

Deep Learning for NLP

Sorokin Semen

Grading system

Three labs (each 0.17)
Three / five (not decided yet) test

50% of cumulative assessment
50% of cumulative assessment

If cumulative assessment ≥ 6 :
free to go or increase your score on the exam
else:
pass the exam (a result of the exam \geq cumulative assessment)

Exam - two topics from different lectures or
(Mb, if you are lucky) questions like "What is the name of the lecturer?" or
"Classify (manually) this image"



Syllabus

- Linear / Logistic regression
- Word embeddings
- Tricks in DL
- Convolution NN
- Recurrent NN
- Language modeling
- Sequence tagging

Framework

Simple

Hard



Deep Learning

Classic Machine Learning



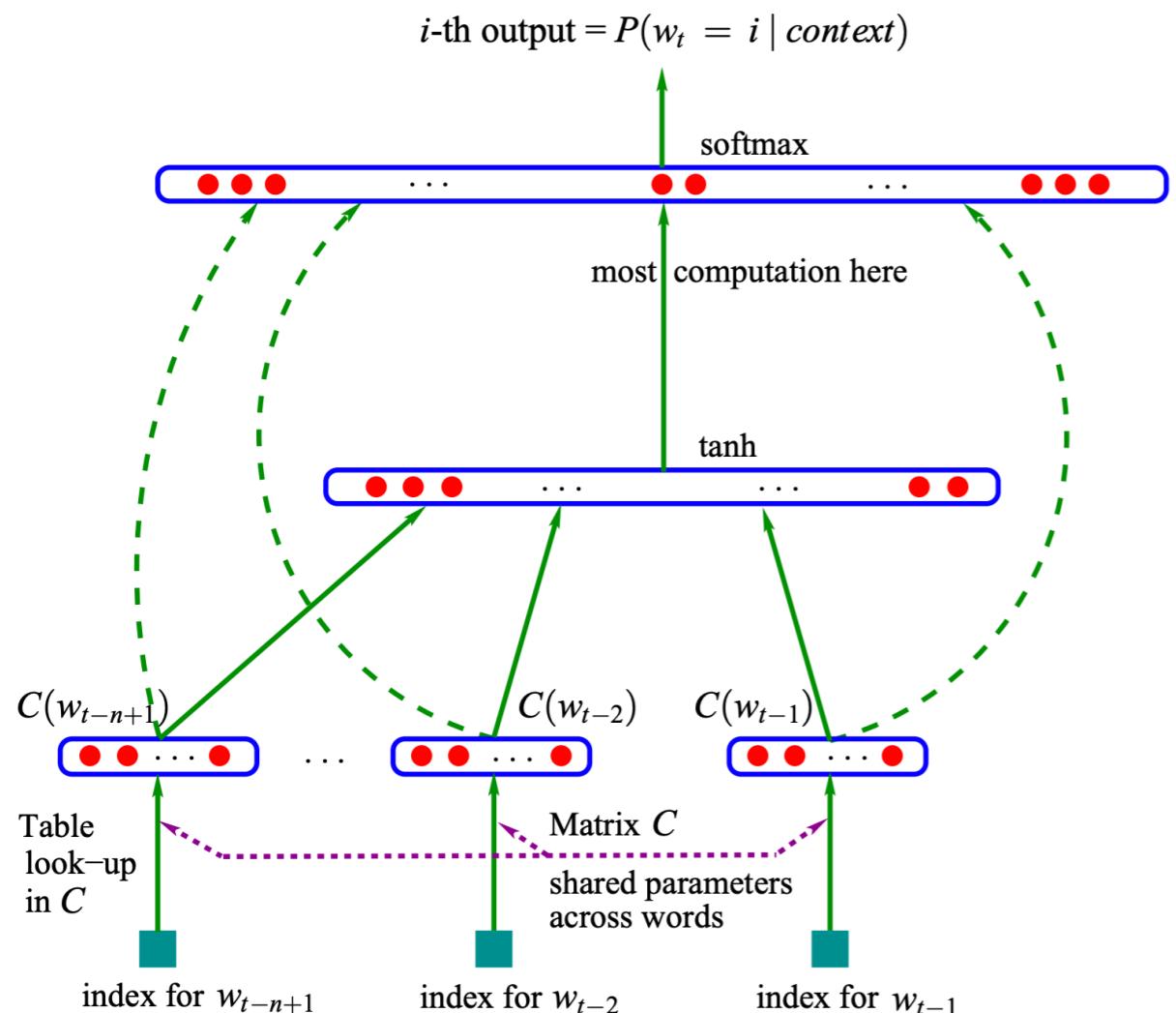
Deep Learning



Deep Learning

Neural Probabilistic Language Model

2003

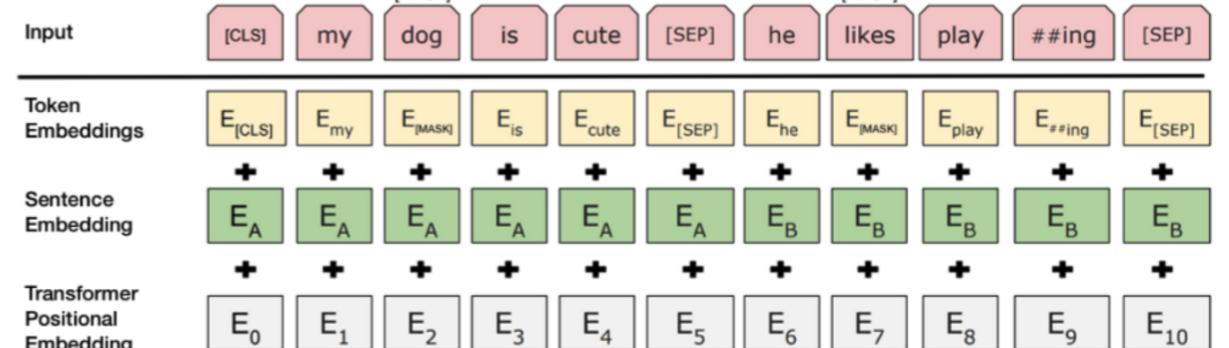


BERT

Q42018

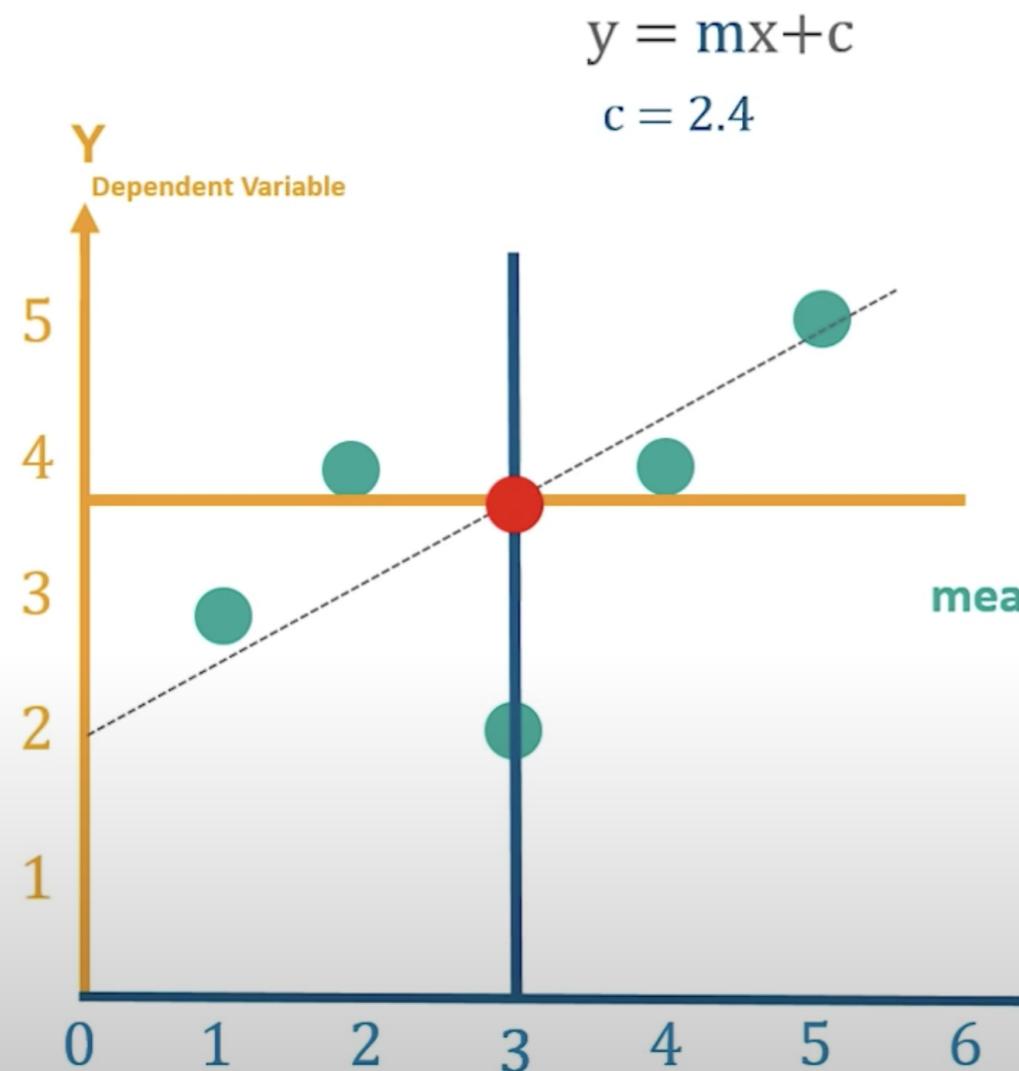
Transformer Encoder
24 Layers
Transformer Encoder

Transformer Encoder



ML Recap

Understanding Linear Regression Algorithm



| x | y | $x - \bar{x}$ | $y - \bar{y}$ | $(x - \bar{x})^2$ | $(x - \bar{x})(y - \bar{y})$ |
|-----|-----|---------------|---------------|-------------------|------------------------------|
| 1 | 3 | -2 | -0.6 | 4 | 1.2 |
| 2 | 4 | -1 | 0.4 | 1 | -0.4 |
| 3 | 2 | 0 | -1.6 | 0 | 0 |
| 4 | 4 | 1 | 0.4 | 1 | 0.4 |
| 5 | 5 | 2 | 1.4 | 4 | 2.8 |

$$\Sigma = 10 \quad \Sigma = 4$$

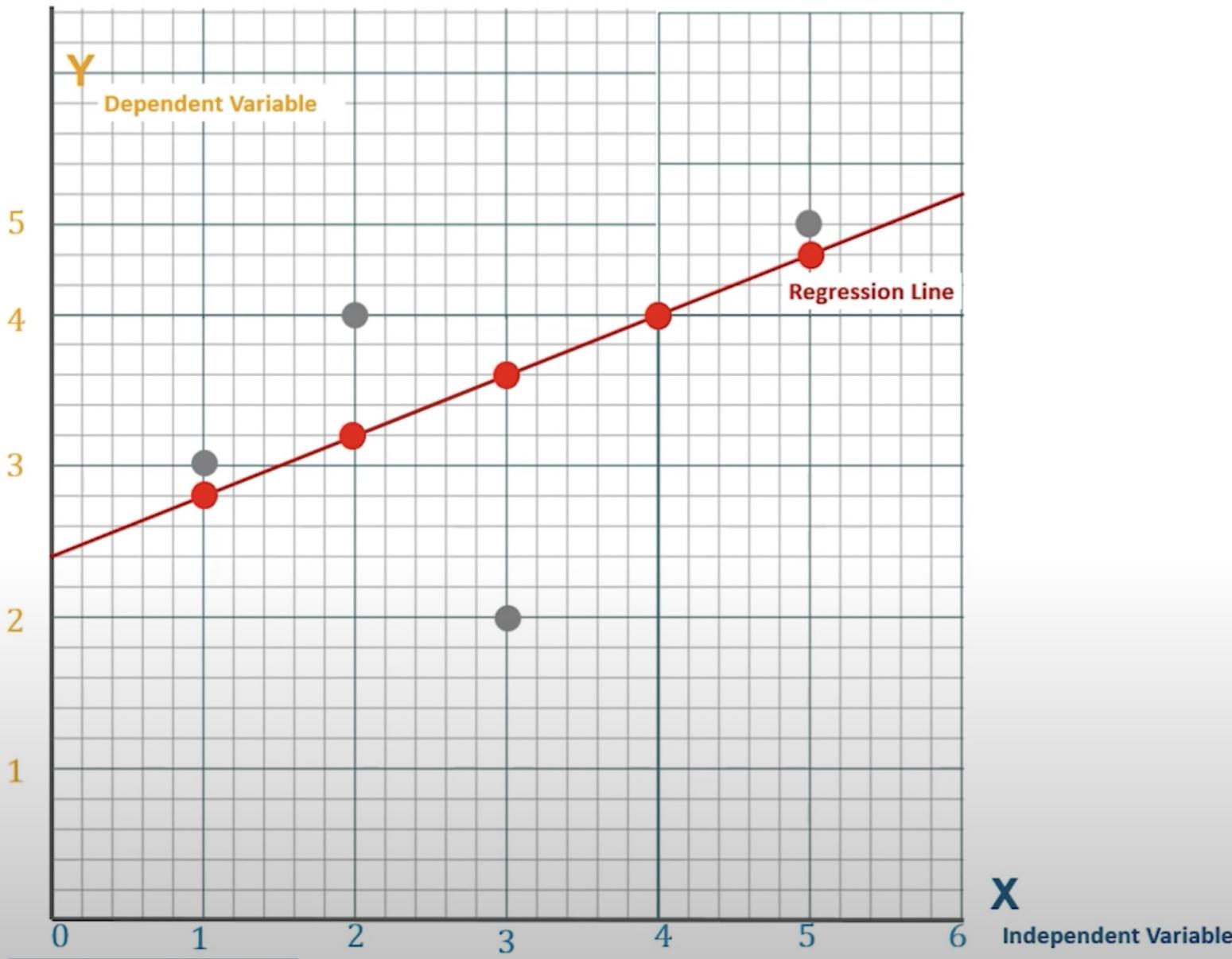
$$m = \sum \frac{(x - \bar{x})(y - \bar{y})}{(x - \bar{x})^2} = \frac{4}{10}$$

$$m = 0.4$$

$$c = 2.4$$

$$y = 0.4x + 2.4$$

Mean Square Error

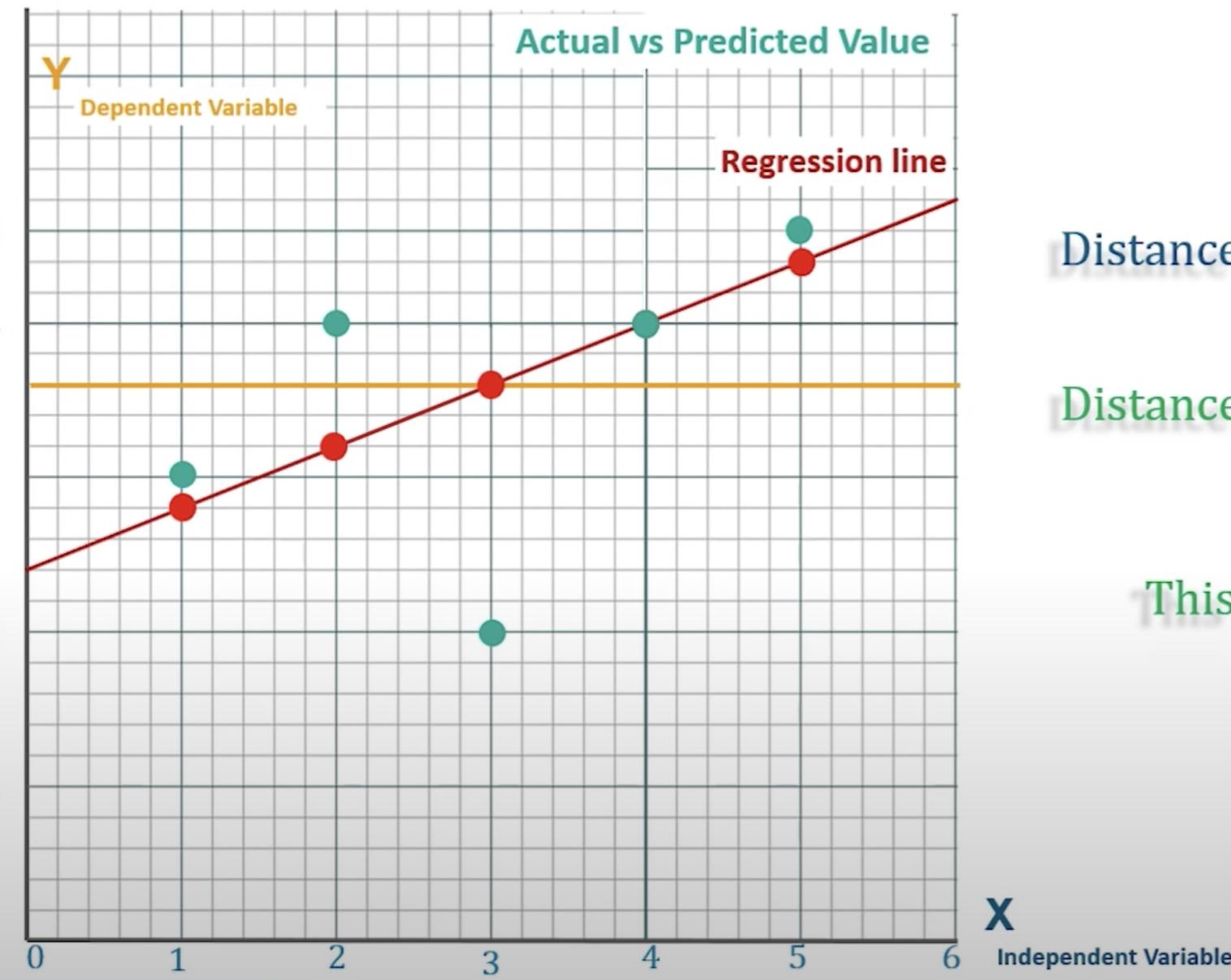


$$\begin{aligned}m &= 0.4 \\c &= 2.4 \\y &= 0.4x + 2.4\end{aligned}$$

For given $m = 0.4$ & $c = 2.4$, lets predict values for y for $x = \{1,2,3,4,5\}$

$$\begin{aligned}y &= 0.4 \times 1 + 2.4 = 2.8 \\y &= 0.4 \times 2 + 2.4 = 3.2 \\y &= 0.4 \times 3 + 2.4 = 3.6 \\y &= 0.4 \times 4 + 2.4 = 4.0 \\y &= 0.4 \times 5 + 2.4 = 4.4\end{aligned}$$

Calculation of R^2



Distance actual - mean

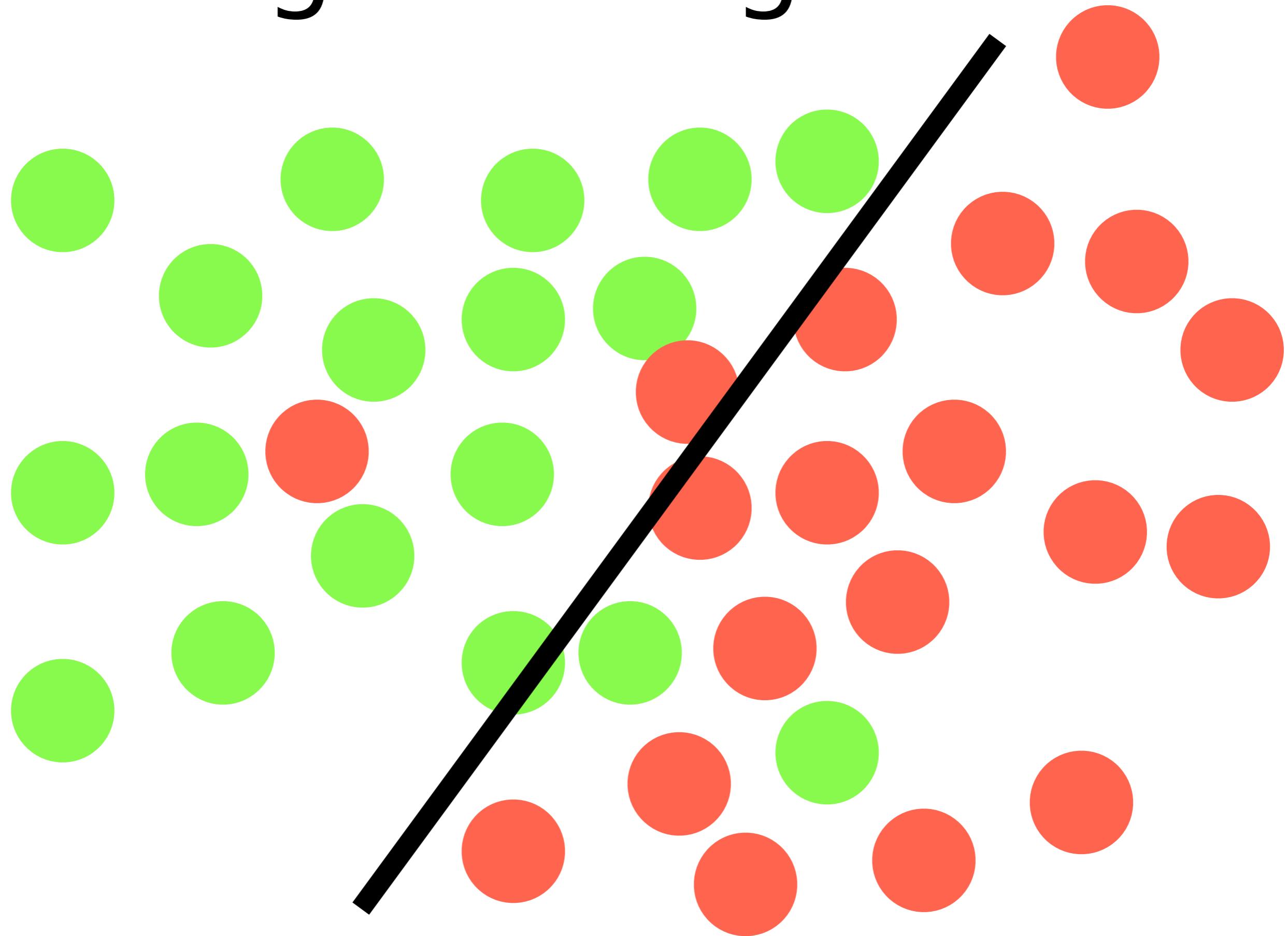
vs

Distance predicted - mean

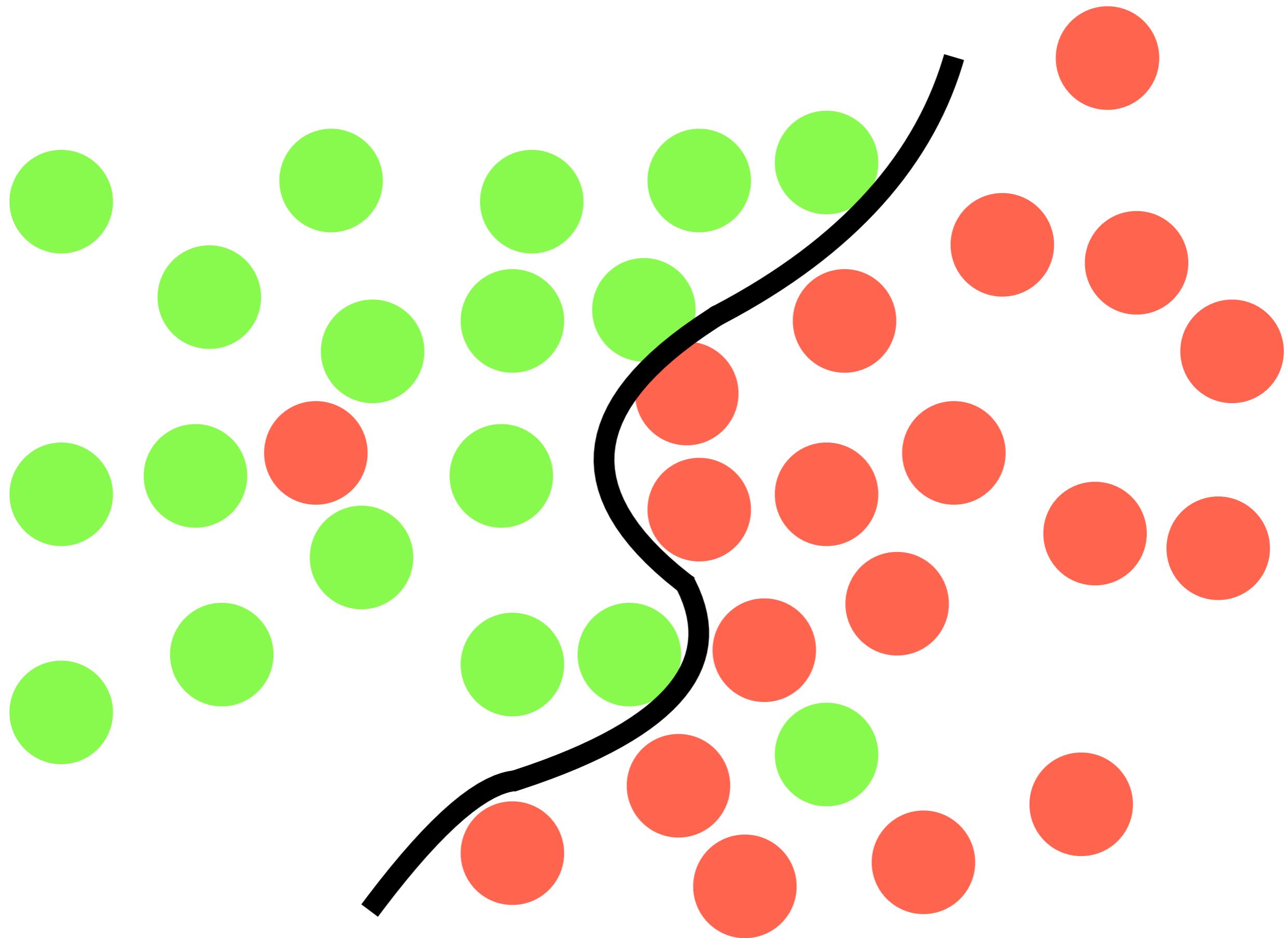
This is nothing but $R^2 =$

$$R^2 = \frac{\sum (y_p - \bar{y})^2}{\sum (y - \bar{y})^2}$$

Logistic Regression

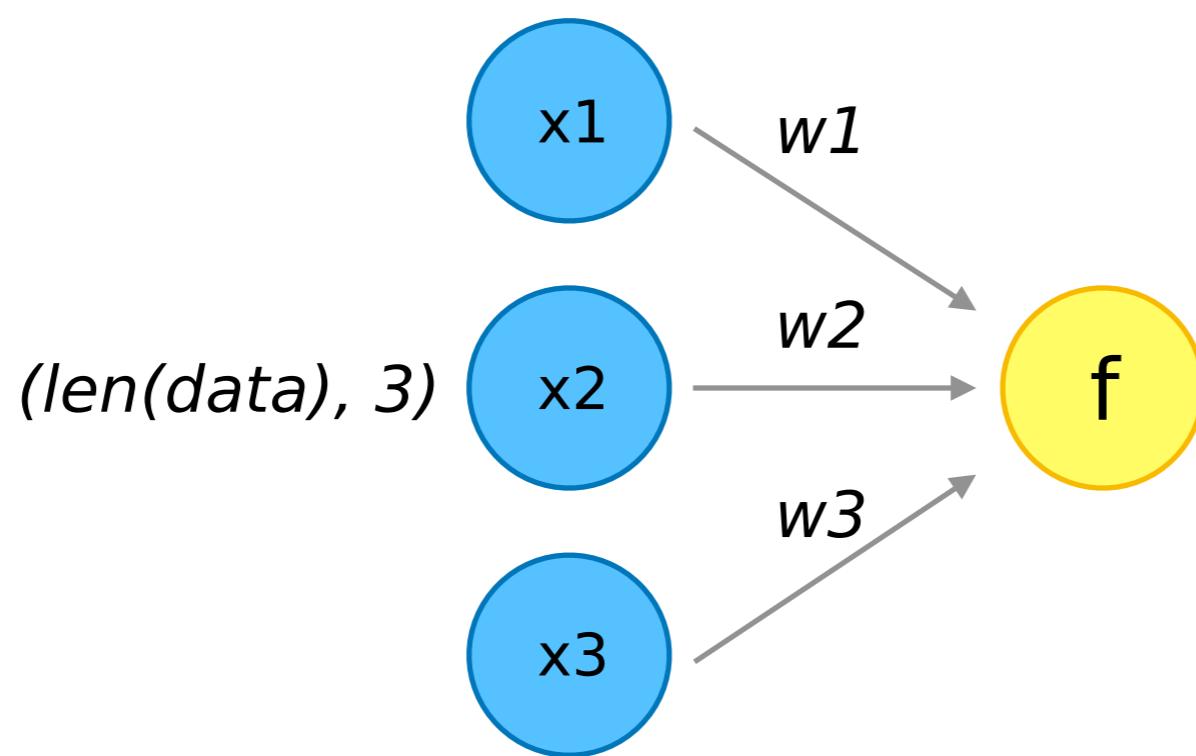


Neural Networks



Logistic Regression

Inference



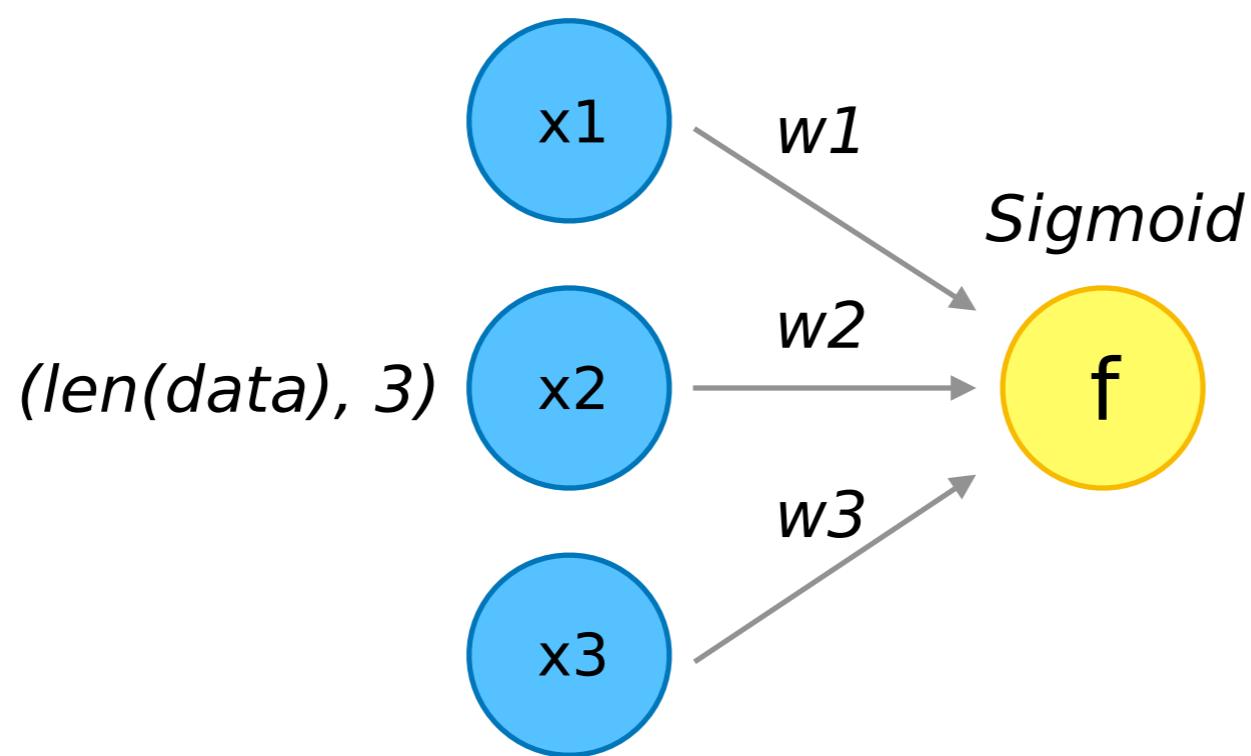
Logistic Regression

Inference

$$f = \mathbf{x} * \mathbf{w} + b$$

Logistic Regression

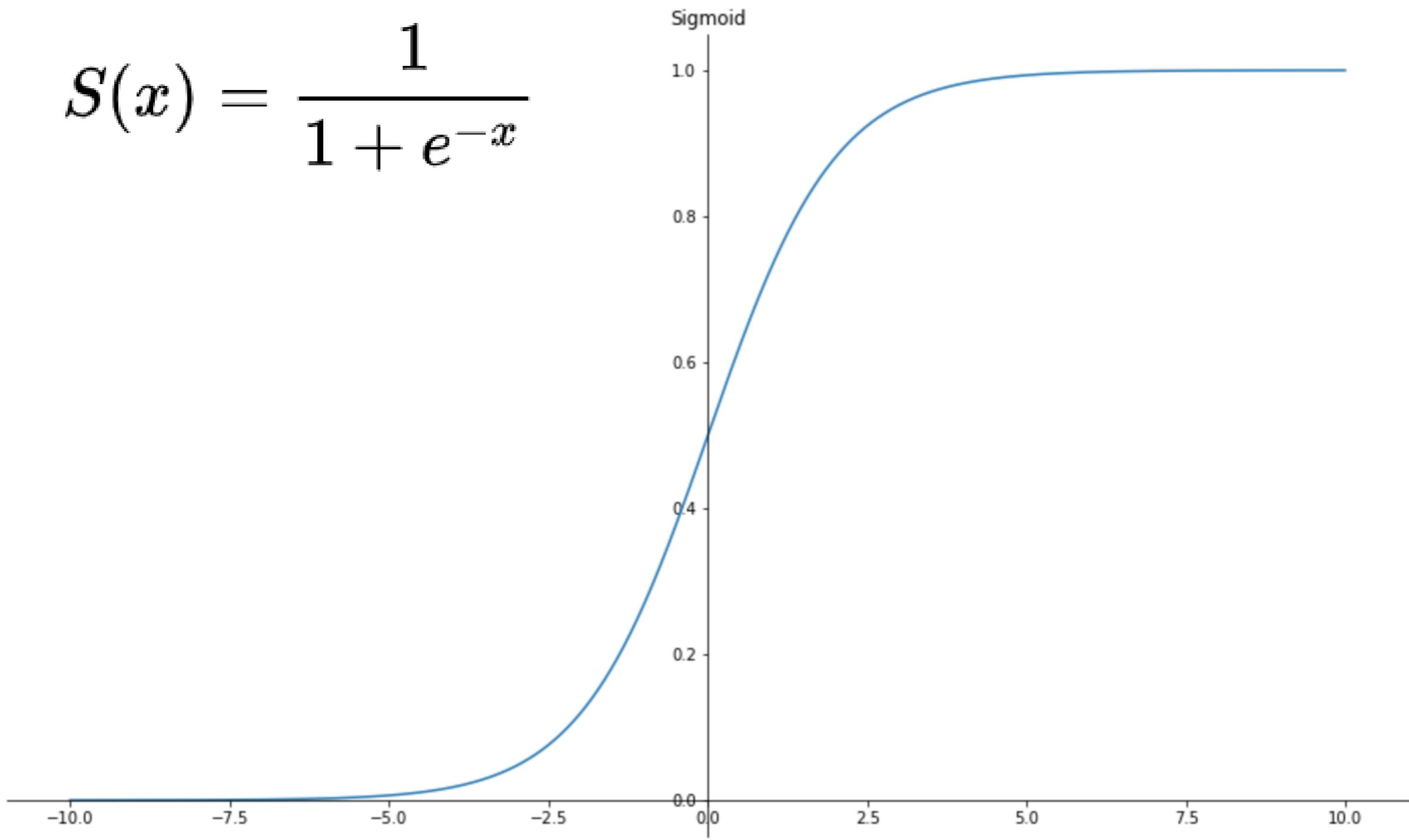
Inference



Logistic Regression

Sigmoid

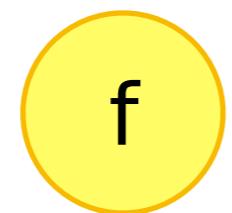
$$S(x) = \frac{1}{1 + e^{-x}}$$



Logistic Regression

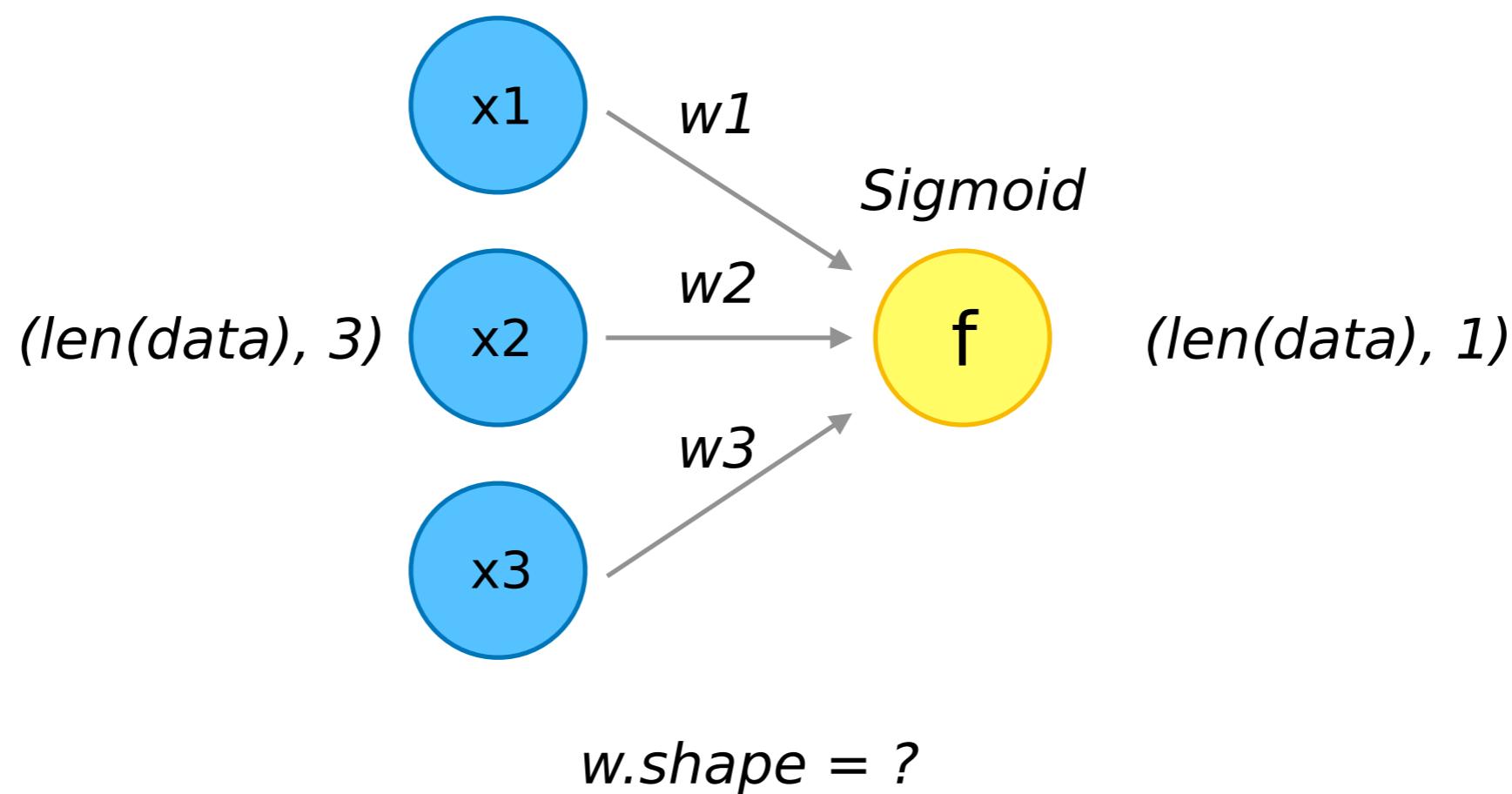
Inference

$$S(x) = \frac{1}{1 + e^{-x}}$$


$$f = 1 / (1 + \exp(-(\mathbf{x} * \mathbf{w} + b)))$$

Logistic Regression

Inference



Logistic Regression

Dot Product

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{in}b_{nj} = \sum_{s=1}^n a_{sn}b_{sj}$$

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{pmatrix}, B = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{pmatrix}$$


$$f = \mathbf{np.dot(x, w)} + b$$

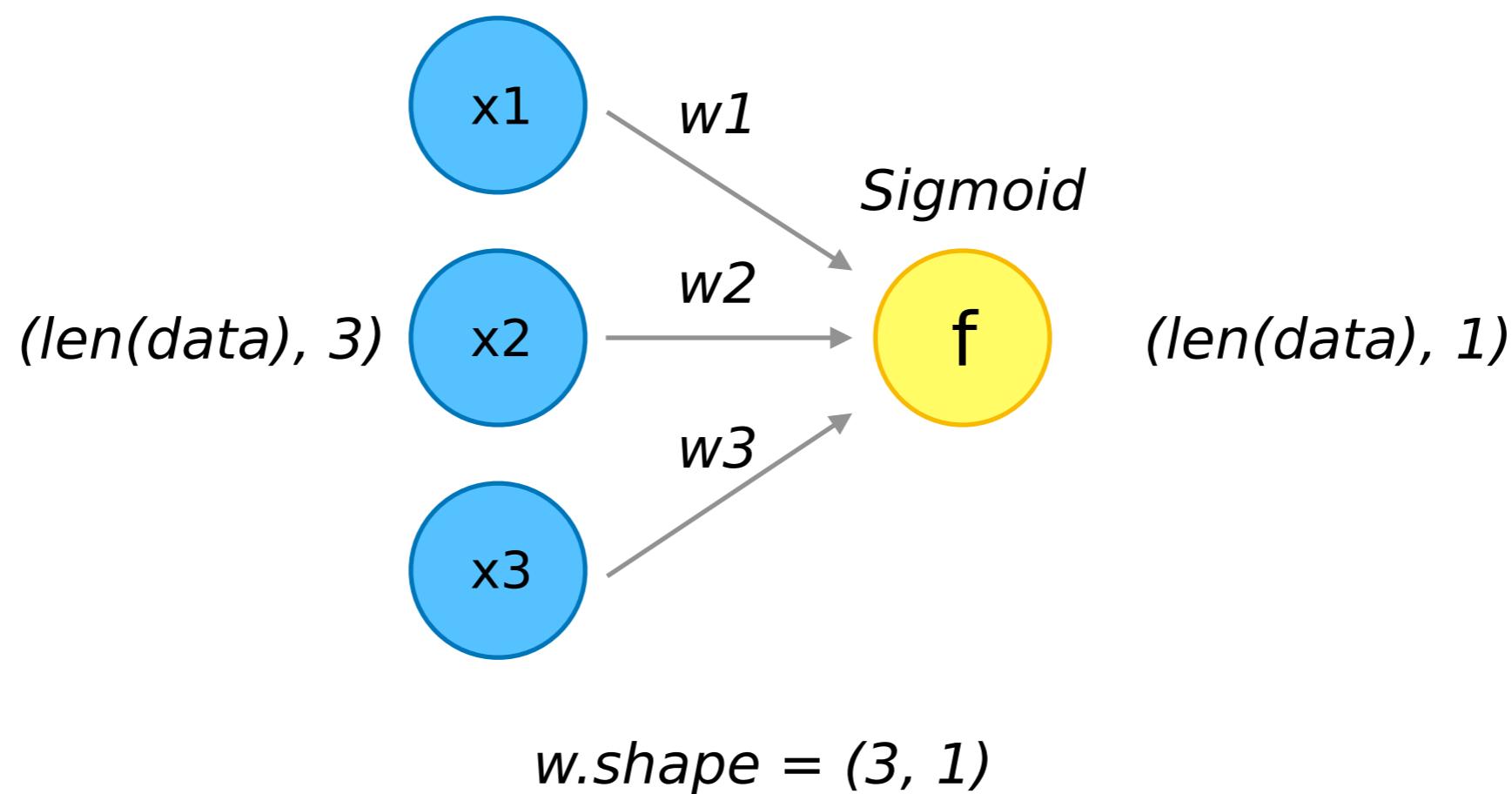
$$AB = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{pmatrix} = \begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} & a_{11}b_{13} + a_{12}b_{23} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} & a_{21}b_{13} + a_{22}b_{23} \\ a_{31}b_{11} + a_{32}b_{21} & a_{31}b_{12} + a_{32}b_{22} & a_{31}b_{13} + a_{32}b_{23} \end{pmatrix}$$

A.shape = (p, m)

B.shape = (n, k) **np.dot(A, B).shape = (p, k) if m == n**

Logistic Regression

Inference



Difference between classification and regression task

Classification predictive modeling problems are different from regression predictive modeling problems.

- Classification is the task of predicting a discrete class label.
- Regression is the task of predicting a continuous quantity.

There is some overlap between the algorithms for classification and regression; for example:

- A classification algorithm may predict a continuous value, but the continuous value is in the form of a probability for a class label.
- A regression algorithm may predict a discrete value, but the discrete value in the form of an integer quantity.

[More info](#)