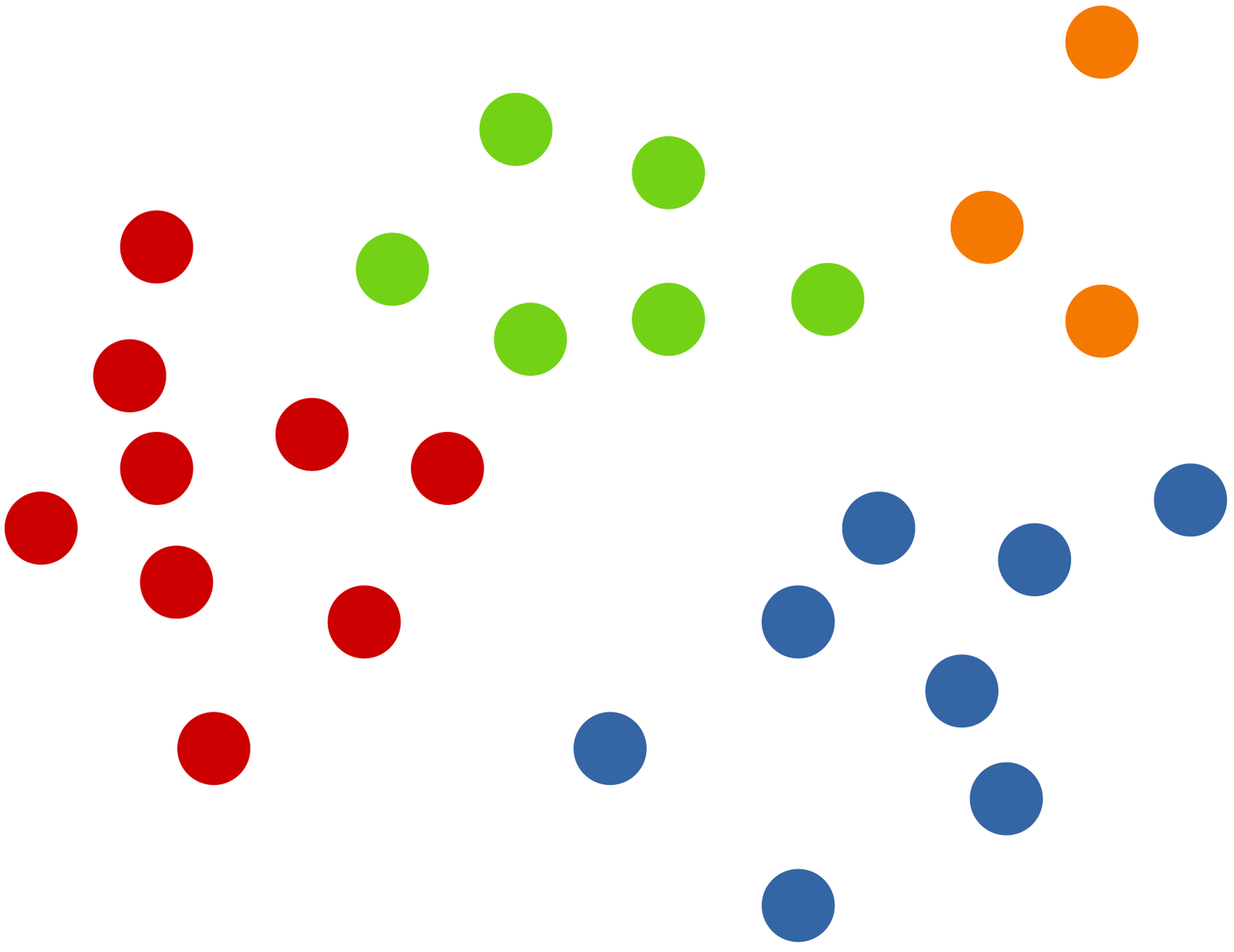
Linear multi-class classification

A simple example

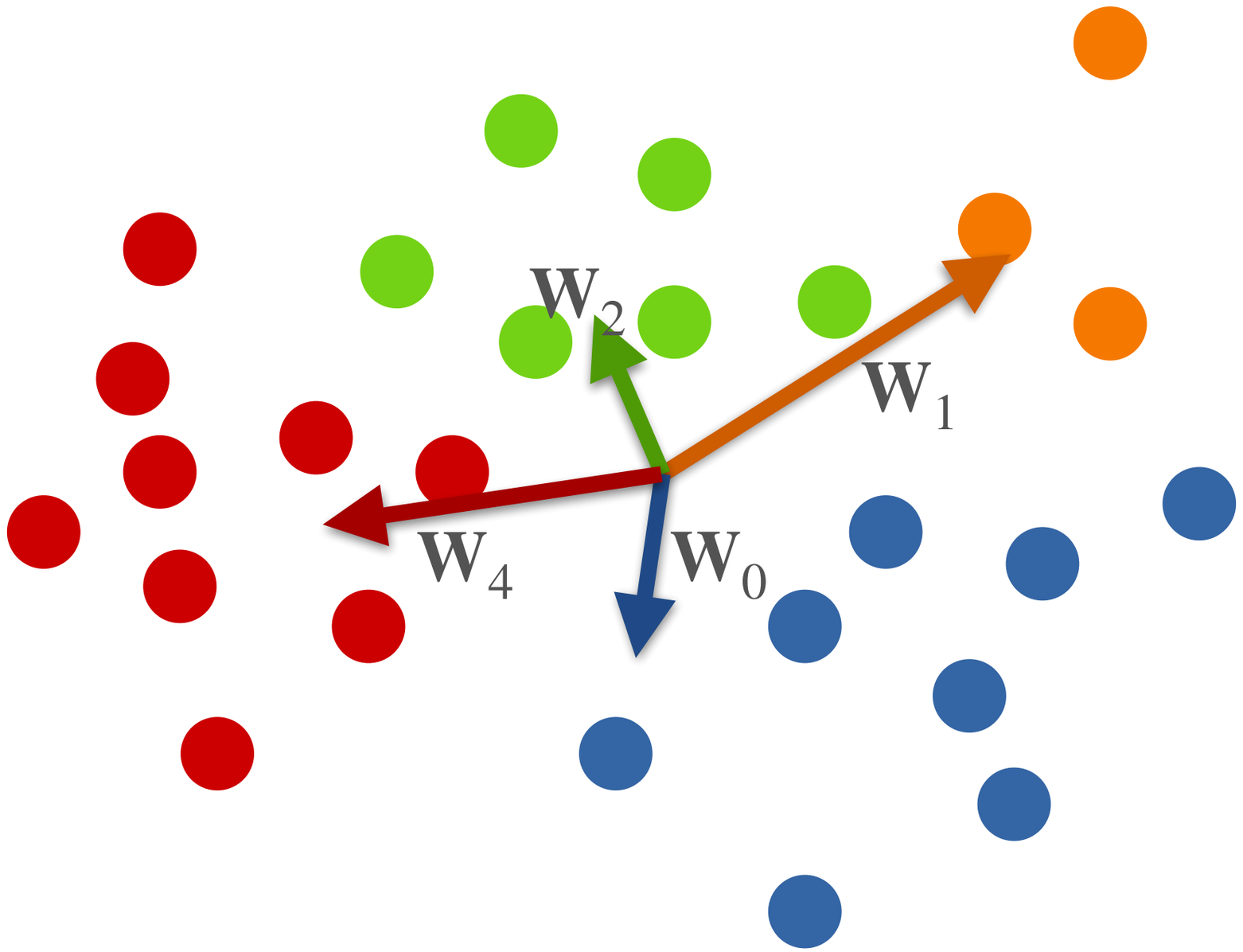
Multinomial linear classifier



Multinomial linear classifier

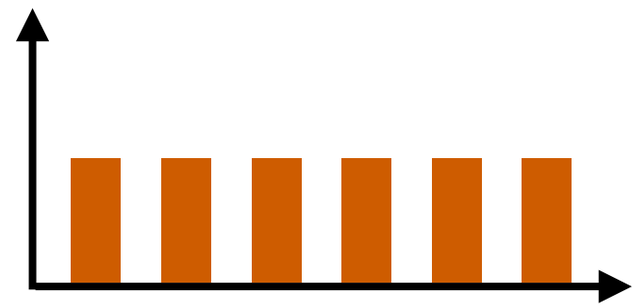
- Input: $\mathbf{x} \in \mathbb{R}^d$ (tensor)
- Label: $y \in \{0,1,\ldots,k\}$
- Parameters: $\mathbf{W} \in \mathbb{R}^{d \times k}, \mathbf{b} \in \mathbb{R}^k$
- $P(y) = \text{softmax}(\mathbf{W}^\top \mathbf{x} + \mathbf{b})_y$

Multinomial linear classifier



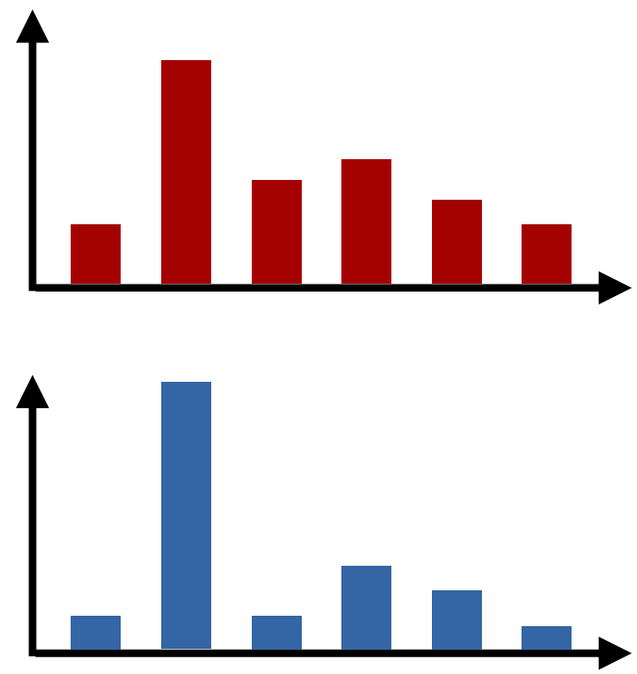
Softmax function

- Maps $\mathbb{R}^k$ to probability p



- $\mathbf{o} = \mathbf{W}^\top \mathbf{x} + \mathbf{b}$

$$\text{softmax}(\mathbf{o})_i = \frac{e^{o_i}}{\sum_{i'} e^{o_{i'}}}$$



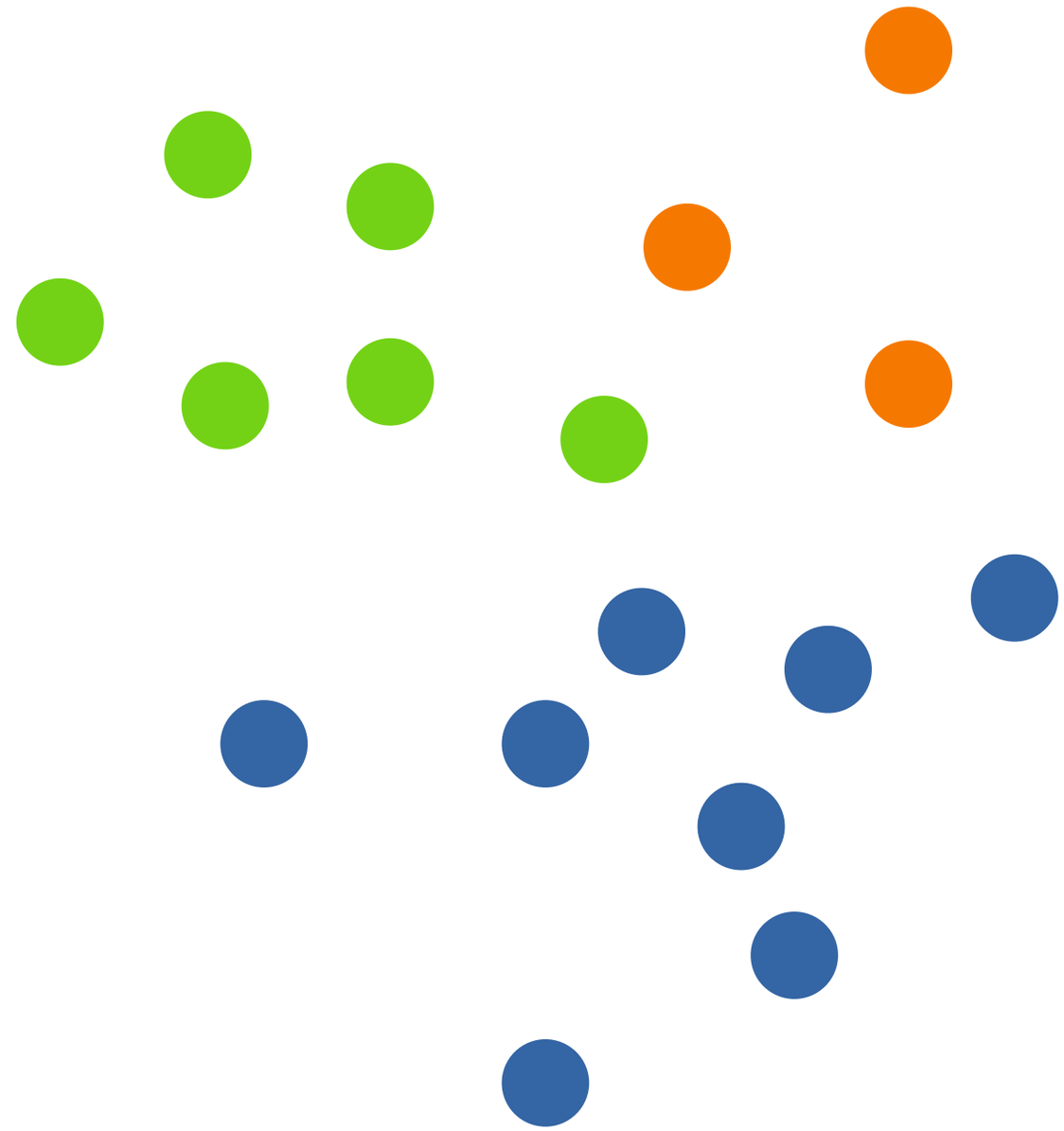
always sum to one

Multinomial logistic regression

- Input: $\mathbf{x} \in \mathbb{R}^d$ (tensor)
- Label: $y \in \{0,1,\ldots,k\}$
- Parameters: $\mathbf{W} \in \mathbb{R}^{d \times k}, \mathbf{b} \in \mathbb{R}^k$
- $P(y) = \text{softmax}(\mathbf{W}^\top \mathbf{x} + \mathbf{b})_y$
- Loss: $-\log p(y)$

One vs all classification

- Train n binary classifier
  - Not calibrated
- Do not use this!



Summary

- Multinomial logistic regression
  - Multiple outputs
  - Softmax instead of sigmoid