Търсене и извличане на информация. Приложение на дълбоко машинно обучение

Стоян Михов

Лекция 9: Намиране на градиент чрез пропагиране назад. Стохастично спускане по градиента.

План на лекцията

1. **Формалности за курса (5 мин)**

2. Намиране на градиент чрез пропагиране назад — Backpropagation (20 мин)

3. Пропагиране назад при логистична регресия (20 мин)

4. Сходимост на спускането по градиента (20 мин)

5. Стохастичен градиент (20 мин)

Формалности

- Засега ще провеждаме занятията онлайн всяка сряда от 8:15 до 12:00 часа.

- Засега ще използваме платформата Google meet: meet.google.com/hue-frfx-axb

- Днес ще използваме едновременно слайдове и бяла дъска. Моля следете съответния екран.

- Благодаря за предадените домашни. Ще се постараем да ги оценим до следващото занятие.

- Второто домашно задание ще бъде публикувано в Moodle около средата на декември.

- Деветата лекция се базира на глави 4 и 5 от втория учебник.

Защо да изучаваме автоматично диференциране, спускане по градиент, стохастичен градиент и т.н.

- Нали в модерните системи за дълбоко обучение тези функции вече са имплементирани за нас?
- Също, защо трябва да изучаваме компилатори, след като те вече са имплементирани за нас?

1. Да знаете какво става под повърхността винаги е полезно.
2. Автоматичното диференциране не винаги работи перфектно — разбирането на принципите е критично при дебъгване и подобряване на моделите.
3. При по-специални модели може да се наложи добавянето на нови модули. За разширяването на системите е съществено познаването на теорията.
4. Може да ви се наложи да участвате в разработването на нови системи за дълбоко обучение.

План на лекцията

1. Формалности за курса (5 мин)

2. **Намиране на градиент чрез пропагиране назад — Backpropagation (20 мин)**

3. Пропагиране назад при логистична регресия (20 мин)

4. Сходимост на спускането по градиента (20 мин)

5. Стохастичен градиент (20 мин)

Числено намиране на градиент

- Нека $f : \mathbb{R}^n \to \mathbb{R}$. Тогава: $\dfrac{\partial f(x_1, \ldots, x_n)}{\partial x_i} = \lim\limits_{h \to 0} \dfrac{f(x_1, \ldots, x_i + h, \ldots, x_n) - f(x_1, \ldots, x_i, \ldots, x_n)}{h}$ т.е. $\dfrac{\partial f(\mathbf{x})}{\partial x_i} = \lim\limits_{h \to 0} \dfrac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h}$, където $\mathbf{e}_i$ е i-тия базисен вектор.

- Полагайки достатъчно малко h, примерно $h = 10^{-4}$, ние можем да намерим приближение на производната: $\dfrac{\partial f(\mathbf{x})}{\partial x_i} \approx \dfrac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x})}{h}$.

- По добра апроксимация на производната: $\dfrac{\partial f(\mathbf{x})}{\partial x_i} \approx \dfrac{f(\mathbf{x} + h\mathbf{e}_i) - f(\mathbf{x} - h\mathbf{e}_i)}{2h}$.

- Градиента намираме, като апроксимираме производната в дадената точка по всяка от n-те координати.

- **Задача**: Докажете, че втората формула ни дава по-добро приближение на производната, като оцените порядъка на грешката.

- **Сложност**: Ако сложността на израза за f е от порядък $O(k)$ то сложността за численото намиране на градиента на f в точката $\mathbf{x}$ е от порядък $O(nk)$.

Аналитично намиране на градиент

- Нека $f : \mathbb{R}^n \to \mathbb{R}$. Нека сме намерили аналитични изрази за производните $\dfrac{\partial f(\mathbf{x})}{\partial x_i}$ за $i = 1,2,\ldots,n$.
- Производната по дадено направление намираме, като заместим в съответния израз за производната с дадената точка.
- Градиента намираме, като заместим в израза за производната по всяка от n-те координати с дадената точка.
- **Сложност**: Ако сложността на изразите за $\dfrac{\partial f(\mathbf{x})}{\partial x_i}$ е от порядък $O(k')$, то сложността за аналитичното намиране на градиента на f в точката $\mathbf{x}$ е от порядък $O(nk')$.
- **Задача**: Докажете, че съществуват изрази със сложност $O(k)$, за които сложността на израза за производната е от порядък $O(2^k)$.

Намиране на градиент чрез пропагиране назад — **Backpropagation**

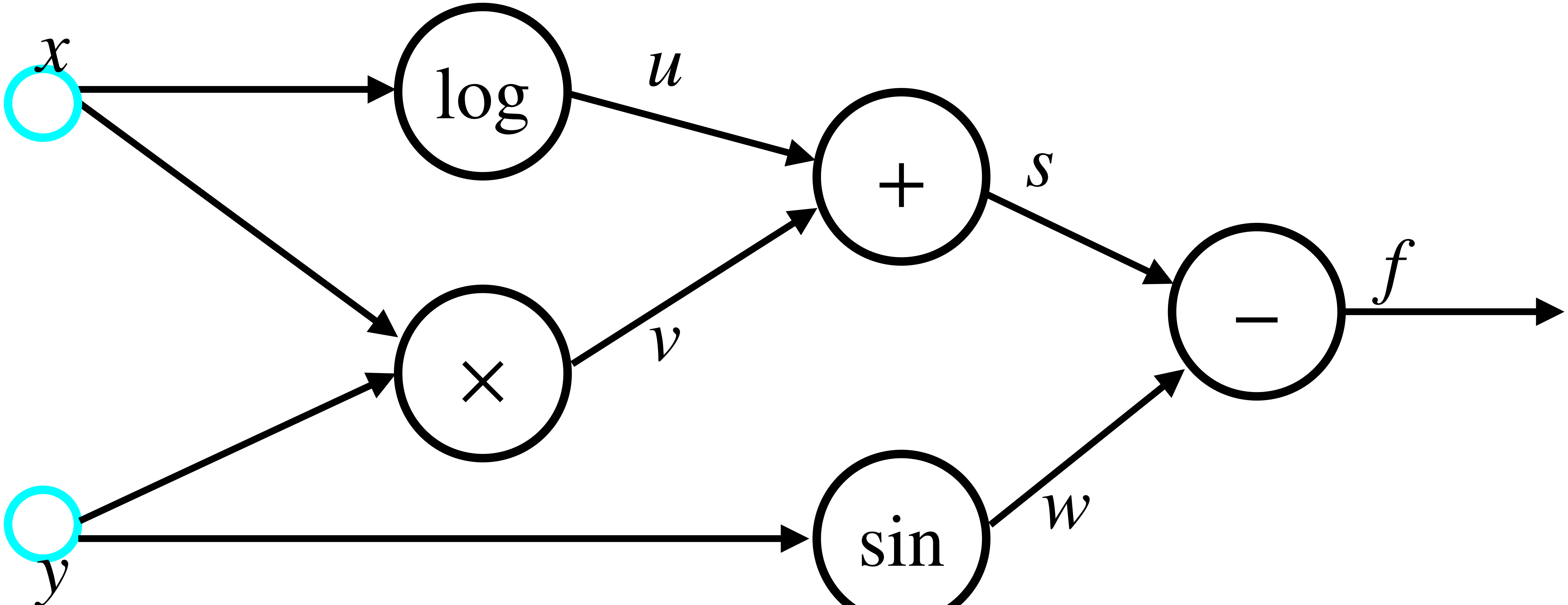
- Намира точните производни и градиент в дадена точка — не е числено приближение.

- Използва се аналитичен израз за производните само за локалните функции — избягва се намирането на изразите за производните на целевата функция.

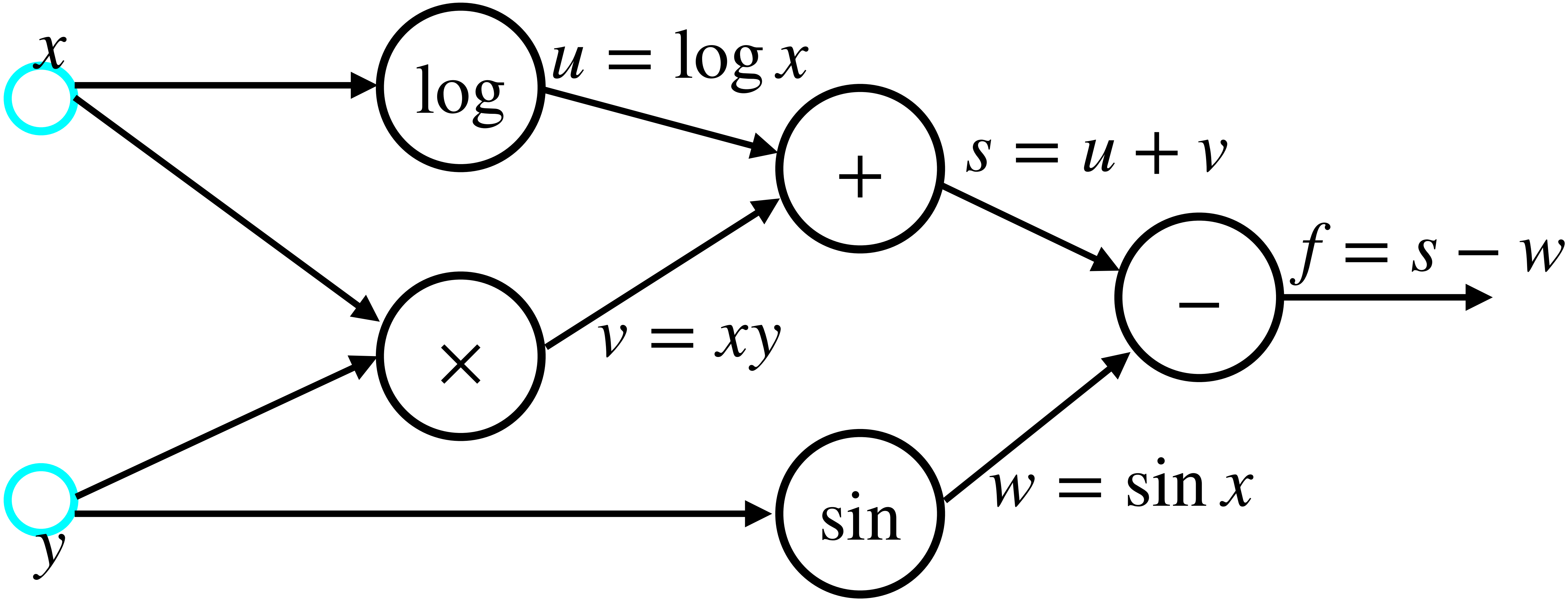
- Преизползват се междинните резултати от изчисленията, с което се постига оптимална изчислителна сложност.

- Сложността за намиране на градиента е $O(k)$, за израз със сложност k.

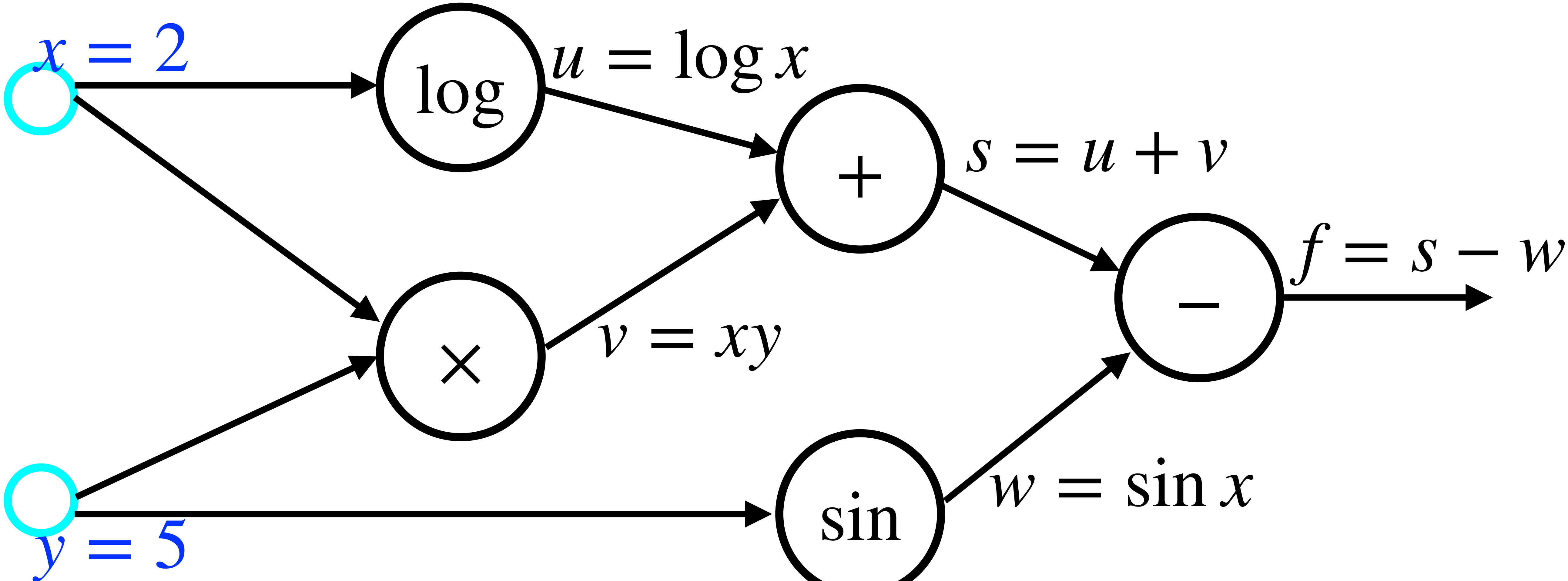
Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$



Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$

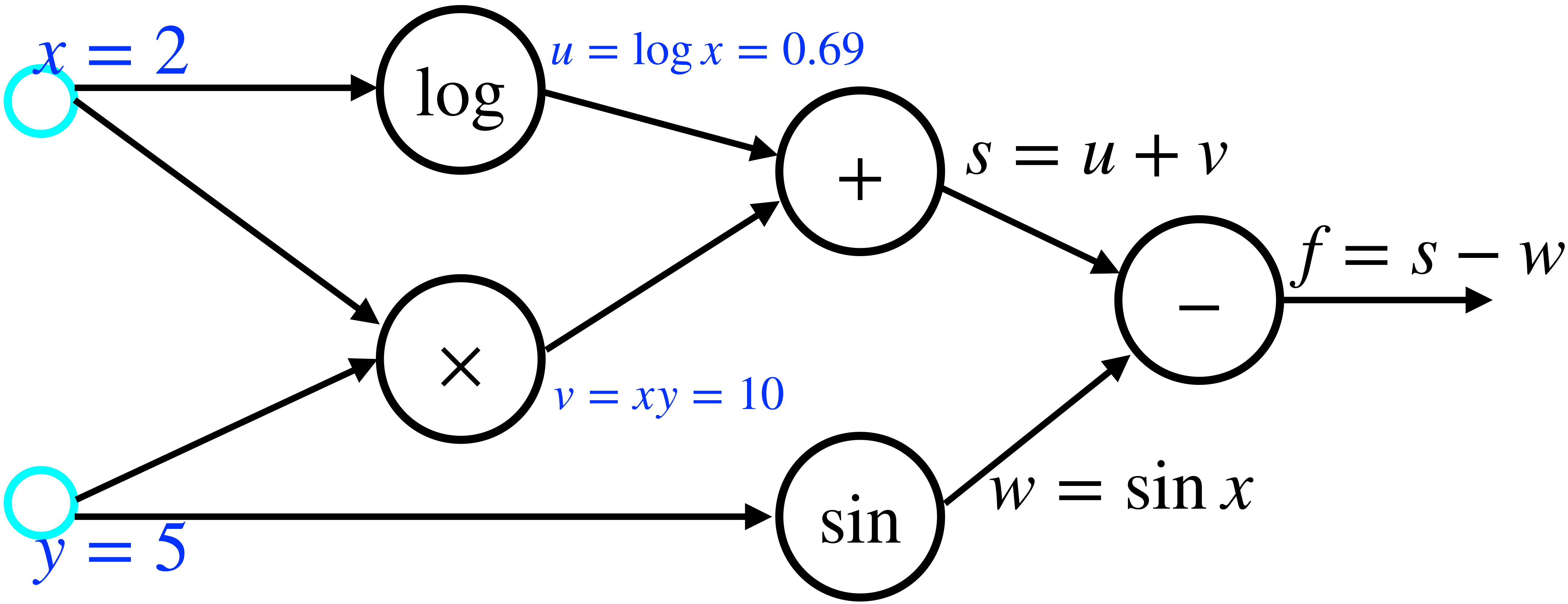


Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$



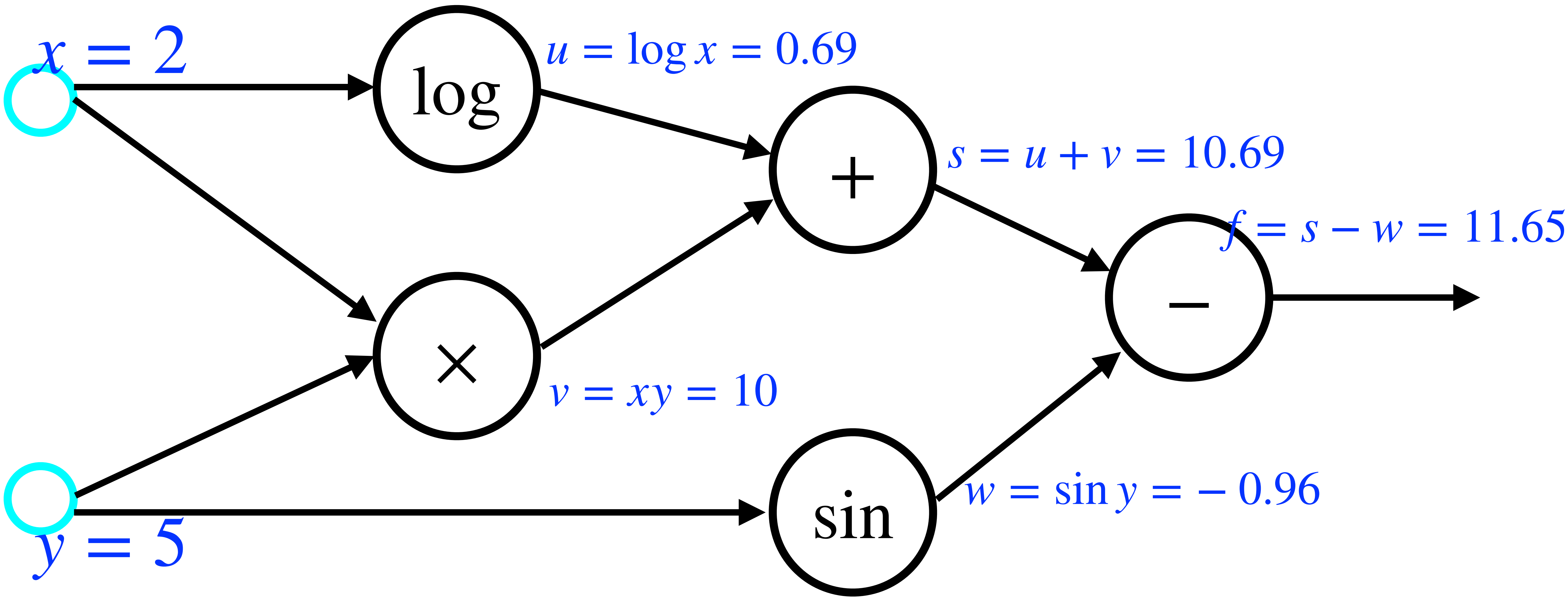
Пропагиране напред — Forward propagation

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$



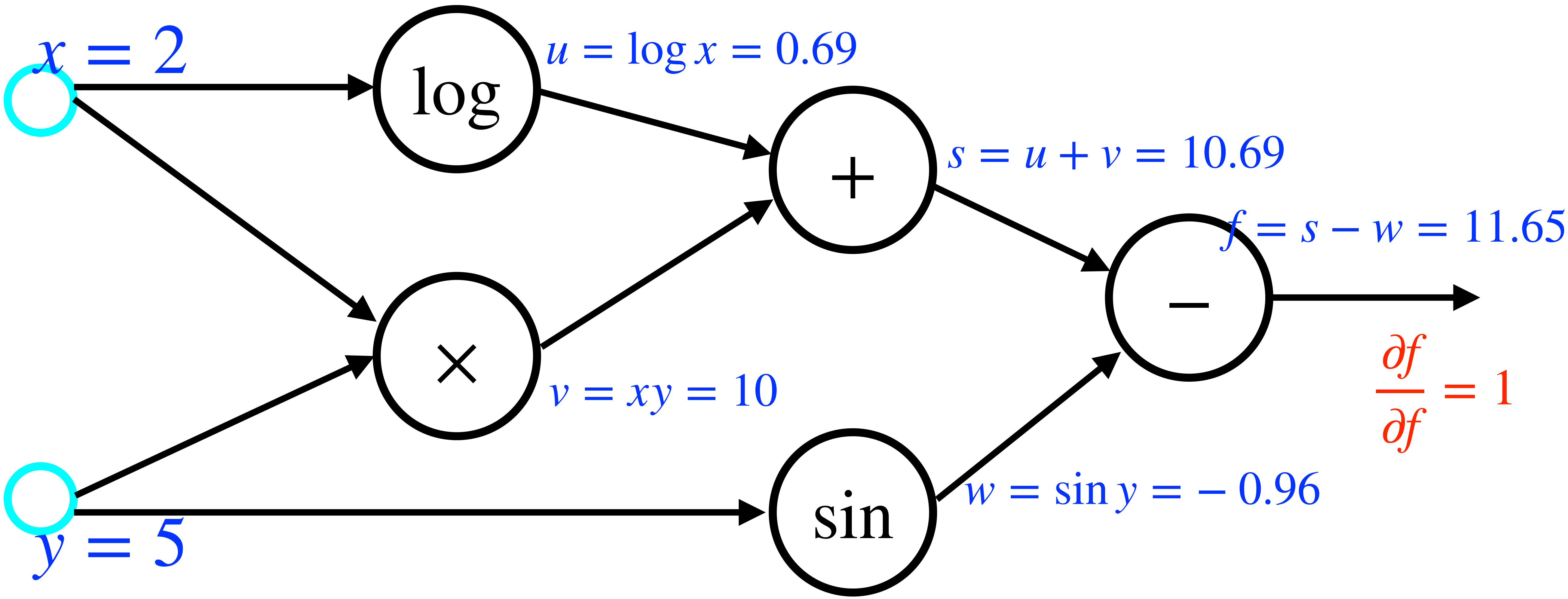
Пропагиране напред — Forward propagation

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$



Пропагиране напред — Forward propagation

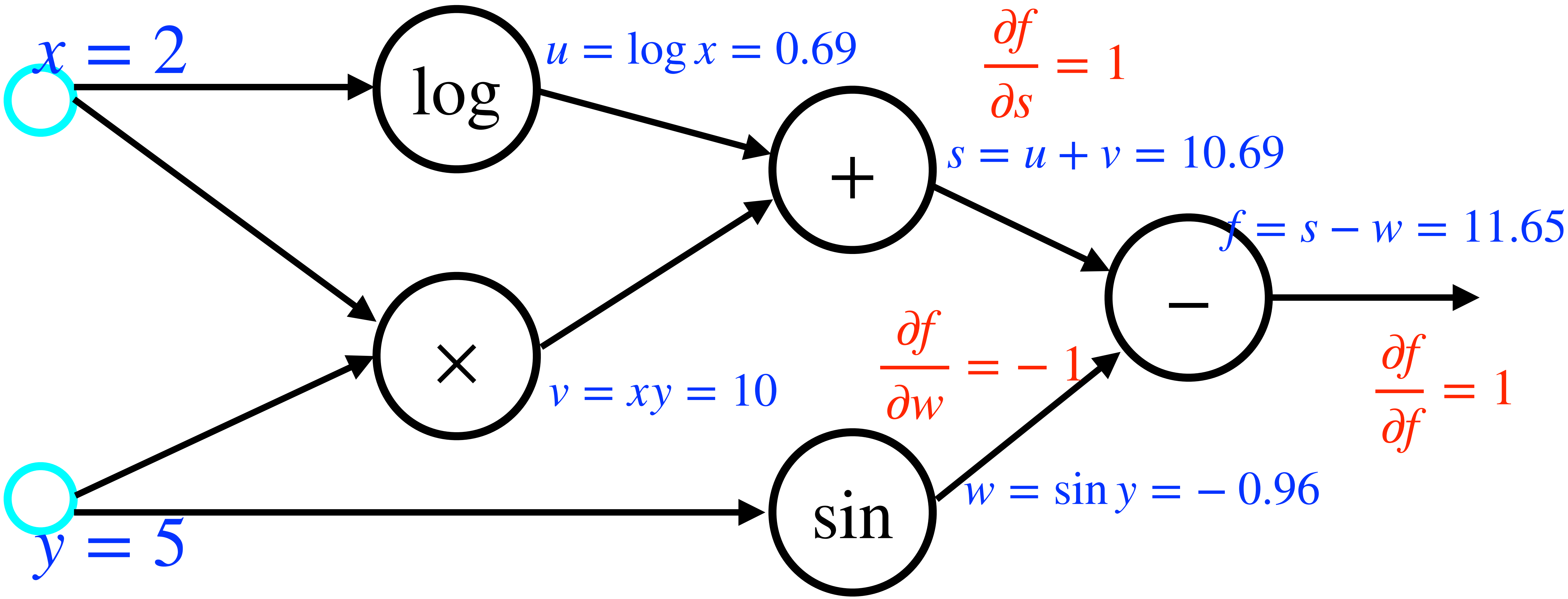
Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$



Пропагиране напред — Forward propagation

Пропагиране назад— Back propagation

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$

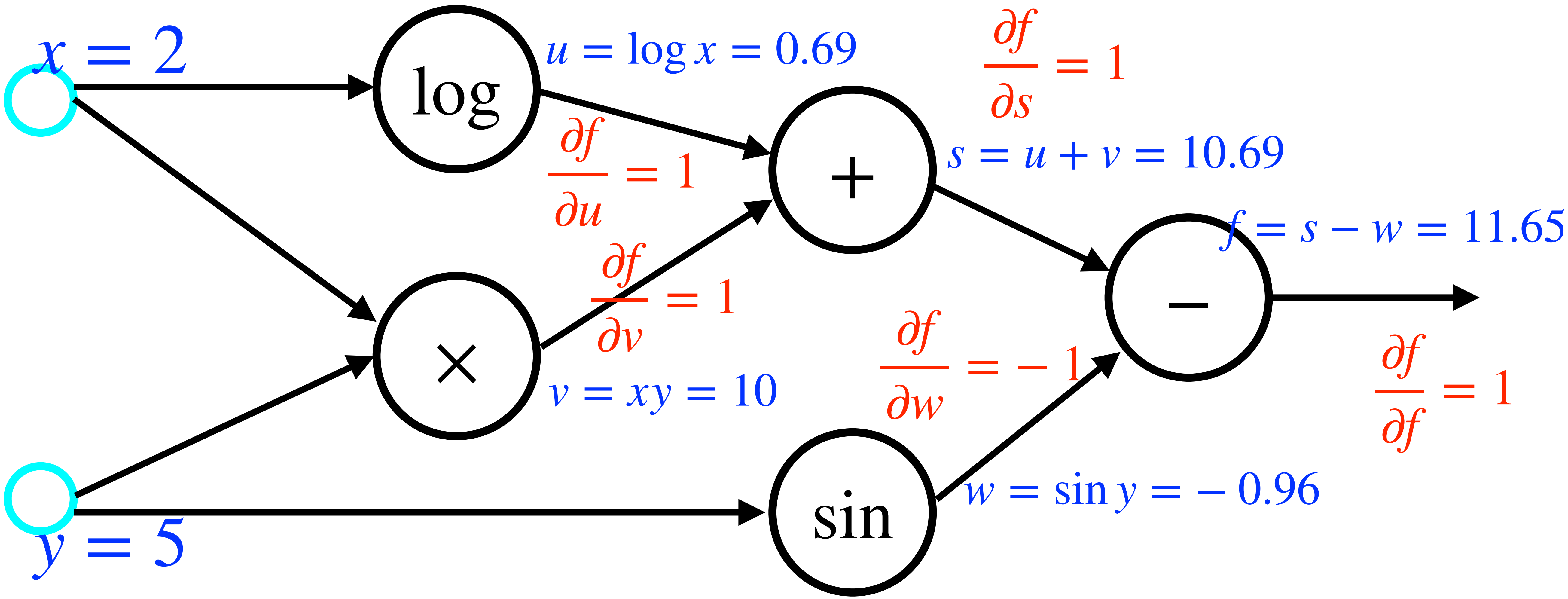


Пропагиране напред — Forward propagation

Пропагиране назад— Back propagation

$$\frac{\partial f}{\partial w} = \frac{\partial f}{\partial f}\frac{\partial f}{\partial w} = \frac{\partial f}{\partial f}(-1) = -1, \quad \frac{\partial f}{\partial s} = \frac{\partial f}{\partial f}\frac{\partial f}{\partial s} = \frac{\partial f}{\partial f}1 = 1$$

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$

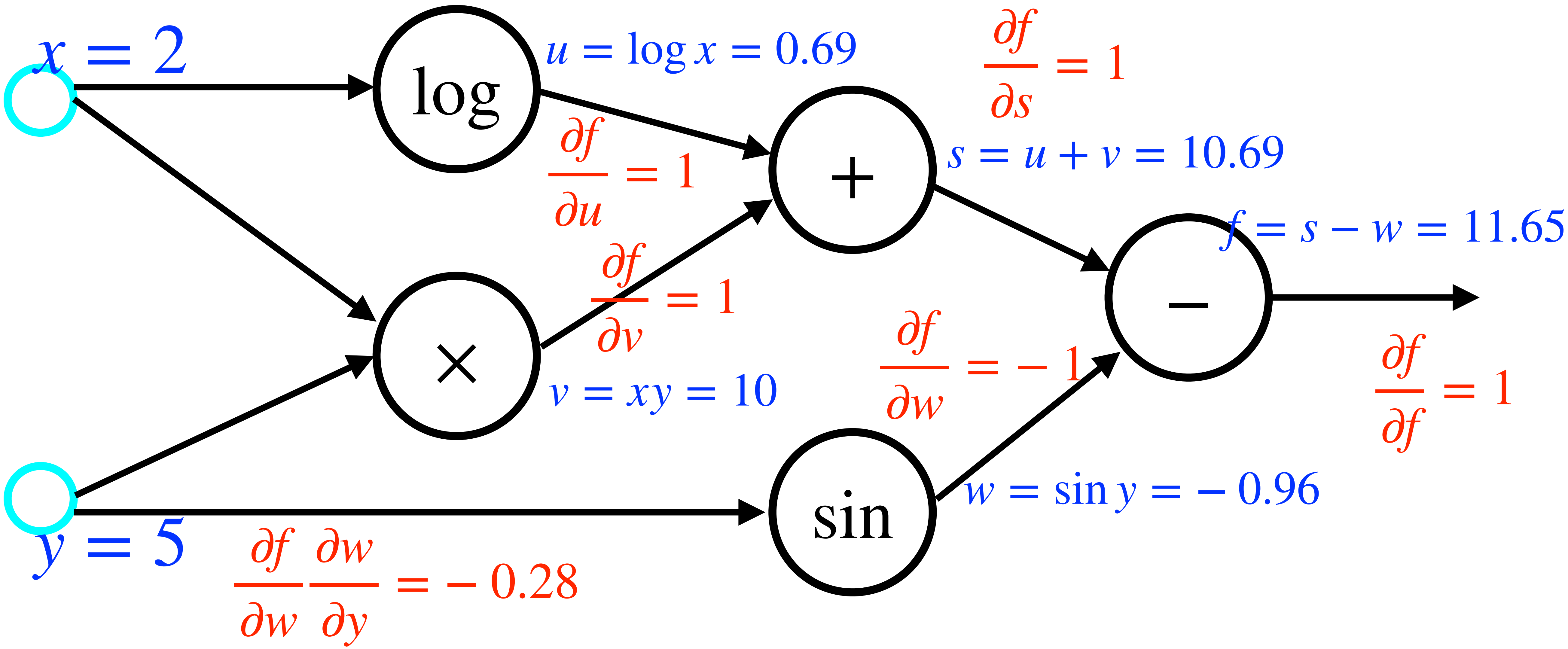


Пропагиране напред — Forward propagation

Пропагиране назад— Back propagation

$$\frac{\partial f}{\partial u} = \frac{\partial f}{\partial s}\frac{\partial s}{\partial u} = \frac{\partial f}{\partial s}1 = 1, \quad \frac{\partial f}{\partial v} = \frac{\partial f}{\partial s}\frac{\partial s}{\partial v} = \frac{\partial f}{\partial s}1 = 1$$

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$

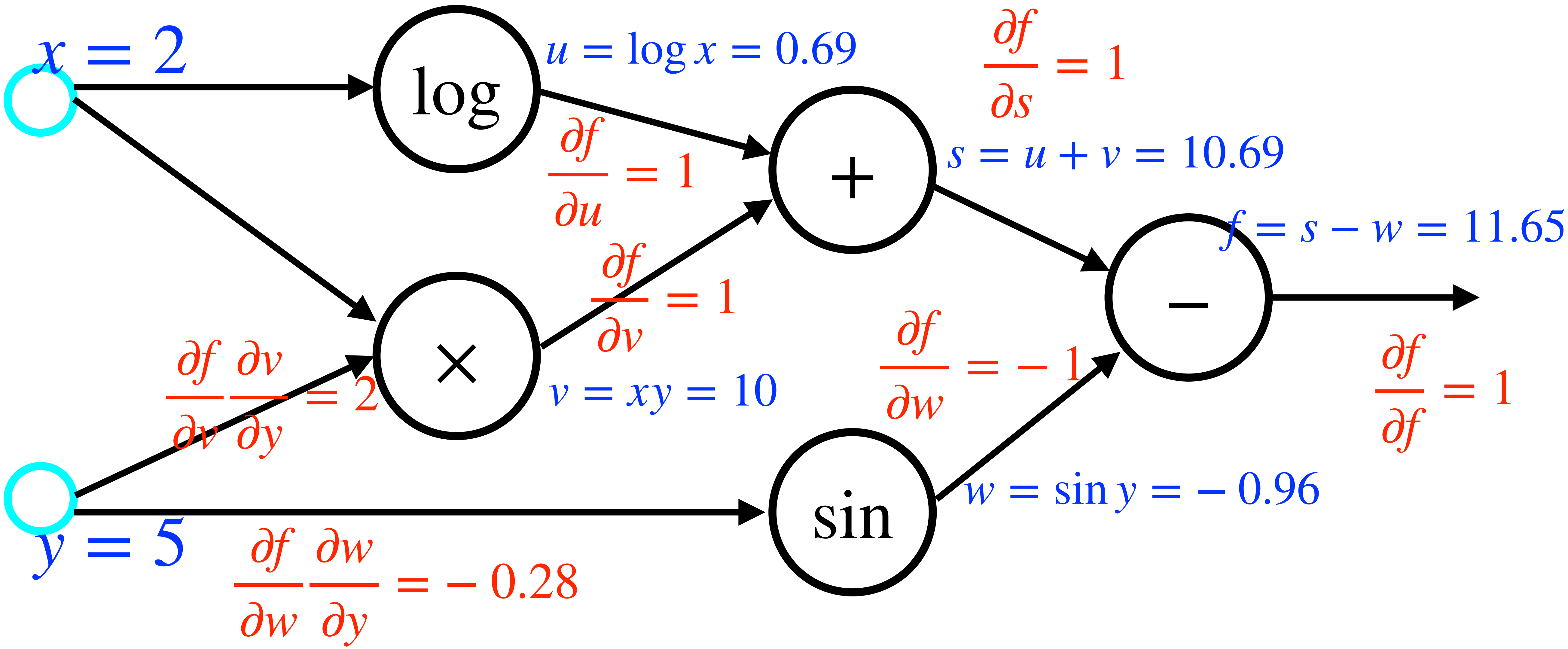


Пропагиране напред — Forward propagation

Пропагиране назад— Back propagation

$$\frac{\partial f}{\partial w}\frac{\partial w}{\partial y} = \frac{\partial f}{\partial w}\cos(y) = (-1)\cos(5) = -0.28$$

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$

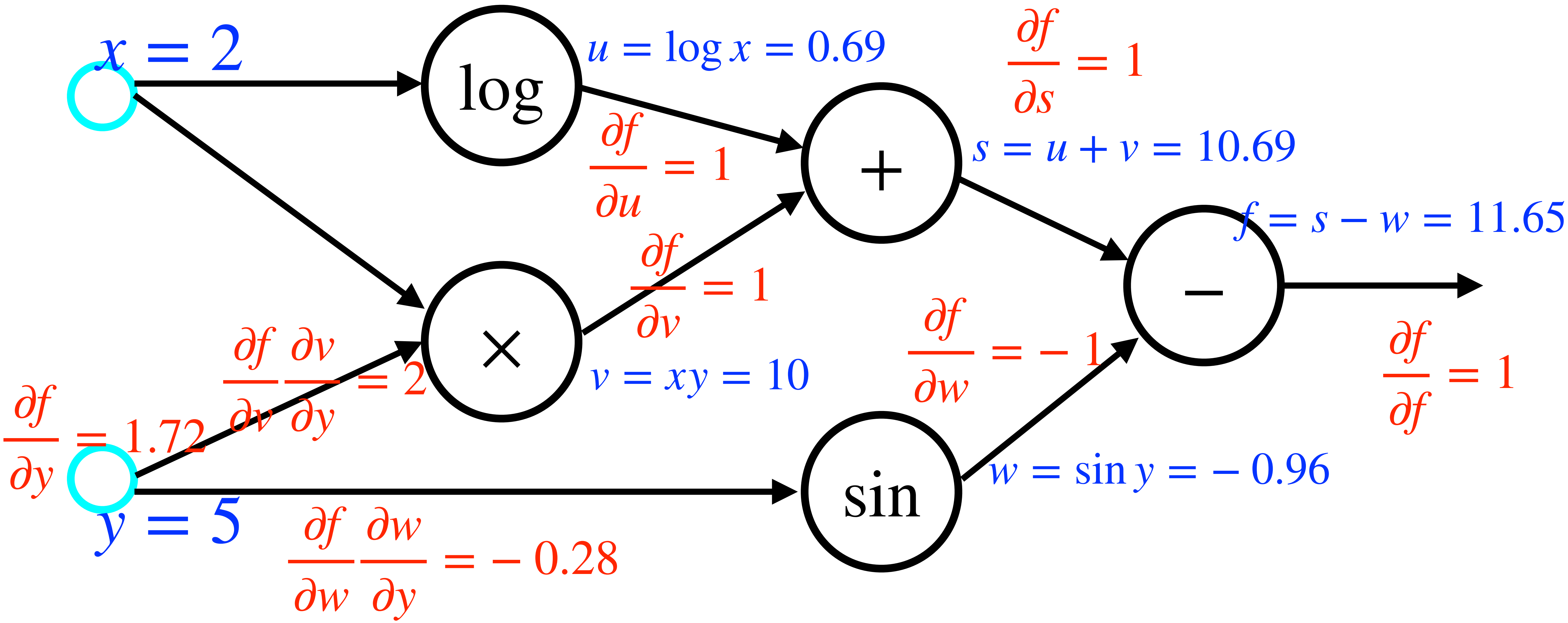


Пропагиране напред — Forward propagation

Пропагиране назад— Back propagation

$$\frac{\partial f}{\partial v}\frac{\partial v}{\partial y} = \frac{\partial f}{\partial v}x = 2$$

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$

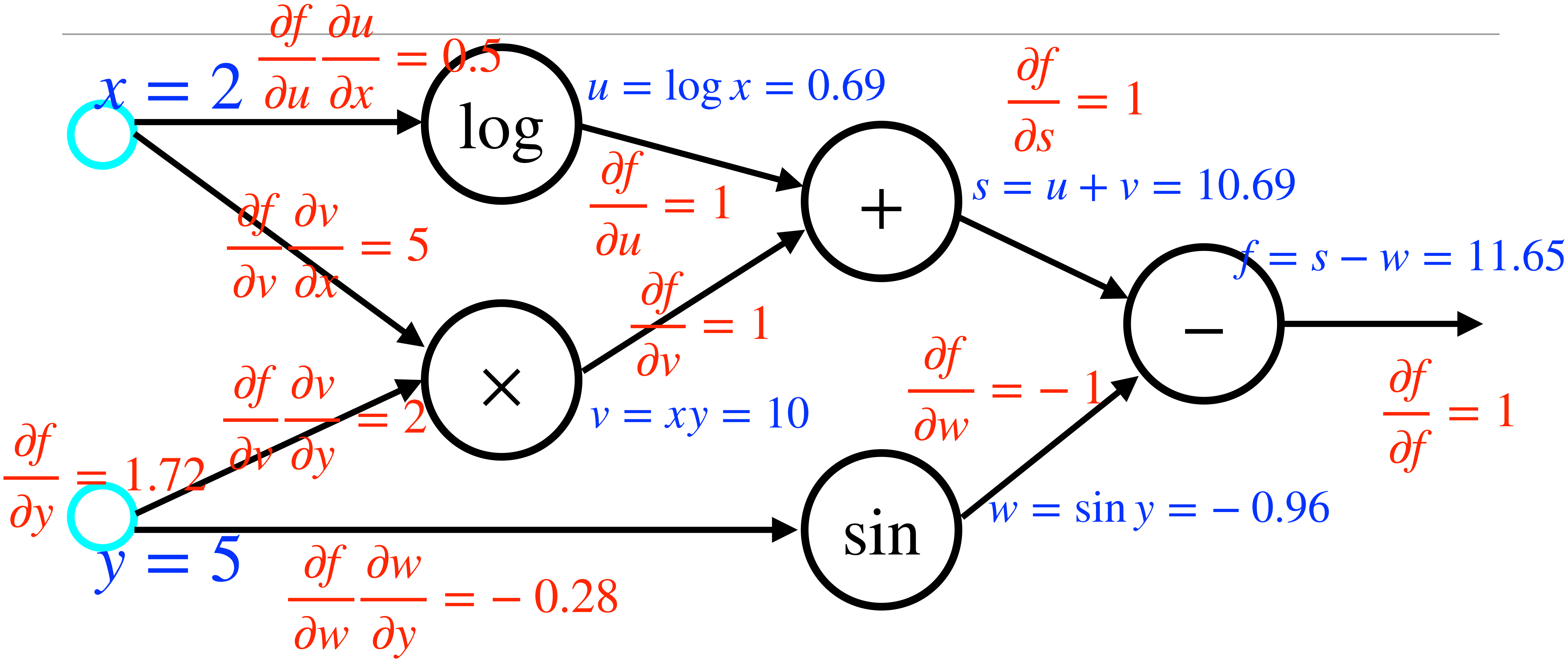


Пропагиране напред — Forward propagation

Пропагиране назад— Back propagation

$$\frac{\partial f}{\partial y} = \frac{\partial f}{\partial v}\frac{\partial v}{\partial y} + \frac{\partial f}{\partial w}\frac{\partial w}{\partial y} = 2 - 0.28 = 1.72$$

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$

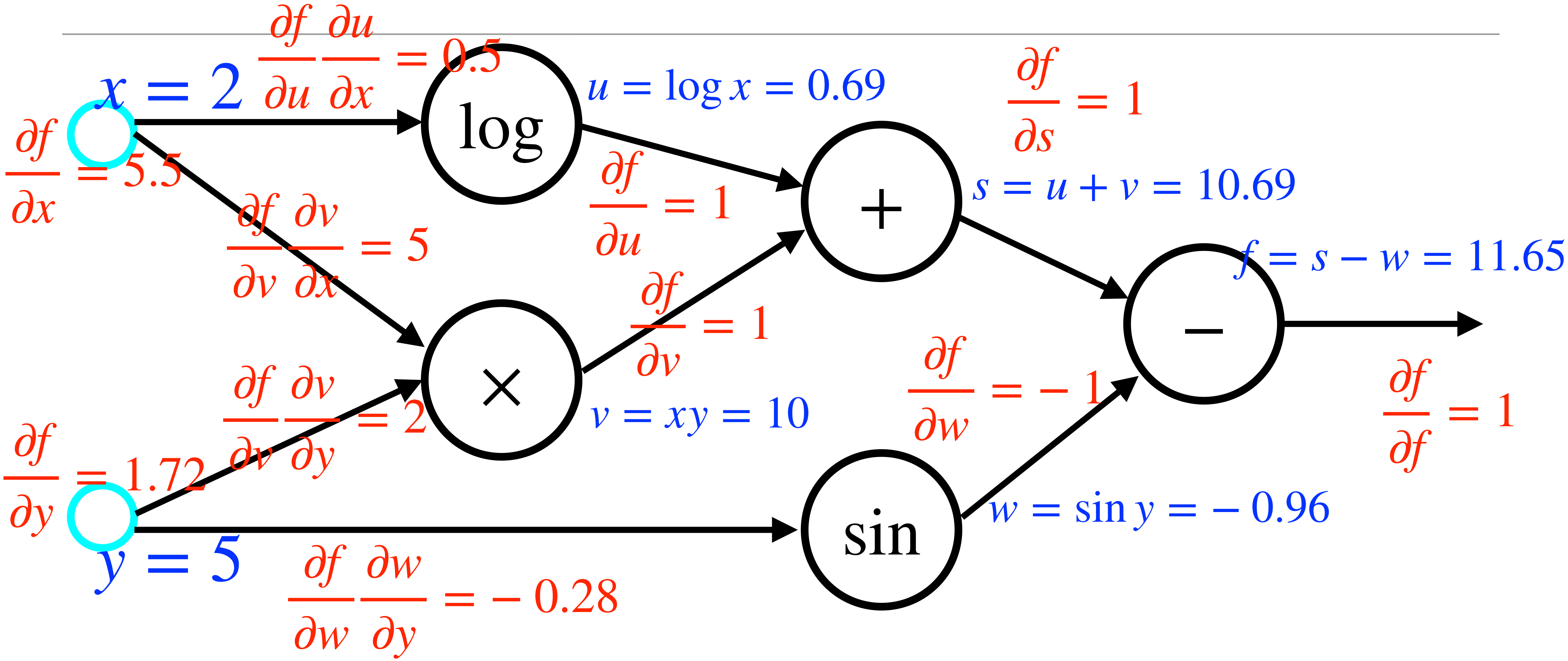


Пропагиране напред — Forward propagation

Пропагиране назад— Back propagation

$$\frac{\partial f}{\partial u}\frac{\partial u}{\partial x} = \frac{\partial f}{\partial u}\frac{1}{x} = 0.5, \frac{\partial f}{\partial v}\frac{\partial v}{\partial x} = \frac{\partial f}{\partial v}y = 5$$

Пример: $f(x, y) = (\log(x) + xy) - \sin(y)$

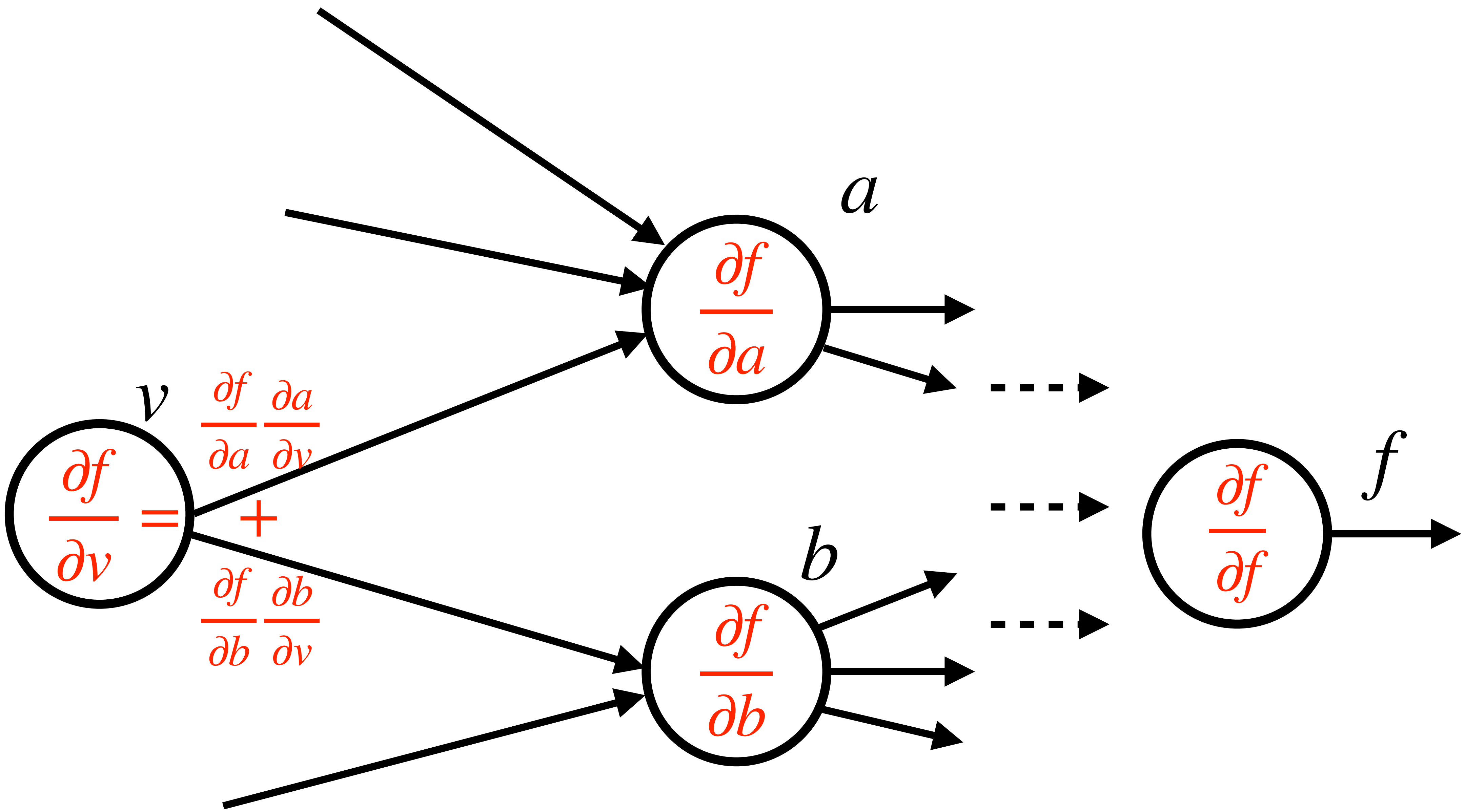


Пропагиране напред — Forward propagation

Пропагиране назад— Back propagation

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial u}\frac{\partial u}{\partial x} + \frac{\partial f}{\partial v}\frac{\partial v}{\partial x} = 5 + 0.5 = 5.5$$

Основен инвариант



Формализация на Backpropagation — изчислителен граф

- Даден е ацикличен граф $G = (V, E), E \subset V \times V$.
- За всеки връх $v \in V$ (непосредствените) предшественици означаваме с $P_G(v) = [p \in V \mid (p, v) \in E]$. Ще предполагаме, че $P_G(v)$ е списък.
- Листата на графа ще означаваме с $L(G) = [v \mid P_G(v) = \emptyset]$.
- Крайни върхове на графа ще означаваме с $T(G) = [v \mid \neg \exists u \in V : (v, u) \in E]$. Ще предполагаме, че в графа съществува единствен краен връх: $|T(G)| = 1$

Формализация на Backpropagation — изчислителен граф

- За всеки връх $v \in V \setminus L(G)$ ще предполагаме, че е дефинирана функция за изчисляване на стойността: $\mathrm{calc}(G, v) : \mathbb{R}^{|P_G(v)|} \to \mathbb{R}$.

- За всеки връх $v \in V \setminus L(G)$ и всеки негов предшественик $u \in P_G(v)$ ще предполагаме, че е дефинирана функция за изчисляване на частната производна на функцията $\mathrm{calc}(G, v)$ спрямо аргумента u, която бележим с $\mathrm{deriv}(G, v, u) : \mathbb{R}^{|P_G(v)|} \to \mathbb{R}$.

Backpropagation алгоритъм

```
Forward(G):
for v in topologicalSort(G) do
    if v in Leafs(G) then read(F(v))
    else
        [p1,p2,...,pk] <- Predecessors(G)(v)
        F(v) <- calc(G,v)(F(p1),…,F(pk))
return F
```

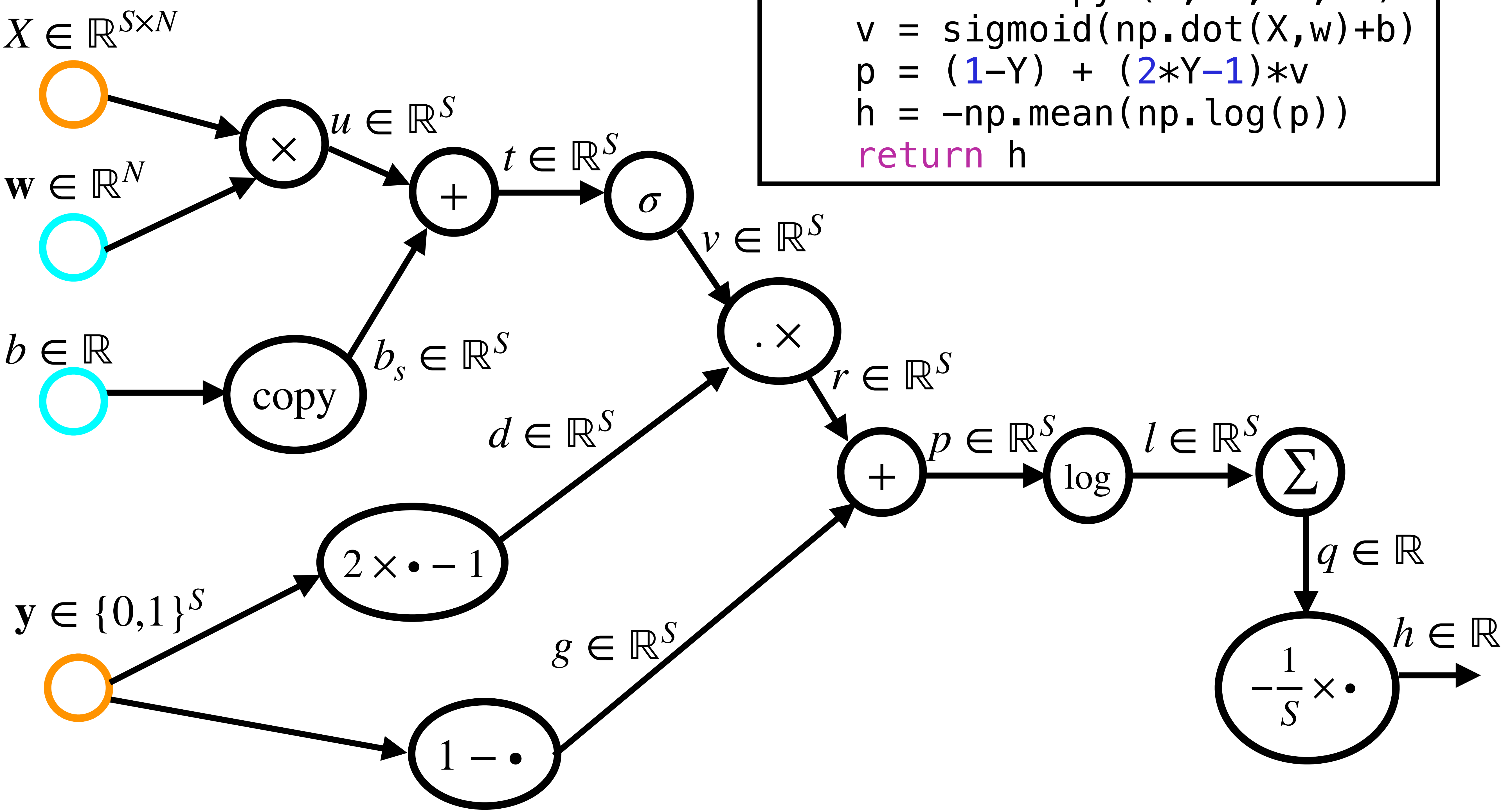
```
Backward(G,F):
for v in topologicalSort(G) do B(v) <- 0
for v in reverseTopologicalSort(G) do
    if v in Top(G) then B(v) <- 1
    [p1,p2,...,pk] <- Predecessors(G)(v)
    for p in Predecessors(G)(v) do
        B(p) += B(v) * deriv(G,v,p)(F(p1),…,F(pk))
return B
```

План на лекцията

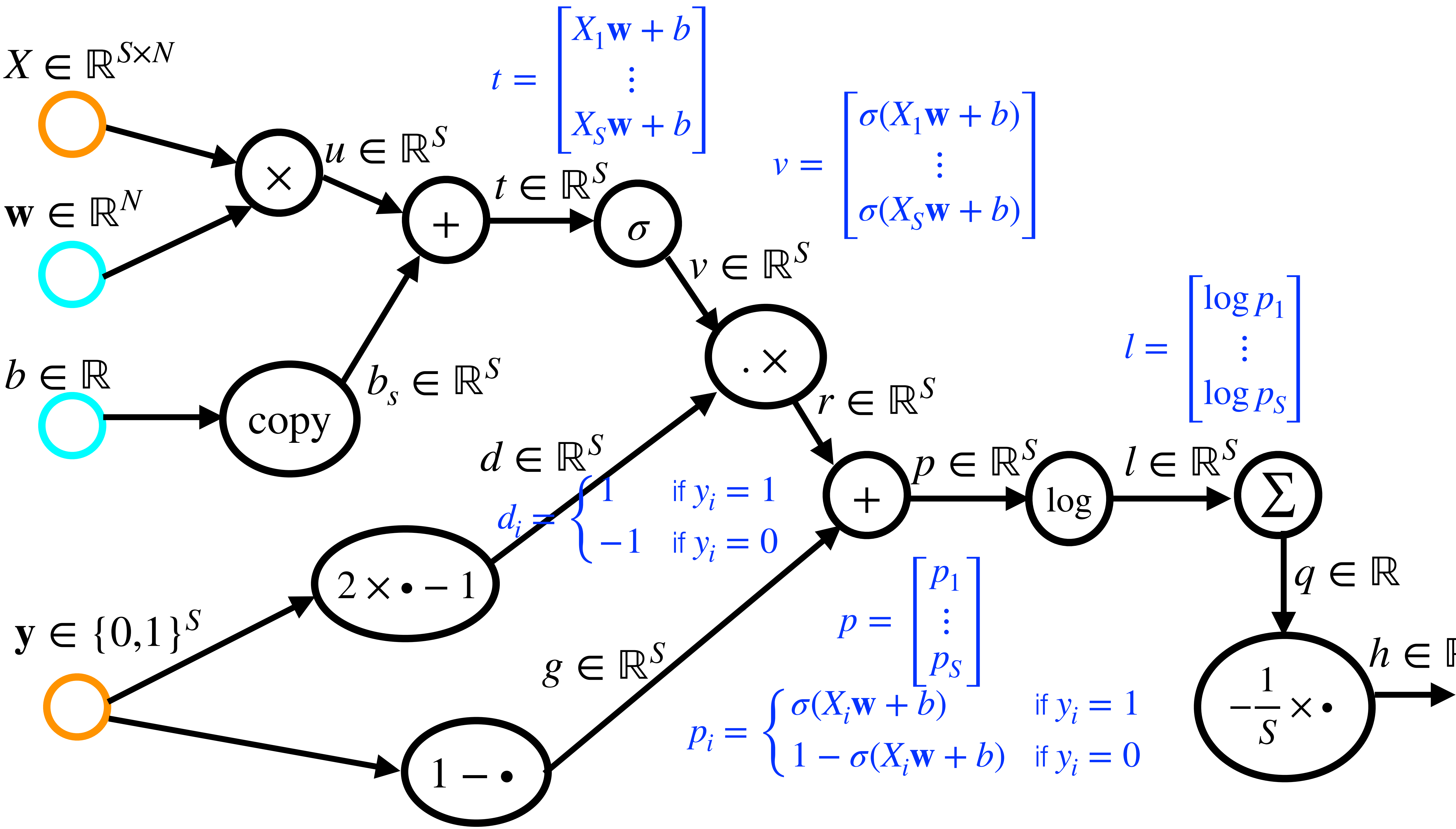
1. Формалности за курса (5 мин)
2. Намиране на градиент чрез пропагиране назад — Backpropagation (20 мин)
3. **Пропагиране назад при логистична регресия (20 мин)**
4. Сходимост на спускането по градиента (20 мин)
5. Стохастичен градиент (20 мин)

Логистична регресия— векторен запис

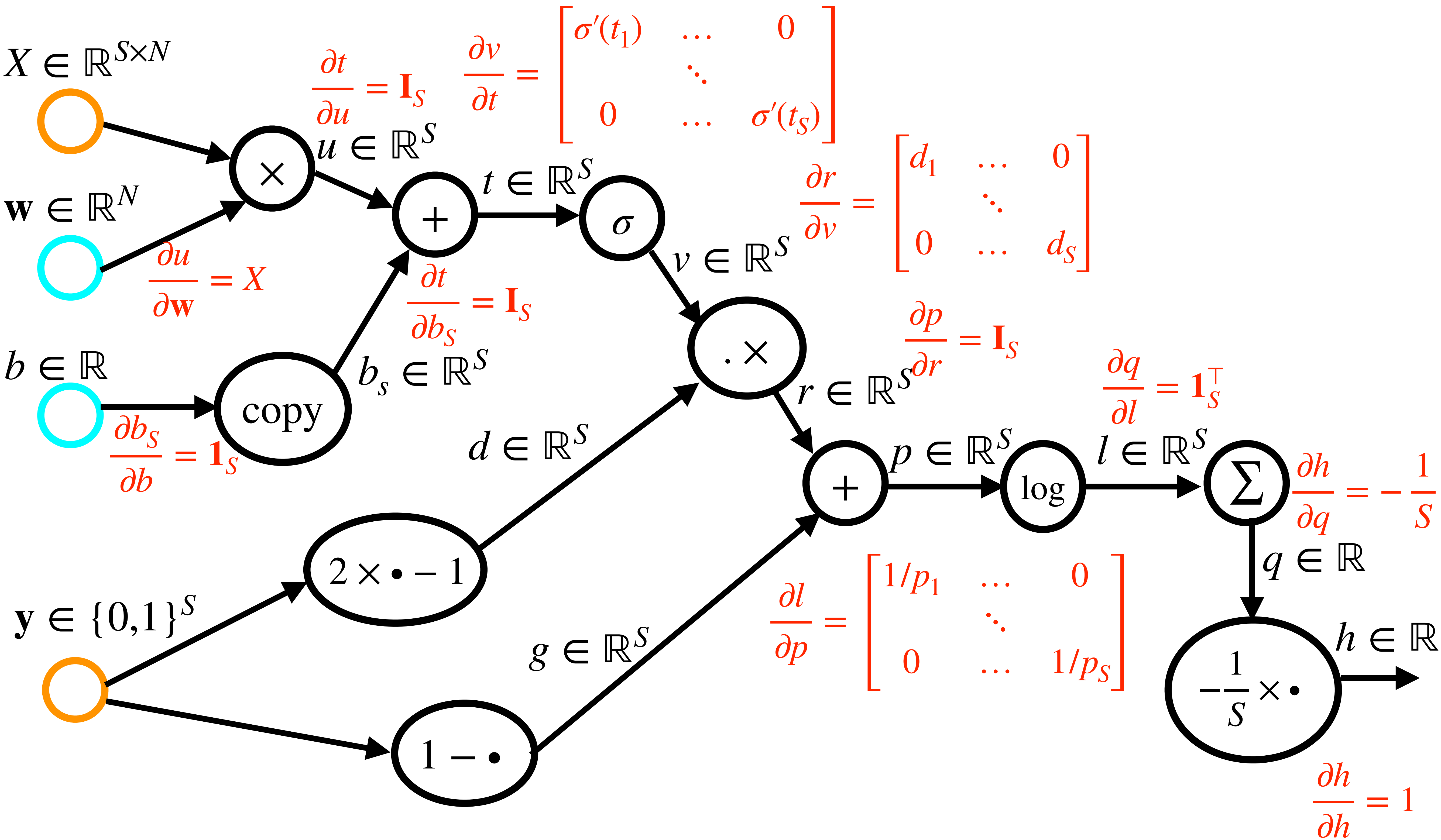
```
def crossEntropyS(X, Y, w, b):
    v = sigmoid(np.dot(X,w)+b)
    p = (1-Y) + (2*Y-1)*v
    h = -np.mean(np.log(p))
    return h
```



Логистична регресия — векторен запис



Логистична регресия — векторен запис



$$
\begin{aligned}
\frac{\partial h}{\partial t} &= \frac{\partial h}{\partial q}\frac{\partial q}{\partial l}\frac{\partial l}{\partial p}\frac{\partial p}{\partial r}\frac{\partial r}{\partial v}\frac{\partial v}{\partial t} \\
&= -\frac{1}{S}\mathbf{1}_S^\top \begin{bmatrix} 1/p_1 \ldots & 0 \\ & \ddots & \\ 0 \ldots & 1/p_S \end{bmatrix} \mathbf{I}_S \begin{bmatrix} d_1 \ldots & 0 \\ & \ddots & \\ 0 \ldots & d_S \end{bmatrix} \begin{bmatrix} \sigma'(t_1) \ldots & 0 \\ & \ddots & \\ 0 \ldots & \sigma'(t_S) \end{bmatrix} = \\
&= \left[-\frac{y_1 - \sigma(X_1\mathbf{w} + b)}{S} \ldots - \frac{y_S - \sigma(X_S\mathbf{w} + b)}{S} \right]
\end{aligned}
$$

$$
\frac{\partial h}{\partial b} = \frac{\partial h}{\partial t}\frac{\partial t}{\partial b_S}\frac{\partial b_S}{\partial b} = \left[-\frac{y_1 - \sigma(X_1\mathbf{w} + b)}{S} \quad \ldots \quad -\frac{y_S - \sigma(X_S\mathbf{w} + b)}{S} \right] \mathbf{I}_S \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = -\frac{1}{S}\sum_{i=1}^{S} y_i - \sigma(X_i\mathbf{w} + b)
$$

$$
\begin{aligned}
\frac{\partial h}{\partial w} &= \frac{\partial h}{\partial t}\frac{\partial t}{\partial u}\frac{\partial u}{\partial w} = \left[-\frac{y_1 - \sigma(X_1\mathbf{w} + b)}{S} \ldots - \frac{y_S - \sigma(X_S\mathbf{w} + b)}{S} \right] \mathbf{I}_S X = \\
&= -\frac{1}{S}\sum_{i=1}^{S} (y_i - \sigma(X_i\mathbf{w} + b))X_i
\end{aligned}
$$

План на лекцията

1. Формалности за курса (5 мин)

2. Намиране на градиент чрез пропагиране назад — Backpropagation (20 мин)

3. Пропагиране назад при логистична регресия (20 мин)

4. **Сходимост на спускането по градиента (20 мин)**

5. Стохастичен градиент (20 мин)

Градиент, Хесиан, развиване в ред на Тейлър

Теорема (Тейлър): Нека $g : \mathbb{R} \to \mathbb{R}$ е два пъти диференцируема с непрекъснати производни в околност на точката $t_0 \in \mathbb{R}$. Нека $t \in \mathbb{R}$ е произволна точка в тази околност. Тогава съществува $\bar{t} \in (t_0, t)$, така че:

$$g(t) = g(t_0) + g'(t_0)(t - t_0) + \frac{1}{2}g''(\bar{t})(t - t_0)^2$$

Ще изведем теоремата на Тейлър за многомерния случай.

Нека $f : \mathbb{R}^n \to \mathbb{R}$. **Градиент** и **Хесиан** на f наричаме съответно:

$$\nabla_{\mathbf{x}} f(\mathbf{x}) = \frac{\partial}{\partial \mathbf{x}} f(\mathbf{x}) = \begin{bmatrix} \frac{\partial}{\partial x_1} f(\mathbf{x}) \\ \frac{\partial}{\partial x_2} f(\mathbf{x}) \\ \vdots \\ \frac{\partial}{\partial x_n} f(\mathbf{x}) \end{bmatrix}, \nabla^2_{\mathbf{x}} f(\mathbf{x}) = \frac{\partial^2}{\partial \mathbf{x}^2} f(\mathbf{x}) = \begin{bmatrix} \frac{\partial^2}{\partial x_1 \partial x_1} f(\mathbf{x}) & \dots & \frac{\partial^2}{\partial x_1 \partial x_n} f(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2}{\partial x_n \partial x_1} f(\mathbf{x}) & \dots & \frac{\partial^2}{\partial x_n \partial x_n} f(\mathbf{x}) \end{bmatrix}$$

Нека $f : \mathbb{R}^n \to \mathbb{R}$. Т.е. за $\mathbf{v} \in \mathbb{R}^n$ имаме, че $f(\mathbf{v}) \in \mathbb{R}$. Разглеждаме функциите $\mathbf{v} : \mathbb{R} \to \mathbb{R}^n$, $\mathbf{v}(t) = \mathbf{x}_0 + t(\mathbf{x} - \mathbf{x}_0)$ и $g : \mathbb{R} \to \mathbb{R}$, $g(t) = f(\mathbf{v}(t)) = f(\mathbf{x}_0 + t(\mathbf{x} - \mathbf{x}_0))$ и прилагаме теоремата на Тейлър за g при $t_0 = 0,\ t = 1$. В такъв случай имаме, че

$$g'(t) = (f(\mathbf{v}(t)))' = \left(\frac{\partial f(\mathbf{v})}{\partial \mathbf{v}}\right)^\top \frac{\partial \mathbf{v}(t)}{\partial t} = (\nabla_\mathbf{v} f(\mathbf{v}))^\top (\mathbf{x} - \mathbf{x}_0),$$

$$g''(t) = \frac{\partial}{\partial t}\left(\left(\frac{\partial f(\mathbf{v})}{\partial \mathbf{v}}\right)^\top (\mathbf{x} - \mathbf{x}_0)\right) = (\mathbf{x} - \mathbf{x}_0)^\top \frac{\partial^2 f(\mathbf{v})}{\partial \mathbf{v}^2}(\mathbf{x} - \mathbf{x}_0) = (\mathbf{x} - \mathbf{x}_0)^\top \nabla^2_\mathbf{v} f(\mathbf{v})(\mathbf{x} - \mathbf{x}_0)$$

Теорема (Тейлър): Нека $f : \mathbb{R}^n \to \mathbb{R}$ е два пъти диференцируема с непрекъснати производни в околност на точката $\mathbf{x}_0 \in \mathbb{R}^n$. Нека $\mathbf{x} \in \mathbb{R}^n$ е произволна точка в тази околност. Тогава съществува $\bar{t} \in (0,1)$, така че ако $\bar{\mathbf{x}} = \mathbf{x}_0 + \bar{t}(\mathbf{x} - \mathbf{x}_0)$, то:

$$f(\mathbf{x}) = f(\mathbf{x}_0) + (\nabla_\mathbf{x} f(\mathbf{x}_0))^\top (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^\top \nabla^2_\mathbf{x} f(\bar{\mathbf{x}})(\mathbf{x} - \mathbf{x}_0)$$

Сходимост на спускането по градиента

- Стремим се да минимизираме кросентропията $H_X[\Pr \| \Pr_\theta] = -\frac{1}{|X|}\sum_{i=1}^{|X|} \log \Pr_\theta[\mathbf{x}^{(i)}]$. Нека положим $f(\theta) = H_X[\Pr \| \Pr_\theta]$.
- Спускаме се по градиента: $\theta_{k+1} = \theta_k - \alpha \nabla_\theta f(\theta_k)$
- От развиването в ред на Тейлър получаваме:
- $f(\theta_{k+1}) = f(\theta_k) - \alpha \nabla f(\theta_k)^\top \nabla f(\theta_k) + \frac{1}{2}(\alpha \nabla f(\theta_k))^\top \nabla^2 f(\bar{\theta})(\alpha \nabla f(\theta_k))$

- Да допуснем, че вторите производни са ограничени: $\|\nabla^2 f(\theta)\| \leq L$. Тогава: $\|\mathbf{u}^\top \nabla^2 f(\theta)\ \mathbf{u}\| \leq L\|\mathbf{u}\|^2$

- Заместваме и получаваме:

$$f(\theta_{k+1}) \leq f(\theta_k) - \alpha\|\nabla f(\theta_k)\|^2 + \frac{\alpha^2 L}{2}\|\nabla f(\theta_k)\|^2$$

- Нека подберем α, така че $\alpha L < 1$. Тогава:

$$f(\theta_k) - f(\theta_{k+1}) \geq \frac{\alpha}{2}\|\nabla f(\theta_k)\|^2$$

- Следователно, на всяка стъпка **стойността на** f **намалява** и разликата на последователните членове на редицата е $\geq \dfrac{\alpha}{2}\|\nabla f(\theta_k)\|^2$.

- Нека сумираме нашето неравенство за $k = 0, 1, \ldots, T-1$:
$$\frac{\alpha}{2}\sum_{k=0}^{T-1}\|\nabla f(\theta_k)\|^2 \le \sum_{k=0}^{T-1} f(\theta_k) - f(\theta_{k+1}) = f(\theta_0) - f(\theta_T)$$

- Нека f^* е глобален минимум на функцията f. Тогава $f(\theta_T) \ge f^*$ и следователно: $\displaystyle\sum_{k=0}^{T-1}\|\nabla f(\theta_k)\|^2 \le \frac{2}{\alpha}(f(\theta_0) - f^*)$.

- Дясната страна е константа, следователно редът при $T \to \infty$ с неотрицателни членове е ограничен, следователно е сходящ и $\|\nabla f(\theta_k)\|^2 \longrightarrow 0$.

- $\|\theta_{k+1} - \theta_k\| = \alpha\|\nabla f(\theta_k)\| \to 0$. Следователно редицата θ_k (както и $f(\theta_k)$) е сходяща.$\blacksquare$

План на лекцията

1. Формалности за курса (5 мин)
2. Намиране на градиент чрез пропагиране назад — Backpropagation (20 мин)
3. Пропагиране назад при логистична регресия (20 мин)
4. Сходимост на спускането по градиента (20 мин)
5. **Стохастичен градиент (20 мин)**

Стандартен стохастичен градиент
Standard Stochastic Gradient

- $f(\theta) = H_X[\Pr \| \Pr_\theta] = -\frac{1}{|X|}\sum_{i=1}^{|X|} \log \Pr_\theta[\mathbf{x}^{(i)}]$. Нека с $H_{X_i}[\Pr \| \Pr_\theta] = -\log \Pr_\theta[\mathbf{x}^{(i)}]$ означим поточковата кросентропия в точката X_i и $f_{X_i}(\theta) = H_{X_i}[\Pr \| \Pr_\theta]$.

- Нека X_{i_k} е случайно наблюдение (семпъл) от X. Тогава:
$\theta_{k+1} = \theta_k - \alpha \nabla_\theta f_{X_{i_k}}(\theta_k)$

- От теоремата на Тейлър получаваме:

$$f(\theta_{k+1}) = f(\theta_k) - \alpha \nabla f(\theta_k)^\top \nabla f_{X_{i_k}}(\theta_k) + \frac{1}{2}(\alpha \nabla f_{X_{i_k}}(\theta_k))^\top \nabla^2 f(\bar{\theta})(\alpha \nabla f_{X_{i_k}}(\theta_k))$$

$$f(\theta_{k+1}) \leq f(\theta_k) - \alpha \nabla f(\theta_k)^\top \nabla f_{X_{i_k}}(\theta_k) + \frac{\alpha^2 L}{2}\|\nabla f_{X_{i_k}}(\theta_k)\|^2$$

Свойства на стандартния стохастичен градиент

- Изчисляването на градиента става $|X|$ пъти по-бързо.

- $$f(\theta_k) - f(\theta_{k+1}) \geq \alpha \nabla f(\theta_k)^\top \nabla f_{X_{i_k}}(\theta_k) - \frac{\alpha^2 L}{2} \|\nabla f_{X_{i_k}}(\theta_k)\|^2$$

- $\alpha \nabla f(\theta_k)^\top \nabla f_{X_{i_k}}(\theta_k)$ може да бъде и отрицателно, следователно нямаме никаква гаранция, че стойността на f намалява.

- Може да подходим **вероятностно**.

Партиден стохастичен градиент
Batched Stochastic Gradient

- Разглеждаме **партида** (batch, minibatch) — извадка от B на брой наблюдения на случайни величини от X: $\mathbf{X}_B = X_1, X_2, \ldots, X_B$. Тогава ако означим с $f_{\mathbf{X}_B}(\theta) = \dfrac{1}{B}\sum_{i=1}^{B} f_{X_i}(\theta_k)$ кросентропията на партидата $\mathbf{X}_B$, то дефинираме презаписването на параметрите като:
- $\theta_{k+1} = \theta_k - \alpha \nabla_\theta f_{X_B}(\theta_k)$.
- Можем да повторим същите разсъждения като при стандартния стохастичен градиент в случая на партида. Разликите са, че времето за намиране на градиента нараства с фактор B, но за сметка на това може да очакваме, че отклонението на партидния градиента от ще намалее пълния градиент ще намалее.

Заключение

- Чрез Backpropagation градиентите на сложни функции се изчисляват автоматично, ефективно и точно, като в същото време се имплементират лесно.
- Спускането по пълния градиент е гарантирано сходящо, но се изчислява бавно.
- При стандартния стохастичен градиент не е гарантирано, че се намалява грешката, но е много по бързо.
- Партидният стохастичен градиент е компромис, при който има много по-висока вероятност за схождане, като е значително по-ефективен от пълния градиент.
- Партидният стохастичен градиент (и неговите вариации) е дефакто стандартният подход в съвременните системи за дълбоко машинно обучение.

Логистична регресия— векторен запис