(\ln(1-h^{(i)})\right) \Rightarrow \min.}$$



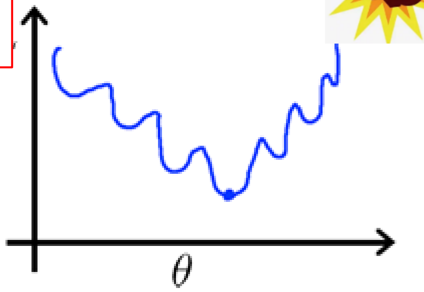
$X \to Z: z = \theta_0 + \theta_1 x.$

$Y \to H: h(z) = \dfrac{1}{1+e^{-z(x)}}.$

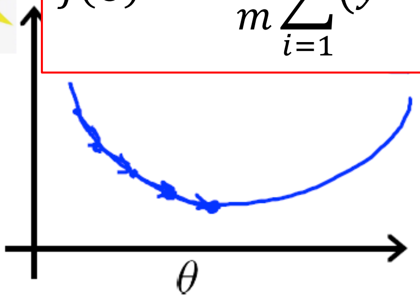


$$X = \left(\begin{pmatrix} x^{(1)} \\ \dots \\ x^{(m)} \end{pmatrix}\right); Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right); \quad \to \quad X = \left(\begin{pmatrix} 1 & x^{(1)} \\ \dots & \dots \\ 1 & x^{(m)} \end{pmatrix}\right); \ \Theta = \left(\begin{pmatrix} \theta_0 \\ \theta_1 \end{pmatrix}\right); Z = X\Theta; \quad \to \quad h(z) = \frac{1}{1+e^{-z(x)}}.$$

$$\boxed{J(\Theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h^{(i)} - y^{(i)}\right)^2 \Rightarrow \min.}$$



$$\boxed{J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}\ln(h^{(i)}) + (1-y^{(i)})(\ln(1-h^{(i)})\right) \Rightarrow \min.}$$



Логистическая регрессия и решение задач классификации / Logistic regression and classification

<u>Основная идея:</u> поиск наилучшей функциональной зависимости бинарного множества $Y = \big((y_i)\big)$ от множества $X = \big((x_i)\big)$ (i=1…m) по критерию минимума некоторой функции качества:

$$\boxed{J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\big(y^{(i)}\ln(h^{(i)}) + (1-y^{(i)})(\ln(1-h^{(i)})\big) \Rightarrow \min.}$$

$$X = \left(\begin{pmatrix} x^{(1)} \\ \dots \\ x^{(m)} \end{pmatrix}\right);\ Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right); \quad \rightarrow \quad X = \left(\begin{pmatrix} 1 & x^{(1)} \\ \dots & \dots \\ 1 & x^{(m)} \end{pmatrix}\right);\ \Theta = \left(\begin{pmatrix} \theta_0 \\ \theta_1 \end{pmatrix}\right); Z = X\Theta; \quad \rightarrow \quad h(z) = \frac{1}{1+e^{-z(x)}}, \rightarrow$$

$$\rightarrow J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\big(y^{(i)}\ln(h^{(i)}) + (1-y^{(i)})(\ln(1-h^{(i)})\big) \Rightarrow \min, \rightarrow \qquad \boxed{\nabla J = \frac{1}{m}X^T(H-Y).}$$

Алгоритм поиска минимума функции качества.

1. Задать нач.зн. Θ случ. образом.
2. Рассч. $H = X\Theta$ и $\nabla J = \frac{1}{m}X^T(H-Y)$.
3. Найти Θ^H : $\Theta^H = \Theta^C - \alpha\nabla J$.
4. Повторять пункты 2-3 до выполнения одного из условий: $J^H - J^C < \delta$, #итер. > N_{max}.
5. Вывод результатов: Θ.

Логистическая регрессия и решение задач классификации / Logistic regression and classification

- **Регрессия** / Regression
  - **Линейная /** Linear
    - **Одна переменная (признак) /** One variable (feature)
    - **Мн. переменных /** Multiple variables (features)
  - **Нелинейная /** Non-Linear
    - **Полиномиальная /** Polynomial
    - **Логистическая /** Logistic
      - **Линейная *z* /** Linear
        - **Одна переменная (признак) /** One variable (feature)
        - **Мн. переменных /** Multiple variables (features)
      - **Нелинейная *z* /** Non-Linear
        - **Полиномиальная /** Polynomial

Логистическая регрессия и решение задач классификации / Logistic regression and classification

Логистическая р. / Logistic r.

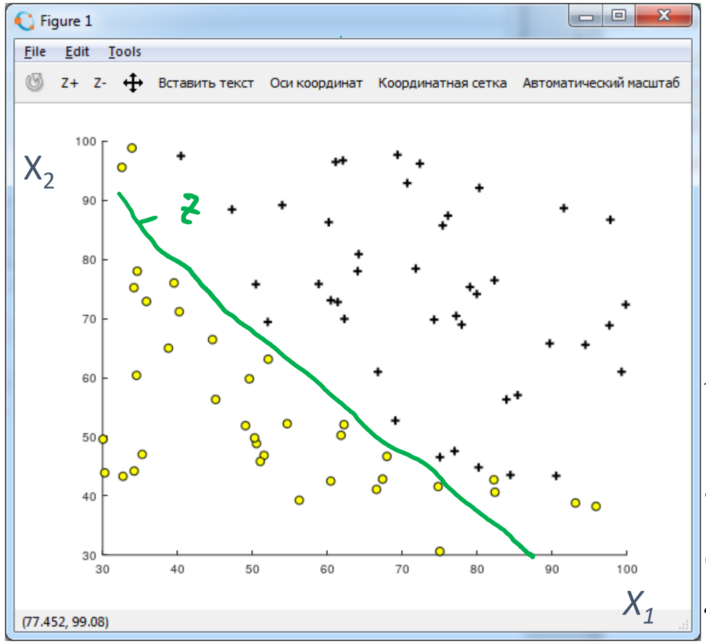
1 перем. x / 1 variable x

Мн.перем.x_k / Multiple var. x_k $(k = 1 \dots n)$

Полином. / Polynomial

$n=2$

$$X = \left(\begin{pmatrix} x_1^{(1)} & x_2^{(1)} \\ \dots & \dots \\ x_1^{(m)} & x_2^{(m)} \end{pmatrix}\right); Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right); \quad \rightarrow \quad X = \left(\begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} \\ \dots & \dots & \dots \\ 1 & x_1^{(m)} & x_2^{(m)} \end{pmatrix}\right); \Theta = \left(\begin{pmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \end{pmatrix}\right); Z = X\Theta; \quad \rightarrow \quad h(z) = \frac{1}{1+e^{-z}}.$$



$$\boxed{J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} \ln(h^{(i)}) + (1 - y^{(i)})(\ln(1 - h^{(i)})\right) \Rightarrow \min.}$$

$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2 = 0$; $\underline{x_2 = -\frac{\theta_0}{\theta_2} - \frac{\theta_1}{\theta_2} x_1}$

Алгоритм поиска минимума функции качества.

1. Задать нач.зн. Θ сл. образом.
2. Рассч. $H = X\Theta$ и $\nabla J = \frac{1}{m} X^T (H - Y)$.
3. Найти Θ^H : $\Theta^H = \Theta^C - \alpha \nabla J$.
4. Повторять пункты 2-3 до выполнения одного из условий: $J^H - J^C < \delta$, #итер. $> N_{max}$.
5. Вывод результатов: Θ. $z = 0$

Логистическая регрессия и решение задач классификации / Logistic regression and classification

Логистическая р. / Logistic r.

- **1 перем. x** / 1 variable x
- **Мн.перем.x_k** / Multiple var. x_k $(k = 1 \dots n)$
- **Полином.** / Polynomial

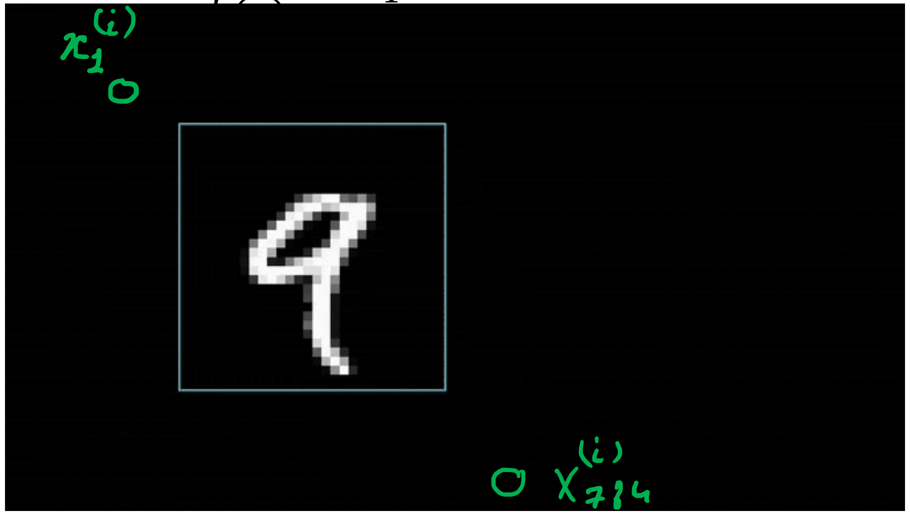
0 - не 9
1 - 9

$$X = \left(\left(\begin{matrix} x_1^{(1)} & x_2^{(1)} & & x_n^{(1)} \\ \dots & \dots & \dots & \dots \\ x_1^{(m)} & x_2^{(m)} & & x_n^{(m)} \end{matrix}\right)\right); \; Y = \left(\left(\begin{matrix} y^{(1)} \\ \dots \\ y^{(m)} \end{matrix}\right)\right);$$

1н 2н ....н нуле

$$\to X = \left(\left(\begin{matrix} 1 & x_1^{(1)} & x_2^{(1)} & & x_n^{(1)} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_1^{(m)} & x_2^{(m)} & & x_n^{(m)} \end{matrix}\right)\right); \; \Theta = \left(\left(\begin{matrix} \theta_0 \\ \dots \\ \theta_n \end{matrix}\right)\right); Z = X\Theta; \quad \to h(z) = \frac{1}{1+e^{-z}}.$$

785



Gray scale picture of "Nine"

$$\boxed{J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)} \ln(h^{(i)}) + (1 - y^{(i)})(\ln(1 - h^{(i)})\right) \Rightarrow \min.}$$

Алгоритм поиска минимума функции качества.

1. Задать нач.зн. Θ случ. образом.
2. Рассч. $H = X\Theta$ и $\nabla J = \frac{1}{m} X^T (H - Y)$.
3. Найти $\Theta^H : \Theta^H = \Theta^C - \alpha \nabla J$.
4. Повторять пункты 2-3 до выполнения одного из условий: $J^H - J^C < \delta$, #итер. $> N_{max}$.
5. Вывод результатов: Θ.

Логистическая регрессия и решение задач классификации / Logistic regression and classification

Логистическая р. / Logistic r.

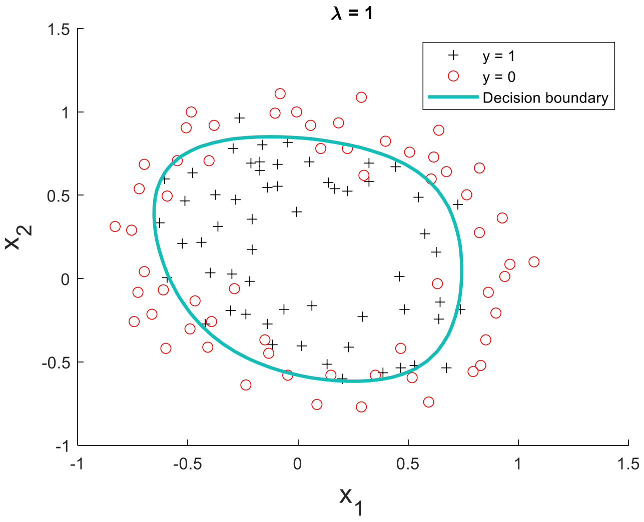
1 перем. x / 1 variable x | **Мн.перем.x_k** / Multiple var. x_k | **Полином.** / Polynomial

$$X = \left(\begin{pmatrix} x_1^{(1)} & x_2^{(1)} \\ \dots & \dots \\ x_1^{(m)} & x_2^{(m)} \end{pmatrix}\right); Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right);$$

$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1 x_2 + \theta_4 x_2 + \theta_5 x_2^2$$

$$h(z) = \frac{1}{1+e^{-z}}.$$

$$\boxed{J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}\ln(h^{(i)}) + (1-y^{(i)})(\ln(1-h^{(i)})\right) \Rightarrow \min.}$$

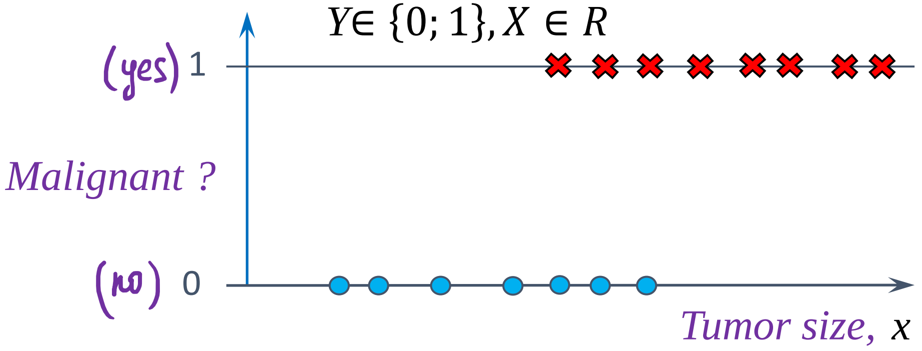


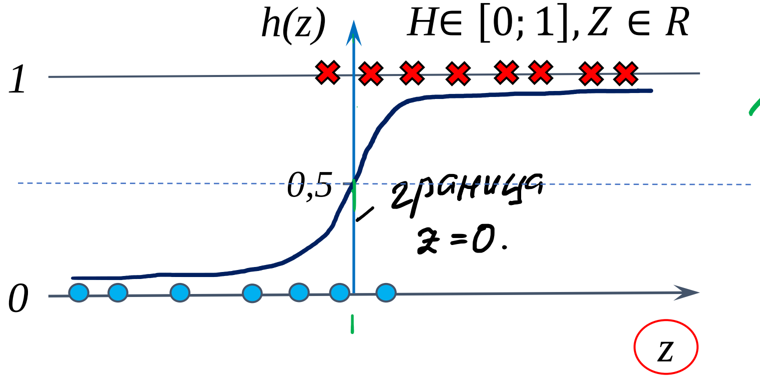
Алгоритм поиска минимума функции качества.

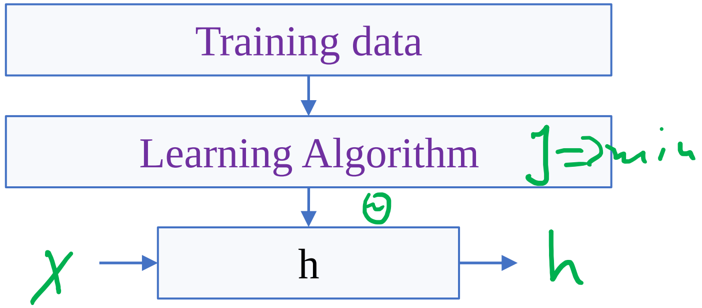
1. Задать нач.зн. Θ случ. образом.
2. Рассч. $H = X\Theta$ и $\nabla J = \frac{1}{m}X^T(H - Y)$.
3. Найти Θ^H : $\Theta^H = \Theta^C - \alpha\nabla J$.
4. Повторять пункты 2-3 до выполнения одного из условий: $J^{H} - J^{C} < \delta$, #итер. $> N_{max}$.
5. Вывод результатов: Θ.

Логистическая регрессия и решение задач классификации / Logistic regression and classification

Классификация. Граница принятия решений / Classification. Decision boundary



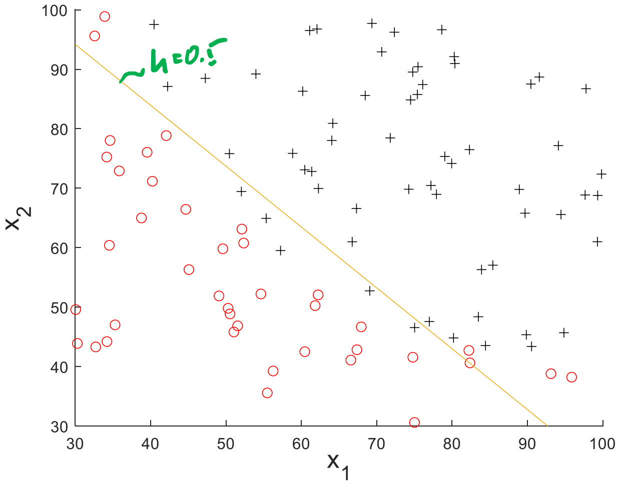


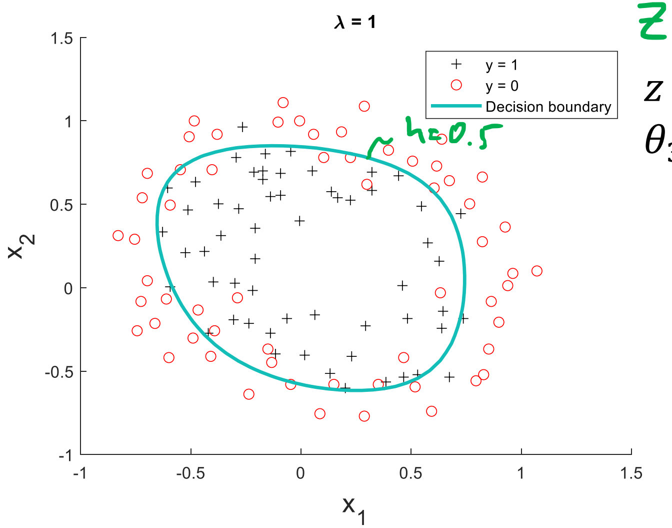


if $h \geq 0.5$, 1

else $h < 0.5$, 0

$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2.$$





$z = 0$

$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 + \theta_5 x_2^2.$$

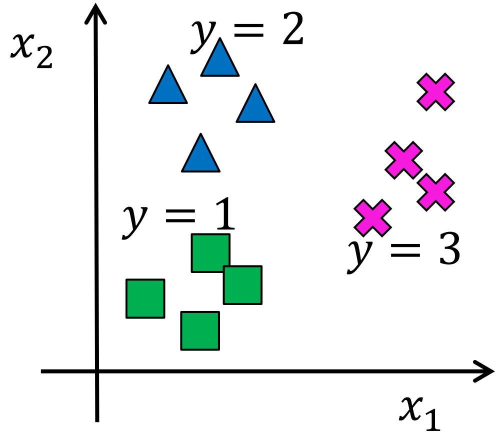
Логистическая регрессия и решение задач классификации / Logistic regression and classification

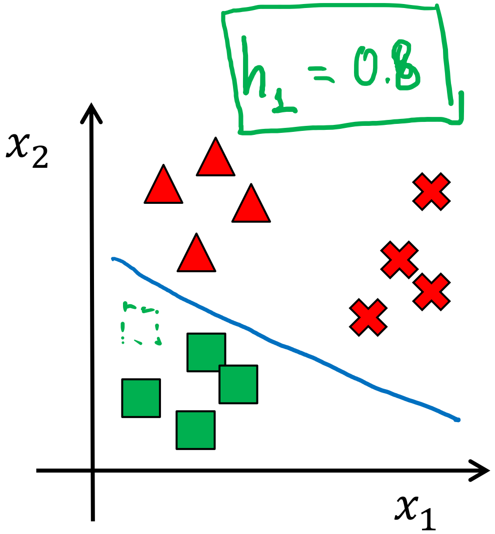
Многоклассовая классификация: один против всех / Multi-class classification: one-vs-all

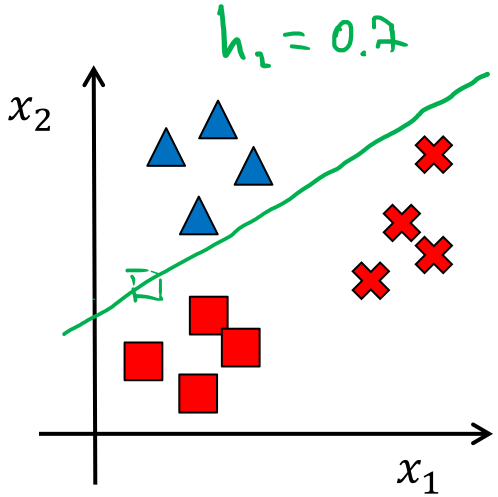
<u>Основная идея:</u> поиск наилучшей функциональной зависимости мультиарного (p-арного) множества $Y = \left((y_i)\right)$, $Y \in \{1\ 2\ 3\ \dots p\}$, от множества $X = \left((x_i)\right)$, $X \in R$, (i=$1\dots m$) по критерию минимума некоторой функции качества $J(\Theta)$ путем последовательного решения "p" задач двухклассовой классификации ($p > 2$).

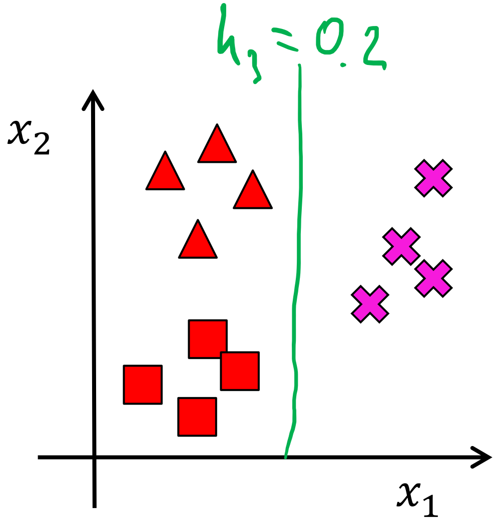
Пример. Диагноз: здоров, ОРВИ, грипп (3 класса).

$$X = \left(\begin{pmatrix} x_1^{(1)} & x_2^{(1)} \\ \dots & \dots \\ x_1^{(m)} & x_2^{(m)} \end{pmatrix}\right); Y = \left(\begin{pmatrix} y^{(1)} \\ \dots \\ y^{(m)} \end{pmatrix}\right), Y \in \{1\ 2\ 3\}.$$



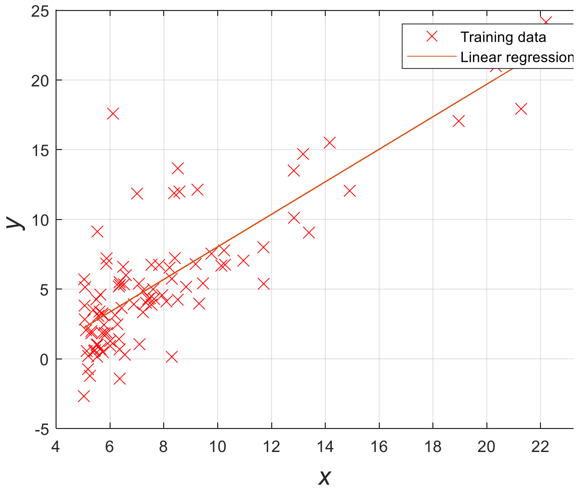


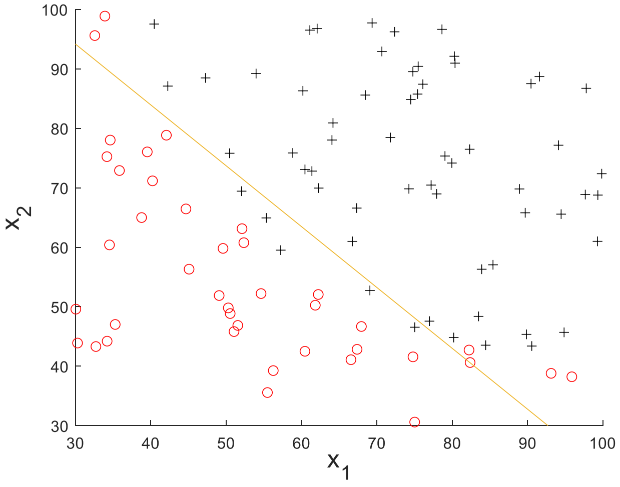


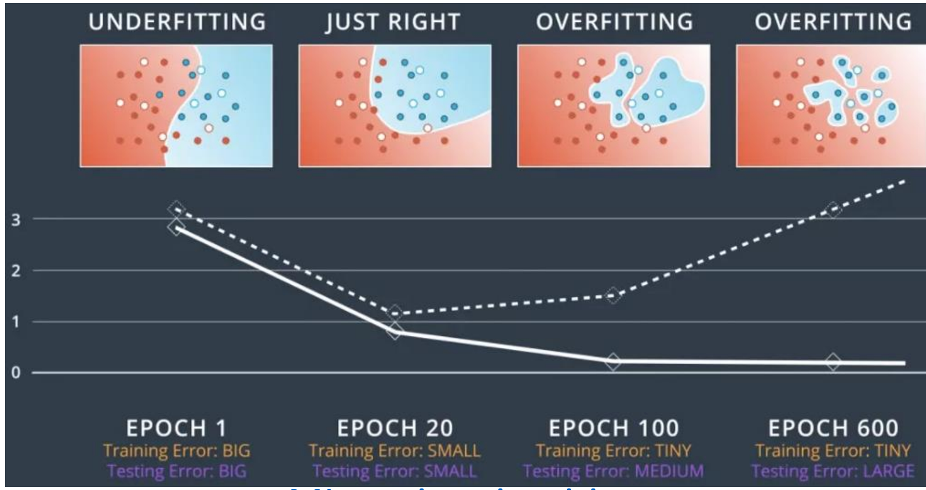


Лекция 1. Искусственный интеллект и основы машинного обучения (МО) / Artificial Intelligence (AI) and Machine Learning (ML) basics

1. Область исследования и основные понятия / Terms
2. Линейная и полиномиальная регрессия / Linear and polynomial regression
3. Логистическая регрессия/ Logistic regression
4. Настройка МО/ ML settings







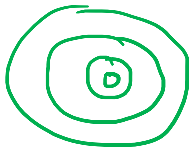
[ML settings intuition]

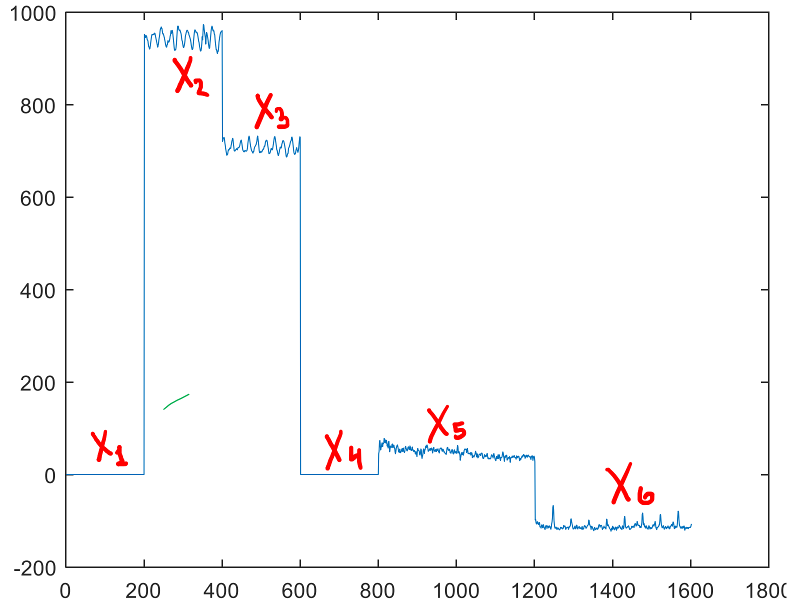
Настройка моделей МО/ ML settings

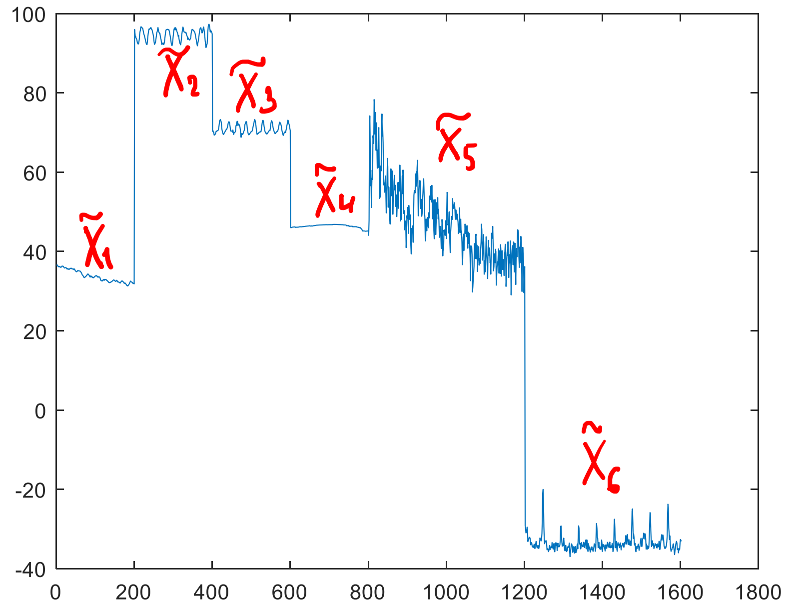
Параметры модели определяются в ходе решения задачи МО. Например, в задачах регрессии параметрами являются компоненты матрицы Θ.
Гиперпараметры задаются пользователем, как правило не единственным образом, и их значения влияют на значения искомых параметров.

1. Масштабирование признаков / Feature Scaling

$$\tilde{x}_i = \frac{x_i}{\max(x_i)}$$







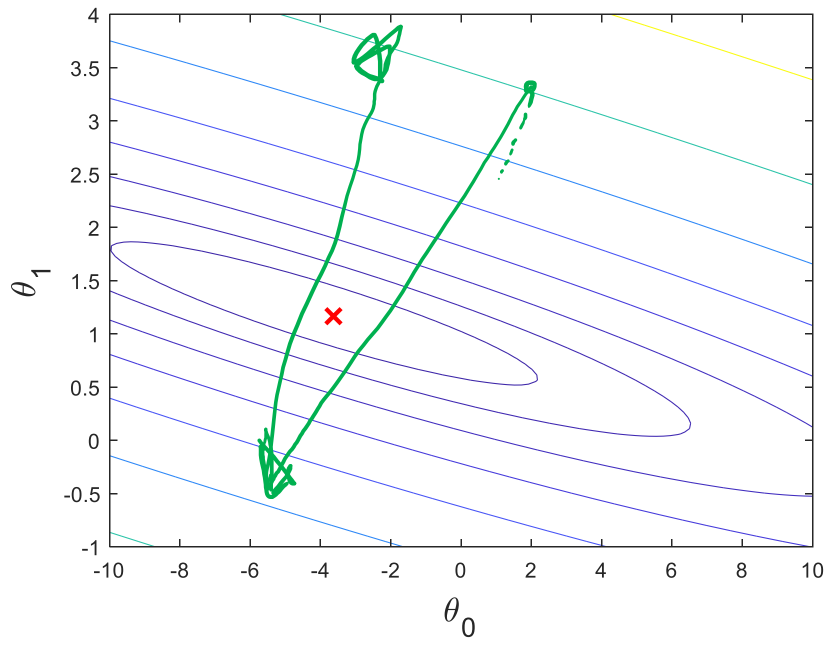
Настройка моделей МО/ ML settings

Параметры модели определяются в ходе решения задачи МО. Например, в задачах регрессии параметрами являются компоненты матрицы Θ.
Гиперпараметры задаются пользователем, как правило не единственным образом, и их значения влияют на значения искомых параметров.

2. Скорость обучения α / Learning rate

$\alpha \cdot \nabla J$

3. Погрешность δ и количество итераций N_{max} / Error and # of iterations



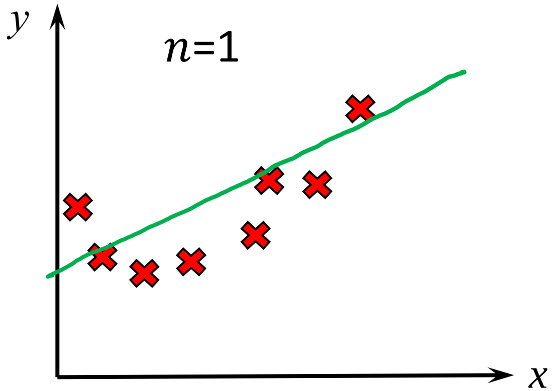
Настройка моделей МО/ ML settings

Параметры модели определяются в ходе решения задачи МО. Например, в задачах регрессии параметрами являются компоненты матрицы Θ.
Гиперпараметры задаются пользователем, как правило не единственным образом, и их значения влияют на значения искомых параметров.

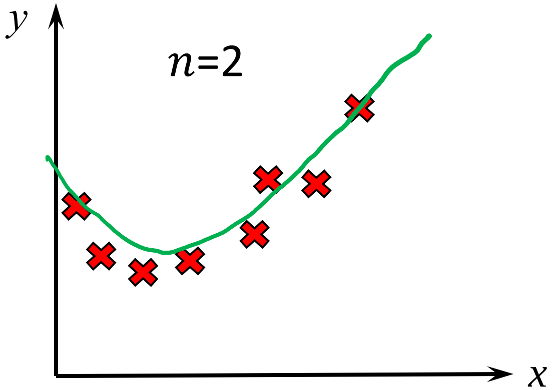
4. Регуляризация / Regularization

$$J(\theta_0, \theta_j) = \frac{1}{2m}\left[\sum_{i=1}^{m}\left(h^{(i)} - y^{(i)}\right)^2 + \lambda \sum_{j=1}^{n} \theta_j^2\right] \Rightarrow \min.$$

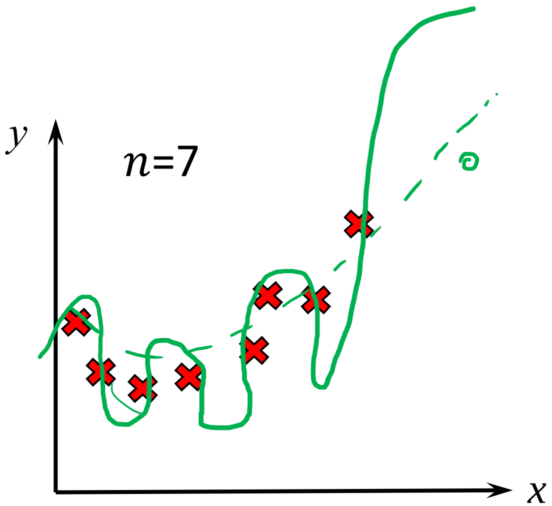
$$h(\theta_0, \theta_j) = \theta_0 + \theta_k x^k,$$
$$(j, k = 1 \ldots n).$$



Недообучение / Underfit, High bias



Норм. / Appropriate fiting



Переобучение / Overfit, High variance

Настройка моделей МО/ ML settings

Параметры модели определяются в ходе решения задачи МО. Например, в задачах регрессии параметрами являются компоненты матрицы Θ.
Гиперпараметры задаются пользователем, как правило не единственным образом, и их значения влияют на значения искомых параметров.

70% 15% 15% Data Set

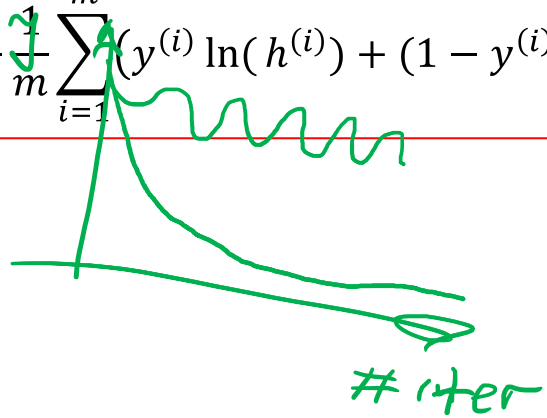
Training Validation Testing

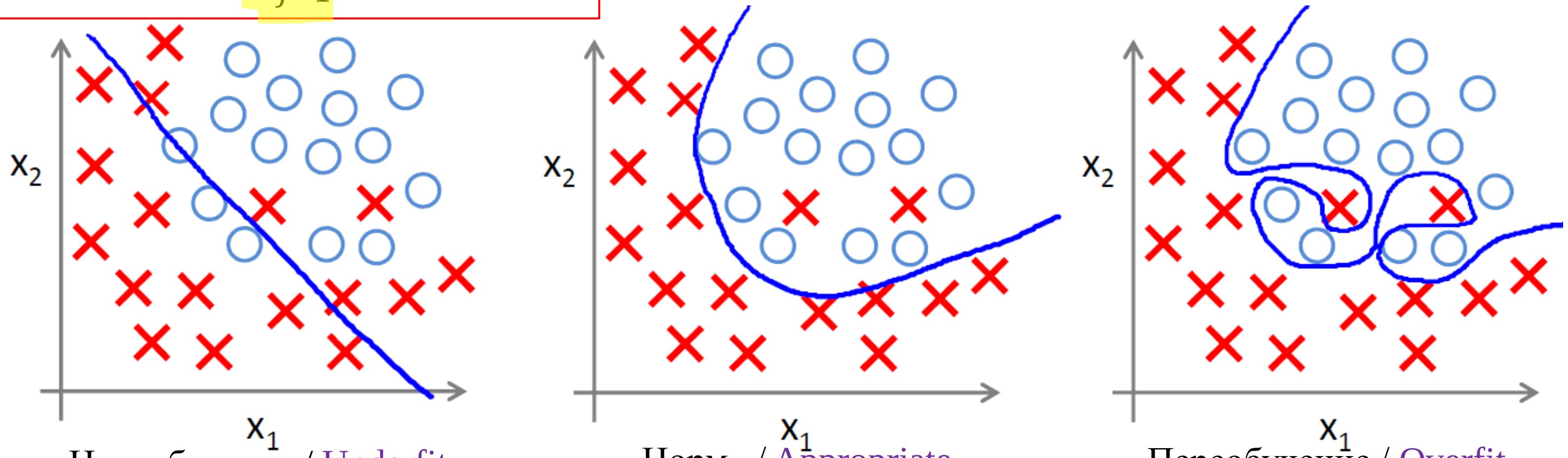
λ

1. Масштабирование признаков / Feature Scaling
2. Скорость обучения α / Learning rate
3. Погрешность δ и количество итераций $N_{\max}$ / Error and # of iterations
4. Регуляризация / Regularization

$$J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}\ln(h^{(i)}) + (1-y^{(i)})(\ln(1-h^{(i)}))\right) + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2 \Rightarrow \min.$$

∇J





Недообучение / Underfit, High bias

Норм. / Appropriate fitting

Переобучение / Overfit, High variance

What is underfitting and overfitting

Самостоятельная работа / Homework

Вопросы:

1. Почему не рекомендуется дифференцировать функции, полученные в результате регрессионного анализа данных?
2. Почему нельзя сокращать «d» в выражении «dy/dx»?
3. В каких случаях математическое ожидание не совпадает со средним арифметическим значением?
4. Каким образом можно улучшить метод градиентного спуска, чтобы находить с его помощью глобальные минимумы, вместо локальных?
5. Можно ли рассмотренные задачи линейной регрессии решить аналитически, без применения метода градиентного спуска?
6. Почему при построении регрессионных моделей обычно не рекомендуется применение полиномов высоких степеней?