Онлайн-курс

# ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

## 2. Точки минимума функций многих переменных

Автор: Шевляков Артём Николаевич

# Функции многих переменных

# Функции многих переменных (ФМП)

**Вы думали, что функции зависят только от одного аргумента?**

**А вот и нет!**

# Функции многих переменных (ФМП)

**Самый простой пример ФМП – это величина «сумма чека в советском магазине».**

**S=0.2*x+0.34*y**

где  x - количество булок хлеба,

y - количество бутылок молока

# Функции многих переменных (ФМП)

**Самый простой пример ФМП – это величина «сумма чека в советском магазине».**

**S=0.2*x+0.34*y-0.15*z**

где x - количество булок хлеба,

y - количество бутылок молока,

# Функции многих переменных (ФМП)

**Самый простой пример ФМП – это величина «сумма чека в советском магазине».**

**S=0.2*x+0.34*y-0.15*z**

где x - количество булок хлеба,

y - количество бутылок молока,

z - число сдаваемых бутылок.

# Функции многих переменных (ФМП)

**Более серьёзный пример. «Площадь прямоугольника» зависит от длин двух его сторон:**

$$f(x_1, x_2) = x_1 x_2$$

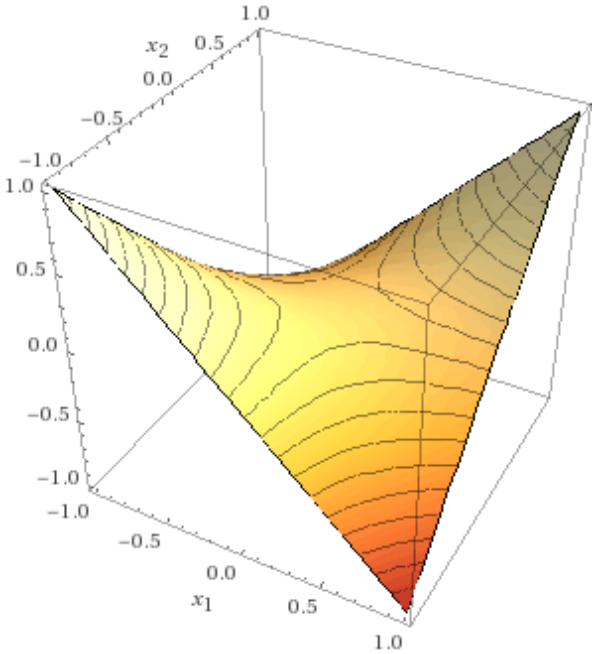
**А как можно представлять себе ФМП?**
График в виде кривой не годится.

Они представляются в виде поверхностей: двумерных (их еще можно нарисовать) или многомерных.

# ФМП  в нашем мозгу

Вот поверхность, задающая функцию $f(x_1, x_2) = x_1 x_2$

Аргументы лежат в плоскости «пола». Над каждой «половой» точкой на нужной высоте расположено значение функции.

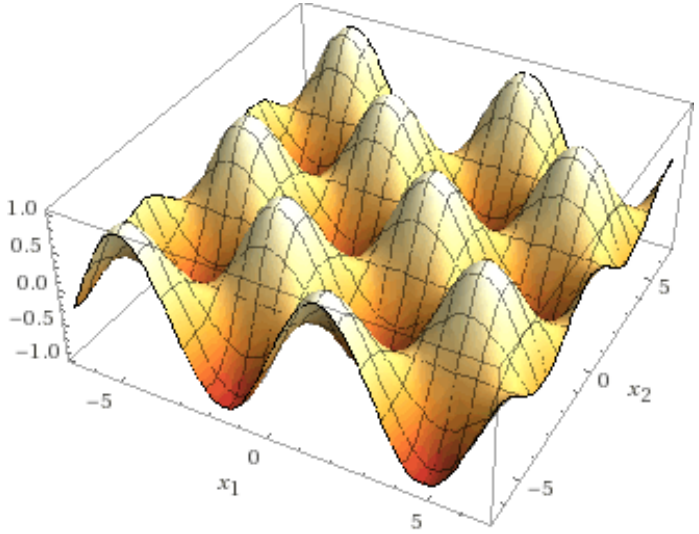


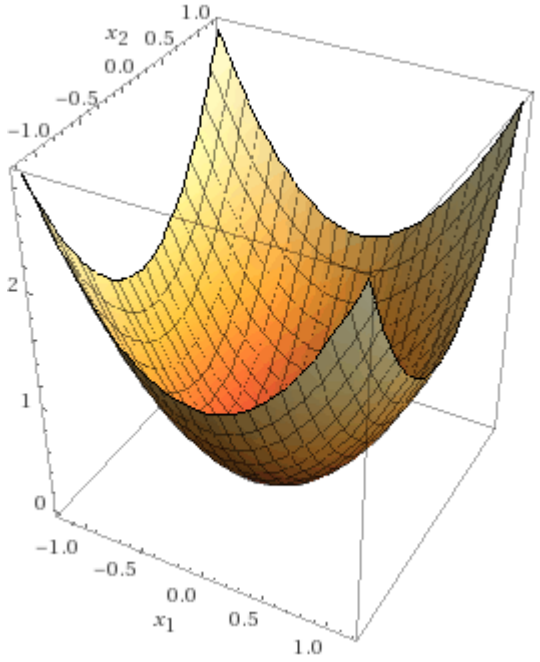
# ФМП в нашем мозгу

Вот поверхность, задающая функцию $f(x_1, x_2) = x_1^2 + x_2^2$

Самая «низкая» точка этой поверхности *(0,0)*.

А вот еще поверхность некоторой ФМП:



# Точка минимума ФМП

Точка $(a_1,a_2)$ является точкой **локального минимума** ФМП $f(x_1,x_2)$, если в некоторой окрестности этой точки выполнено неравенство

$$f(a_1, a_2) < f(x_1, x_2)$$

для всех точек $(x_1,x_2)$ **из данной окрестности.**

Точка $(a_1,a_2)$ является точкой **глобального минимума** ФМП $f(x_1,x_2)$, если

$$f(a_1, a_2) < f(x_1, x_2)$$

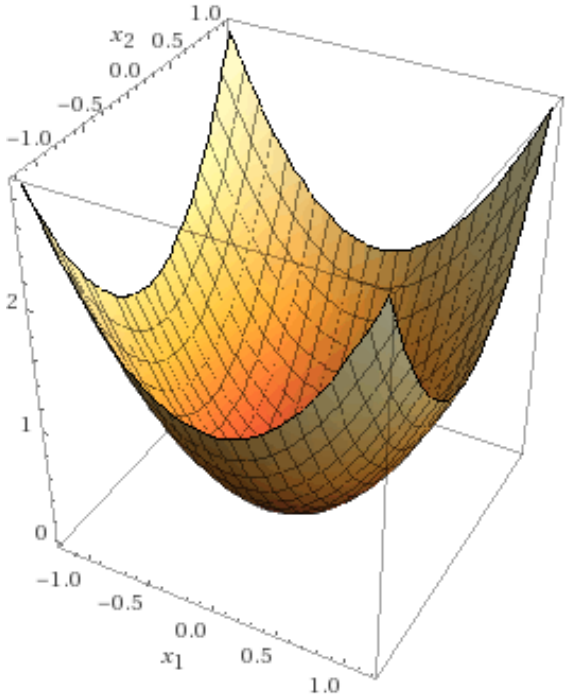
для всех точек $(x_1,x_2)$ **из области определения функции** $f(x_1,x_2)$.

# Пример минимума ФМП

Вот поверхность, задающая функцию $f(x_1, x_2) = x_1^2 + x_2^2$

Точка этой поверхности *(0,0)* является точкой локального минимума.

**На самом деле точка (0,0) – это глобальный минимум функции** $f(x_1, x_2) = x_1^2 + x_2^2$ .



# Точка минимума ФМП

Точка $(a_1,\ldots,a_n)$ является точкой **локального минимума** ФМП $f(x_1,\ldots,x_n)$, если в некоторой окрестности этой точки выполнено неравенство $f(a_1,\ldots,a_n) < f(x_1,\ldots,x_n)$ для всех точек $(x_1,\ldots,x_n)$ из данной окрестности.


Точка $(a_1,\ldots,a_n)$ является точкой **глобального минимума** ФМП $f(x_1,\ldots,x_n)$, если $f(a_1,\ldots,a_n) < f(x_1,\ldots,x_n)$ для всех точек $(x_1,\ldots,x_n)$ из области определения функции $f(x_1,\ldots,x_n)$.

# ФМП без минимума

**А вот у функции «сумма чека в советском магазине»**

**S=0.2*x+0.34*y-0.15*z**

минимума нет.

Ибо с ростом числа бутылок функция неограниченно убывает.

# Где глобальный минимум на этом фото?

**Да вот же он!**

А можете найти еще пару локальных минимумов на этом фото?

# Потренируемся искать минимумы

Представьте, что верхняя поверхность вашего тела (при условии, что вы лежите на спине)– это поверхность некоторой функции от двух аргументов.

**Где у неё локальные и глобальные минимумы?**

# Главная задача!

**Нужно уметь находить у ФМП глубокие минимумы. В идеале – найти глобальный минимум.** А как это сделать? Даже если мы нашли один локальный минимум, то как мы можем быть уверены в том, что нет минимумов глубже его?

Как по этому фото понять, что где-то существует Марианская впадина?

Онлайн-курс

# ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

## 2. Точки минимума функций многих переменных

# Производные функций многих переменных

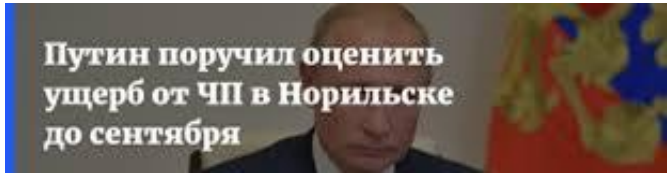
# Производная? Ну куда же без нее!

Пусть $f(x_1,x_2)$ - функция от двух аргументов. Оказывается, у нее существует не одна, а целых две производных:

$\frac{\partial f}{\partial x_1}$ **частная производная** по аргументу $x_1$,

$\frac{\partial f}{\partial x_2}$ **частная производная** по аргументу $x_2$

Частная производная (ЧП) считается по обычным правилам **в предположении, что все остальные аргументы функции являются константами.**



# Примеры частных производных

Пусть $f(x_1, x_2) = x_1 x_2$, тогда $\dfrac{\partial f}{\partial x_1} = x_2, \ \dfrac{\partial f}{\partial x_2} = x_1$

Если $f(x_1, x_2) = x_1^2 - 5x_2^2 + 2x_1 x_2 + 3x_1 - x_2 + 10$

, то

$$\frac{\partial f}{\partial x_1} = 2x_1 + 2x_2 + 3,$$

$$\frac{\partial f}{\partial x_2} = -10x_2 + 2x_1 - 1$$

# Частные производные ФМП

**Если ФМП $f(x_1,\ldots,x_n)$ зависит от *n* аргументов, то у нее будет *n* частных производных (ЧП)**


$$\frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n}$$

# Свойства ЧП

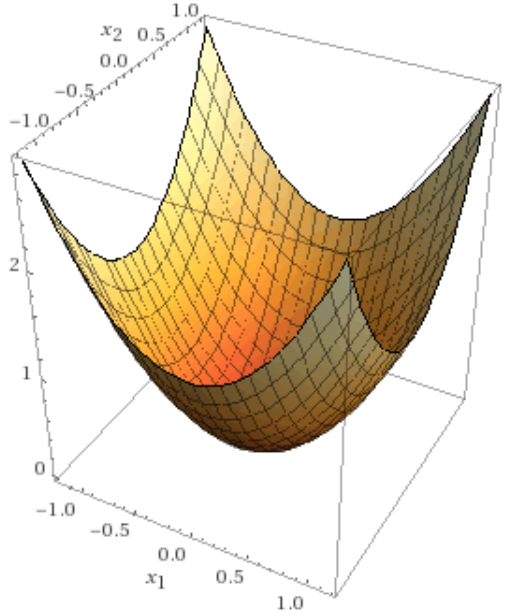
Если точка $(a_1,a_2)$ является точкой локального минимума ФМП $f(x_1,x_2)$, то **все ЧП равны нулю в этой точке**!

Для функций одного аргумента, это, конечно же, тоже выполнено.

Например, все ЧП

$$f(x_1, x_2) = x_1^2 + x_2^2$$

равны *0* в точке *(0,0)*.



# Градиент

**Вектор** $\nabla f = (\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2})$ **для ФМП** $f(x_1,x_2)$ **называется градиентом**.

Для ФМП от *n* переменных и градент подлиннее:

$$\nabla f = (\frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n})$$

Онлайн-курс

# ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

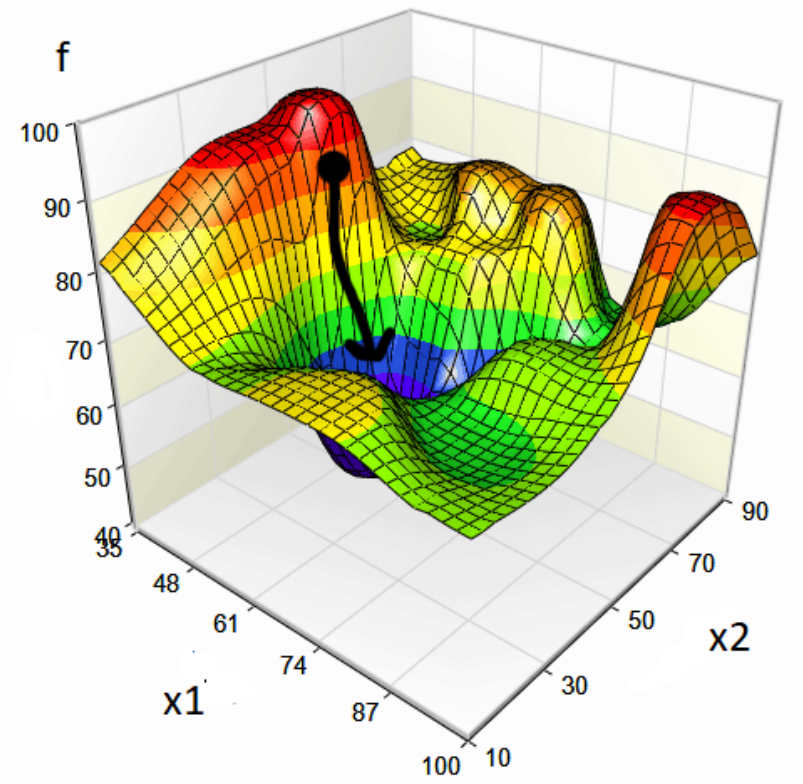
## 2. Точки минимума функций многих переменных

# Градиентный спуск для функций многих переменных

# Ищем минимум по шагам

Неужели мы глупее шарика?

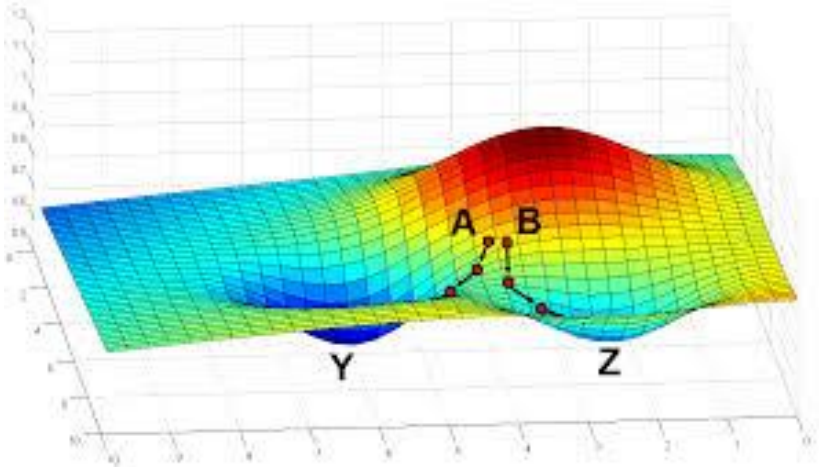
Мы будем снова моделировать спуск шарика. Но на этот раз движется по поверхности, а не по одномерному графику функции (как в первой главе курса).



# Достижимость глобального минимума

Глобальный минимум не обязательно будет достигнут при таком спуске.

Позиция шарика перед началом спуска существенно влияет на результат (это было справедливо и для функции одной переменной).



# Градиентный спуск (ГС)

ГС для ФМП ничем **принципиально не отличается** от ГС для функции одного аргумента («Математика — сила!»).

Вспомним ГС для одного аргумента:

Выберем длину шага *h*. Пусть шарик находится в точке с координатой $a$. Тогда новая *a* точка равна:

$$a := a - f'(a) * h$$

После этого процесс повторяется, мы получаем новую позицию для шарика итд.

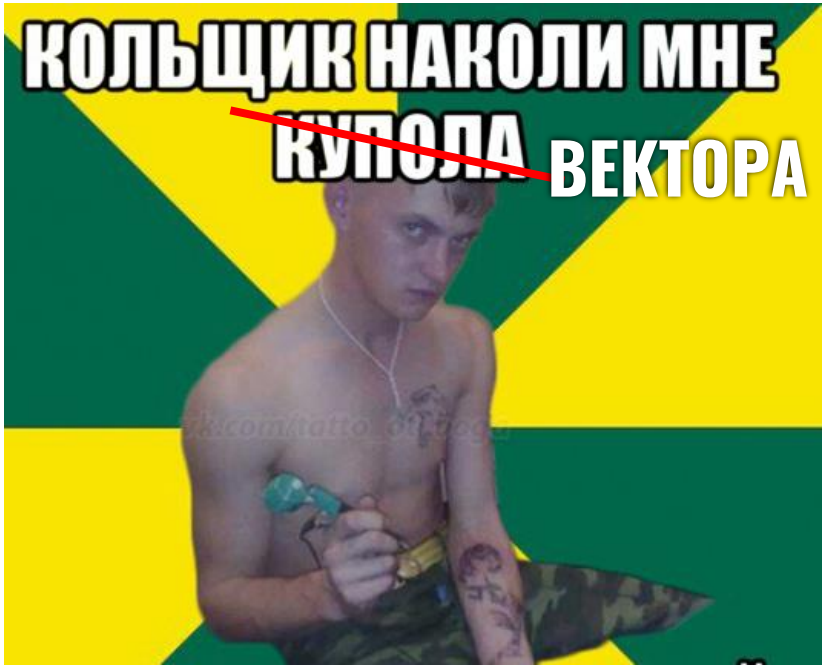
# Градиентный спуск (ГС)

А теперь я просто заменю числа на векторы в формуле для ГС:

$$a := a - f'(a) * h$$

Будет так:

$$\bar{a} := \bar{a} - h\nabla f(\bar{a})$$



# Градиентный спуск для ФМП

Выберем длину шага *h* (например, *h=0.01*). Пусть шарик находится в точке с координатой $(a_1, a_2)$ . Тогда новые координаты равны:

$$(a_1, a_2) := (a_1, a_2) - h(\frac{\partial f}{\partial x_1}(a_1, a_2), \frac{\partial f}{\partial x_2}(a_1, a_2))$$

$$a_1 := a_1 - h \cdot \frac{\partial f}{\partial x_1}(a_1, a_2)$$

$$a_2 := a_2 - h \cdot \frac{\partial f}{\partial x_2}(a_1, a_2)$$

После этого процесс повторяется, мы получаем новую позицию для шарика и т.д.

# Градиентный спуск (ГС)

**Если $\bar{a}_n$ – позиция нашего шарика на *n*-м шаге, то новая позиция равна:**

$$\bar{a}_{n+1} = \bar{a}_n - h\nabla f(\bar{a}_n)$$


Стандартное значение шага *h=0.01*

# Пример ГС

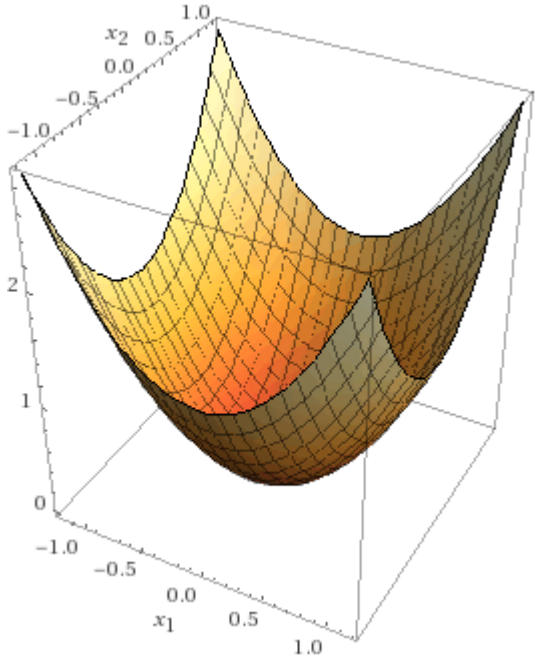
Проведем несколько итераций ГС для ФМП $f(x_1, x_2) = x_1^2 + x_2^2$

Пусть *h=0.1* и начальная позиция равна $\bar{a}_0 = (1, 2)$

Мы знаем, что (0,0) - точка минимума,

посмотрим, как ГС её найдёт.

Градиент: $\nabla f = (2x_1, 2x_2)$



# Шаг 1

Значение градиента в точке $\bar{a}_0 = (1, 2)$ равно

$$\nabla f(\bar{a}_0) = (2 \cdot 1, 2 \cdot 2) = (2, 4)$$

Новая позиция шарика:

$$\bar{a}_1 = \bar{a}_0 - h \cdot \nabla f(\bar{a}_0) = (1, 2) - 0.1 \cdot (2, 4) = (0.8, 1.6)$$

А ГС умный! Он понял, что по второму аргументу мы дальше от точки минимума, и поэтому нужно сдвинуться дальше, чем по первому аргументу.

# Шаг 2

Значение градиента в точке $\bar{a}_1 = (0.8, 1.6)$ равно

$$\nabla f(\bar{a}_1) = (2 \cdot 0.8, 2 \cdot 1.6) = (1.6, 3.2)$$

Новая позиция шарика:

$$\bar{a}_2 = \bar{a}_1 - h \cdot \nabla f(\bar{a}_1) =$$
$$= (0.8, 1.6) - 0.1 \cdot (1.6, 3.2) = (0.64, 1.28)$$

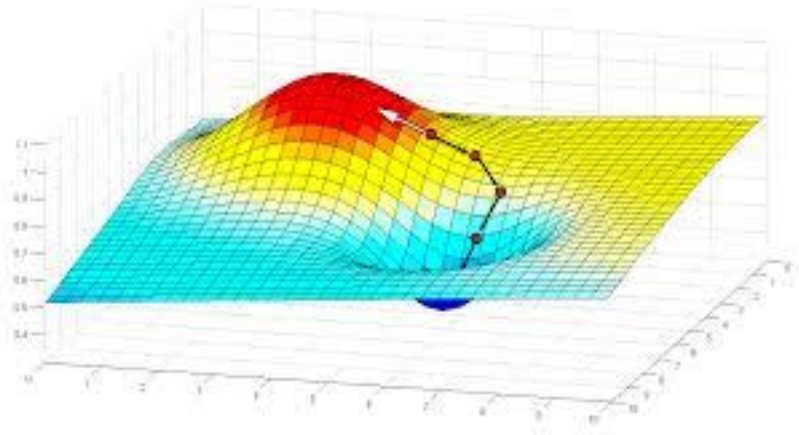
Мы ещё ближе к точке минимума!

И так далее...

# Геометрический смысл градиента

**Градиент (как вектор) показывает направление наибольшего возрастания функции.**

А мы специально идем в противоположном направлении, чтобы спуститься как можно глубже (из-за этого в формуле ГС стоит знак «минус»).



Онлайн-курс

# ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

## 2. Точки минимума функций многих переменных

# Модификации градиентного спуска

# Вернемся к поиску глубоких минимумов

**Как понять, что где-то есть впадины поглубже?**

Совет из фантастики: нужен улавливатель гравитационных волн!

То есть во время ГС нужно не просто катиться под уклон, а смотреть по сторонам и можно анализировать уже пройденный путь.

# 1. Модификации с шагом *h*

**Постановка задачи**: вам **побыстрее** хочется провалиться в канализационный колодец, расположенный в 10 метрах от вас. Согласитесь, что разумнее сначала сделать несколько длинных шагов, а потом корректировать движение с помощью более мелких шагов.

**Идея**: на первых итерациях градиентного спуска шаг $h$ может быть большим, а потом нужно его плавно уменьшать.

**По каким формулам уменьшают длину шага *h*?**

# 1. Модификации с шагом *h*

**Первая формула:** $h_{n+1} := h_0 \cdot (1 - \frac{n}{T})$

**Смысл:** Перед началом ГС у нас шаг $h_0$. После каждой итерации шаг равномерно уменьшается.

**Сюрприз:** после *T* итераций (*T* - константа, задаваемая перед ГС) шаг станет равным *0*. ГС останавливается.

**В общем, у вас есть *(T+1)* шагов, чтобы найти хороший минимум.**

# Модификации с шагом *h*

**Первая формула:** $h_{n+1} := h_0 \cdot (1 - \frac{n}{T})$

**Например:** если $h_0=0.1$, $T=5$, то получим такую последовательность шагов

| | | | | | | |
|---|---|---|---|---|---|---|
| *h:* | *0.1* | *0.08* | *0.06* | *0.04* | *0.02* | *0* |

То есть на 1й итерации ГС будем использовать шаг 0.1, на 2й - 0.08 и т.д.

# Пример

**Дано:** $f(x_1,x_2)=x^2_1+x^2_2$ $a_0=(1,2)$ $h_0=0.1$ T=10. Градиент равен $(2x_1,2x_2)$

**Сначала считаем последовательность шагов:**

0.1 0.09 0.08 0.07 0.06 0.05 0.04 0.03 0.02 0.01 0

**Сделаем два шага ГС.**

$a_1=(1,2)-0.1*(2,4)=(0.8,1.6)$

$a_2=(0.8,1.6)-0.09*(1.6,3.2)=(0.656,1.312)$

и т.д.

# 1. Модификации с шагом *h*

**Вторая формула:** $h_{n+1} := h_0 \cdot e^{-\frac{n}{T}}$

**Смысл:** Перед началом ГС у нас шаг $h_0$. После каждой итерации шаг **плавно** уменьшается**.** Константа *T* (задаваемая перед ГС) определяет скорость уменьшения длины шага *h*. Чем больше *T*, тем медленнее убывает *h*.

**Например:** если $h_0$*=0.1, T=1,* то получим такую последовательность шагов

*h: 0.1 0.037 0.014 0.005 0.002 0.0007*

А дальше всё аналогично предыдущему алгоритму.

# 2. Метод Momentum

**Идея**: до сего момента шарик (который катится по поверхности) был без каких-либо физических характеристик. Так давайте припишем ему некоторую массу и после каждого шага будем вычислять физическую скорость шарика.

**Зачем**: шарик с большой массой и скоростью будет легко миновать неглубокие минимумы и по инерции пролетать плато.

Осталось лишь выразить всю эту физику с помощью формулы.

# 2. Метод Momentum

**Пусть $\gamma \in [0, 1]$, тогда вычисление новой позиции шарика при ГС происходит по формуле:**

$$\bar{a}_{n+1} := \gamma \bar{a}_n - h \nabla f(\bar{a}_n)$$

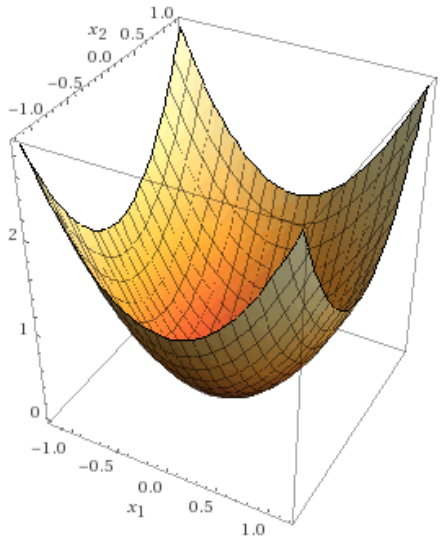
**Смысл** $\gamma$: это как бы масса (инерция) шарика. Этот параметр определяет, насколько сильно новое направление движения зависит от пройденного пути.

Если $\gamma = 1$, то мы имеем **классический** ГС.

# 2. Метод Momentum

**Сравним работу классического ГС и метода Momentum.** Ранее мы показали, что классический ГС для $f(x_1, x_2) = x_1^2 + x_2^2$, *h=0.1* приводит к такому «путешествию» шарика (градиент функции равен $\nabla f = (2x_1, 2x_2)$):

**(1 2) (0.8 1.6) (0.64 1.28)**




**Пусть γ=2 (шарик потяжелел). Тогда:**

$a_1$=2*(1, 2)-0.1(2, 4)=(0.9, 1.8)

$a_2$=2*(0.9, 1.8)-0.1(1.8, 3.6)=(0.81, 1.62)

$a_2$=2*(0.81, 1.62)-0.1(1.62, 3.24)=(0.729, 1.458)

Действительно, шарик медленнее разгоняется.

# 3. Adagrad

Список литературы:

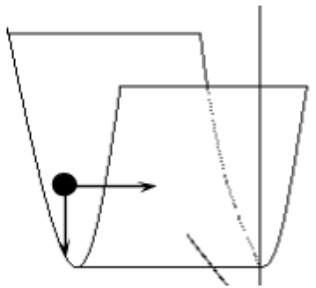
# 3. Adagrad (адаптивный градиент)

**Проблема** («градиентный коммунизм»):, по всем координатам градиент умножается на один и тот же коэффициент *h*.

$$(a_1, a_2) := (a_1, a_2) - h(\frac{\partial f}{\partial x_1}(a_1, a_2), \frac{\partial f}{\partial x_2}(a_1, a_2))$$

Это не всегда оправдано, поскольку по одной координате можно сдвигаться более длинными шагами, а по другим координатам – нужны более мелкие шаги.

**Идея**: а давайте градиент умножать на разные числа, в зависимости от направления.



# 3. Adagrad

Мы накапливаем информацию о градиентах со всех предыдущих позициях шарика $$\bar{G}_{n+1} = \bar{G}_n + (\nabla f(\bar{a}_n))^2$$

**(здесь покоординатное возведение в квадрат).**

Короче говоря, $\bar{G}_{n+1} := (\nabla f(\bar{a}_0))^2 + (\nabla f(\bar{a}_1))^2 + \ldots + (\nabla f(\bar{a}_n))^2$

– это сумма квадратов градиентов с прошлых итераций.

**Новая позиция шарика:** $$\bar{a}_{n+1} = \bar{a}_n - \frac{h}{\sqrt{\bar{G}_{n+1} + \epsilon}} \nabla f(\bar{a}_n)$$

**(здесь градиент покоординатно делится на вектор $\sqrt{\bar{G}_{n+1} + \epsilon}$ )**

Здесь $\varepsilon > 0$ – маленькое число, нужное для избежания деления на 0.

# 3. Adagrad

Сравним работу классического ГС и Adagrad. Ранее мы показали, что классический ГС для $f(x_1, x_2) = x_1^2 + x_2^2$ , *h=0.1* приводит к такому «путешествию» шарика (градиент функции равен $\nabla f = (2x_1, 2x_2)$ ):

(1 2) (0.8 1.6) (0.64 1.28)

Пусть $\varepsilon = 0$ (мы пофигистически полагаем, что деления на 0 в нашей формуле не произойдет).
Шаг прежний *h=0.1* Найдем траекторию ГС.

# 3. Adagrad

**Шаг 1:** берем точку $a_0=(1,2)$:

$$\nabla f(\bar{a}_0) = (2, 4) \quad \bar{G}_1 = (4, 16)$$

$$\bar{a}_1 := (1, 2) - \frac{0.1}{\sqrt{(4, 16)}} \cdot (2, 4) = (0.9, 1.9)$$

**Шаг 2:** берем точку $a_1=(0.9,1.9)$:

$$\nabla f(\bar{a}_1) = (1.8, 3.8),$$
$$\bar{G}_1 = (4, 16) + (3.24, 14.44) = (7.24, 30.44),$$
$$\bar{a}_2 := (0.9, 1.9) - \frac{0.1}{\sqrt{(7.24, 30.44)}} \cdot (1.8, 3.8) =$$
$$= (0.9, 1.9) - \frac{0.1 \cdot (1.8, 3.8)}{(2.69, 5.52)} =$$
$$(0.9, 1.9) - 0.1(0.67, 0.69) = (0.833, 1.831)$$

# 4. Adam

# 4. Adam

**Недостаток Adagrad: градиенты за все прошлые шаги вносят одинаковый вклад в сумму $\bar{G}_n$ .**

Но это не всегда логично. Градиенты с далёких шагов могут быть нерелевантны в текущей точке.

Вы же не используете свой давний опыт альпинизма при перемещении по городской улице!

**Второй недостаток: выражение $\bar{G}_n$ неограниченно растёт, слишком быстро уменьшая шаг.**

# 4. Adam

**Идея**: а давайте запоминать **взвешенные суммы** градиентов (и их квадратов) с прошлых шагов.

**Будем разбирать главную формулу метода Adam:**

$$\bar{a}_{n+1} = \bar{a}_n - \frac{h}{\sqrt{\bar{R}_{n+1} + \epsilon}} \bar{M}_{n+1}$$

Как будет видно дальше, вектора $M_{n+1}$ $(R_{n+1})$ хранят информацию о предыдущих градиентах (квадратов градиентов).

# 4. Adam

$$\bar{M}_{n+1} = \frac{\bar{m}_{n+1}}{1 - \beta_1^{n+1}}$$

$$\bar{R}_{n+1} = \frac{\bar{r}_{n+1}}{1 - \beta_2^{n+1}}$$

где $\beta_1, \beta_2$ – числа-константы, их смысл см. далее.

# 4. Adam

$$\bar{m}_{n+1} = \beta_1 \bar{m}_n + (1-\beta_1)\nabla f(\bar{a}_n)$$

$$\bar{r}_{n+1} = \beta_2 \bar{r}_n + (1-\beta_2)(\nabla f(\bar{a}_n))^2$$

$$\bar{m}_0 = \bar{r}_0 = \bar{0}$$

Т.е. вектора $m_{n+1}$,$r_{n+1}$ накапливают информацию о взвешенном среднем градиента (квадрата градиента) с предыдущих позиций ГС.

Числа $\beta_1, \beta_2$ показывают, **насколько сильно забываются давние значения** градиента и квадрата градиента. Чем меньше $\beta_1, \beta_2$, тем быстрее старые значения забываются. Если $\beta_1, \beta_2 = 0$, то мы вообще не используем никакую историю.

Обычно берут: $\boldsymbol{\beta_1}=0.9$, $\boldsymbol{\beta_2}=0.999$.

# 4. Adam (пример спуска)

$f(x_1, x_2) = x_1^2 + x_2^2$ $\nabla f = (2x_1, 2x_2)$ $\varepsilon = 0$, *h=1,* $a_0$*=(10,20),*
(мы пофигистически полагаем, что деления на *0* в нашей формуле не произойдет),
$\beta_1, \beta_2$ *=0.5.*

$$\nabla f(a_0) = (20, 40),$$
$$(\nabla f(a_0))^2 = (400, 1600),$$
$$m_1 = 0.5 \cdot (0, 0) + (1 - 0.5) \cdot (20, 40) = (10, 20),$$
$$r_1 = 0.5 \cdot (0, 0) + (1 - 0.5) \cdot (400, 1600) = (200, 800),$$

$$M_1 = (10, 20)/(1 - 0.5^1) = (20, 40),$$
$$R_1 = (200, 800)/(1 - 0.5^1) = (400, 1600),$$

$$a_1 = (10, 20) - \frac{1 \cdot (20, 40)}{\sqrt{(400, 1600)}} = (9, 19)$$

# 4. Adam (вторая итерация)

$f(x_1, x_2) = x_1^2 + x_2^2$ $\nabla f = (2x_1, 2x_2)$ , *h=1, $a_1$=(9,19),* $\varepsilon = 0$, $\beta_1, \beta_2$ *=0.5.*

$$\nabla f(a_1) = (18, 38),$$
$$(\nabla f(a_1))^2 = (324, 1444),$$
$$m_2 = 0.5(10, 20) + (1 - 0.5)(18, 38) = (14, 29),$$
$$r_2 = 0.5(200, 800) + (1 - 0.5)(324, 1444) = (262, 1122),$$
$$M_2 = (14, 29)/(1 - 0.5^2) = (18.7, 38.7),$$
$$R_2 = (262, 1122)/(1 - 0.5^2) = (349.3, 1496),$$
$$a_2 = (9, 19) - \frac{1 \cdot (18.7, 38.7)}{\sqrt{(349.3, 1496)}} = (8, 18)$$

Онлайн-курс

# ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

## 2. Точки минимума функций многих переменных

# Выводы

# Выводы

- Мы поняли, что функции могут зависеть от нескольких переменных.
- Такие функции можно дифференцировать по каждой переменной (частные производные).
- Для функций многих переменных работает алгоритм градиентного спуска.
- Мы рассмотрели разные модификации градиентного спуска.
- Апофеоз сегодняшней лекции – метод Adam, который по умолчанию используется в тренировках нейронных сетей.