

CS 331

Computer Vision

10 / 22 / 2026

Instructor: Michael Eckmann

Topics

- Questions / Comments?
- Show example program that learns a quadratic
- Show some more numpy
 - Matrix multiplication
 - Elementwise multiplication
 - Different ways to sum, based on axis
- Logistic regression

More Numpy

- Matrix multiplication
- vs.
- Elementwise multiplication
- Different uses of np.sum

Logistic Regression

- Logistic regression is a classification technique.
- It is not well named, because regression means outputting a real valued scalar and classification means outputting a discrete label. So, don't be fooled by the regression in the name.
- We will start off with 2-class classification --- that is, predict one of two possible class labels. (e.g. distinguish only between pictures of cats and pictures of dogs)
- y , the output, is in the set $\{0, 1\}$ (e.g. 0 could mean cat, 1 could mean dog)

Logistic Regression

- Our prediction for linear regression was

$$\hat{y} = w^T x$$

$$w = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_n \end{bmatrix} \quad x = \begin{bmatrix} x_0=1 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

- For logistic regression, we wrap that in the sigmoid function g so that we can interpret the result as an estimated probability that $y=1$, given input x and weights w .

$$\hat{y} = g(w^T x)$$

Logistic Regression

- The sigmoid function in general is

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

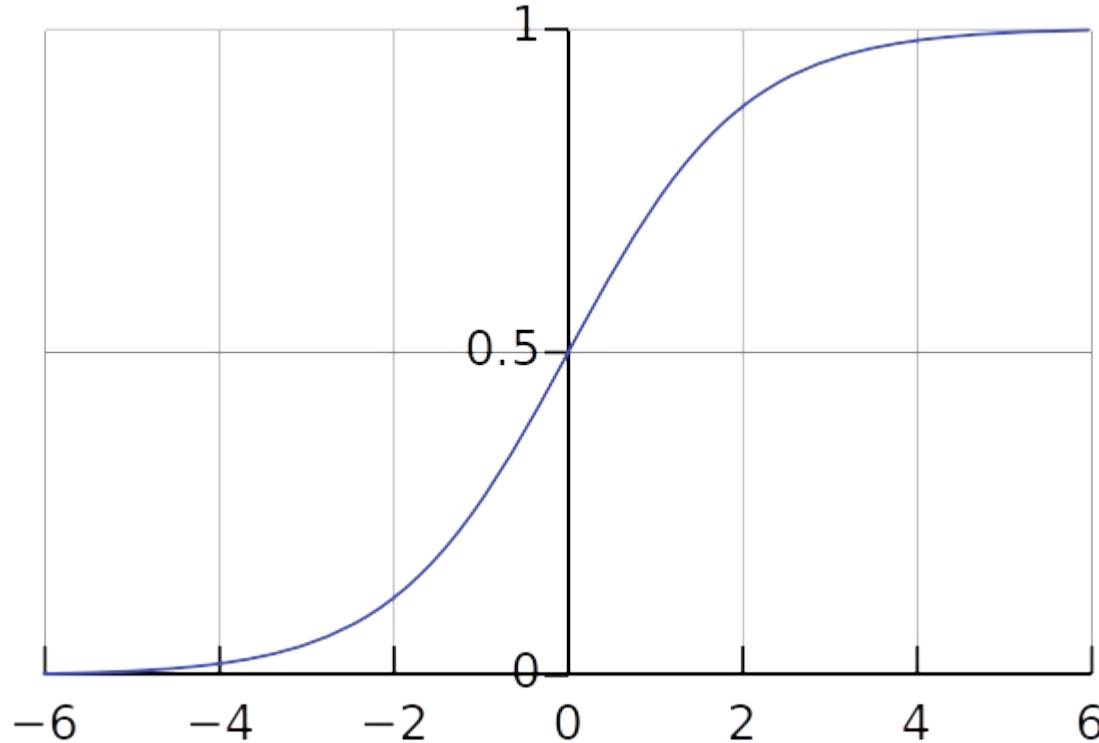
- For logistic regression, the variable of the sigmoid is

which results in this for the prediction

$$g(w^T x) = \frac{1}{1 + e^{-w^T x}}$$

$$w^T x$$

Sigmoid Function Plot



Note: for positive inputs, output is > 0.5 and ≤ 1 , and for negative inputs, output is < 0.5 and ≥ 0

Logistic Regression

- As I said before, we will interpret the result of the function as the estimated probability that $y=1$, given input x and weights w .
- We can write this as: $p(y=1 \mid x, w)$
- Since there are two classes only and the result is an estimated probability, the sum of the two probabilities must equal 1, so
- $p(y=0 \mid x, w) = 1 - p(y=1 \mid x, w)$
- We will predict $y=0$ when the estimated probability for class 0 is > 0.5 . Similarly we will predict $y=1$ when the estimated probability for class 1 is > 0.5 .

Logistic Regression

- Let's look back at the graph of sigmoid.
- For what values of input will we predict $y=1$ and for what values of input will we predict $y=0$?

Logistic Regression

- Let's look back at the graph of sigmoid.
- Recall: we predict $y=1$ when output is ≥ 0.5 and $y=0$ when output is < 0.5
- For what values of input will we predict $y=1$ and for what values of input will we predict $y=0$?

Logistic Regression

- Let's look back at the graph of sigmoid.
- Recall: we predict $y=1$ when output is ≥ 0.5 and $y=0$ when output is < 0.5
- For what values of input will we predict $y=1$ and for what values of input will we predict $y=0$?
- Answer: when the input is positive we predict $y=1$, when the input is negative we predict $y=0$

Logistic Regression

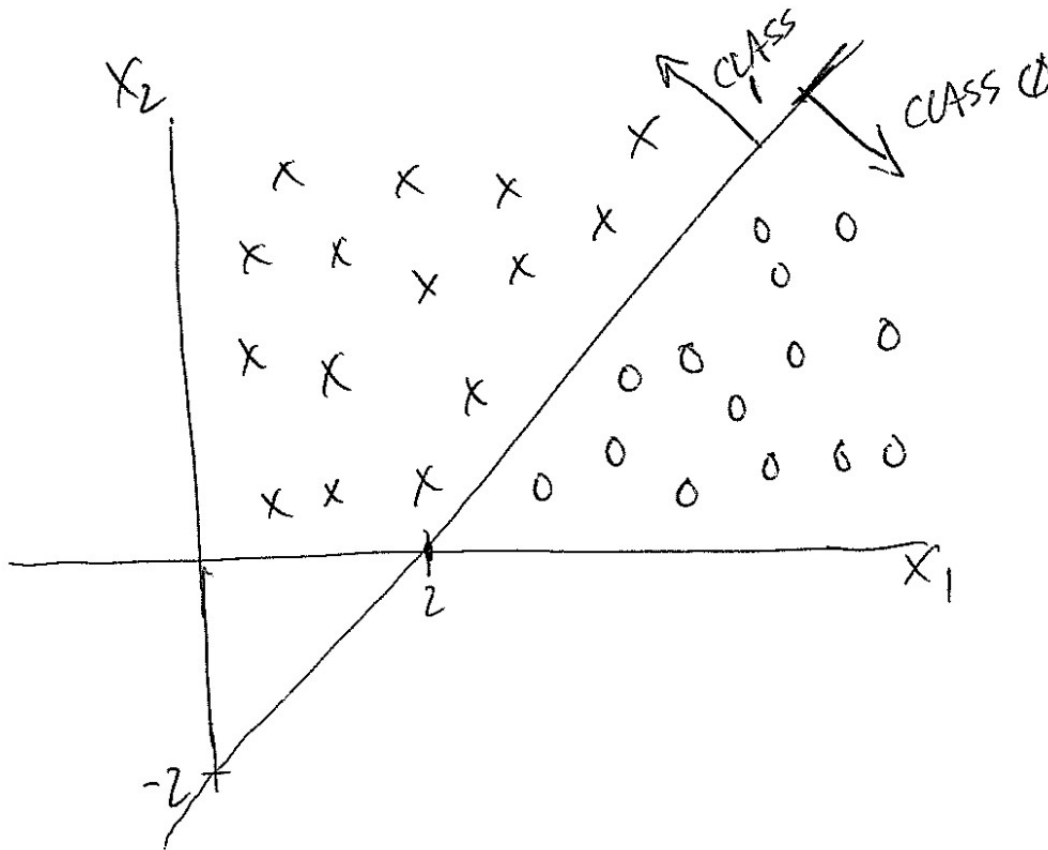
- Let's look back at the graph of sigmoid.
- Recall: we predict $y=1$ when output is ≥ 0.5 and $y=0$ when output is < 0.5
- For what values of input will we predict $y=1$ and for what values of input will we predict $y=0$?
- Answer: when the input is positive we predict $y=1$, when the input is negative we predict $y=0$
- What's our input to the sigmoid function?

Logistic Regression

- Let's look back at the graph of sigmoid.
- Recall: we predict $y=1$ when output is ≥ 0.5 and $y=0$ when output is < 0.5
- For what values of input will we predict $y=1$ and for what values of input will we predict $y=0$?
- Answer: when the input is positive we predict $y=1$, when the input is negative we predict $y=0$
- What's our input to the sigmoid function?
- Answer:
- So, we only have to check the sign of
to predict 1 or 0.

Logistic Regression

- Let's look at a linear decision boundary for two inputs and some training examples.



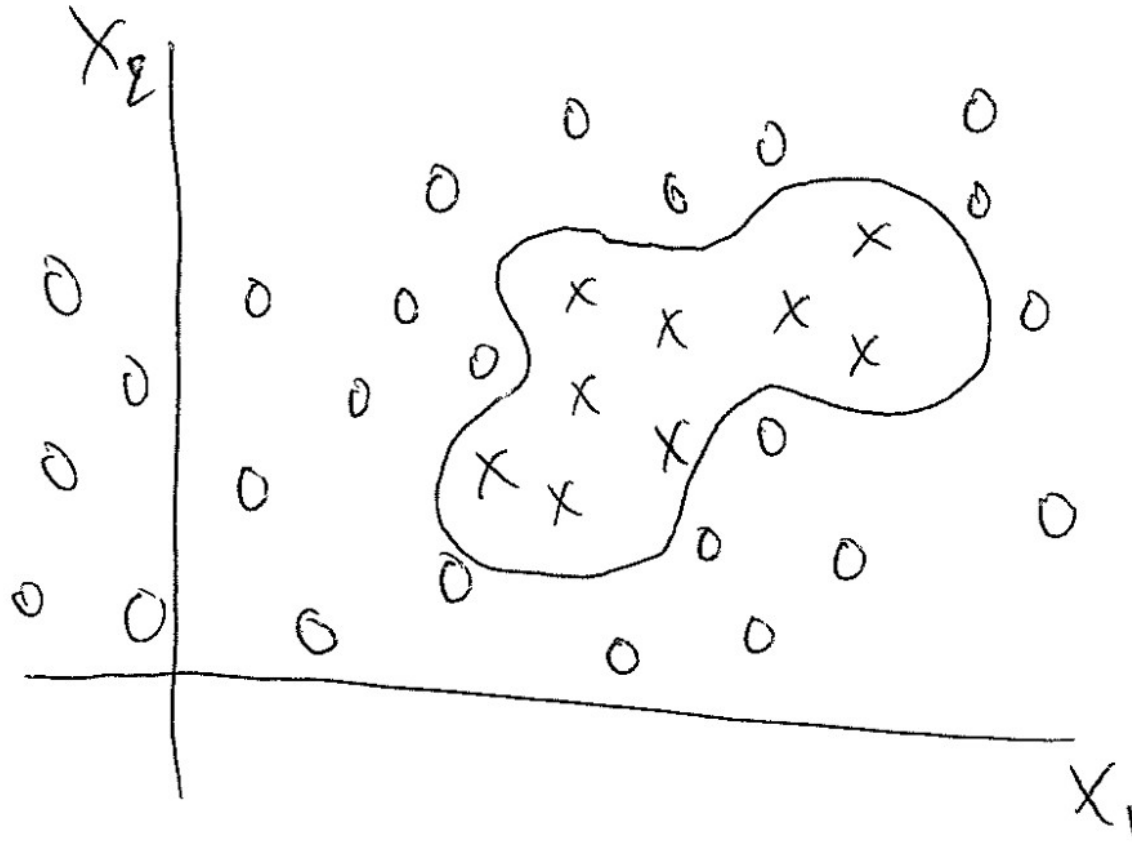
SUPPOSE
 $w_0 = 2$ $w_1 = -1$
 $w_2 = 1$

$w_0x_0 + w_1x_1 + w_2x_2 = 0$
is the line.

when ≥ 0
we have class 1
when < 0
we have class 0.

Logistic Regression

- Decision boundaries can be much more complex shaped



Logistic Regression

- Recall the loss function for linear regression. We might consider the same loss function for logistic regression, except we replace the $\hat{y}$ with the sigmoid.

$$J(w) = \frac{1}{2m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})^2$$

$$\hat{y} = w^T x \text{ for linear regression.}$$

COULD CONSIDER SAME FOR LOGISTIC REG.

$$\text{BUT w/ } \hat{y} = g(w^T x) = \frac{1}{1 + e^{-w^T x}}.$$

Logistic Regression

- The problem with that loss function is it is NOT convex. Recall convex and non-convex functions and the problems that occur with gradient descent on non-convex functions.

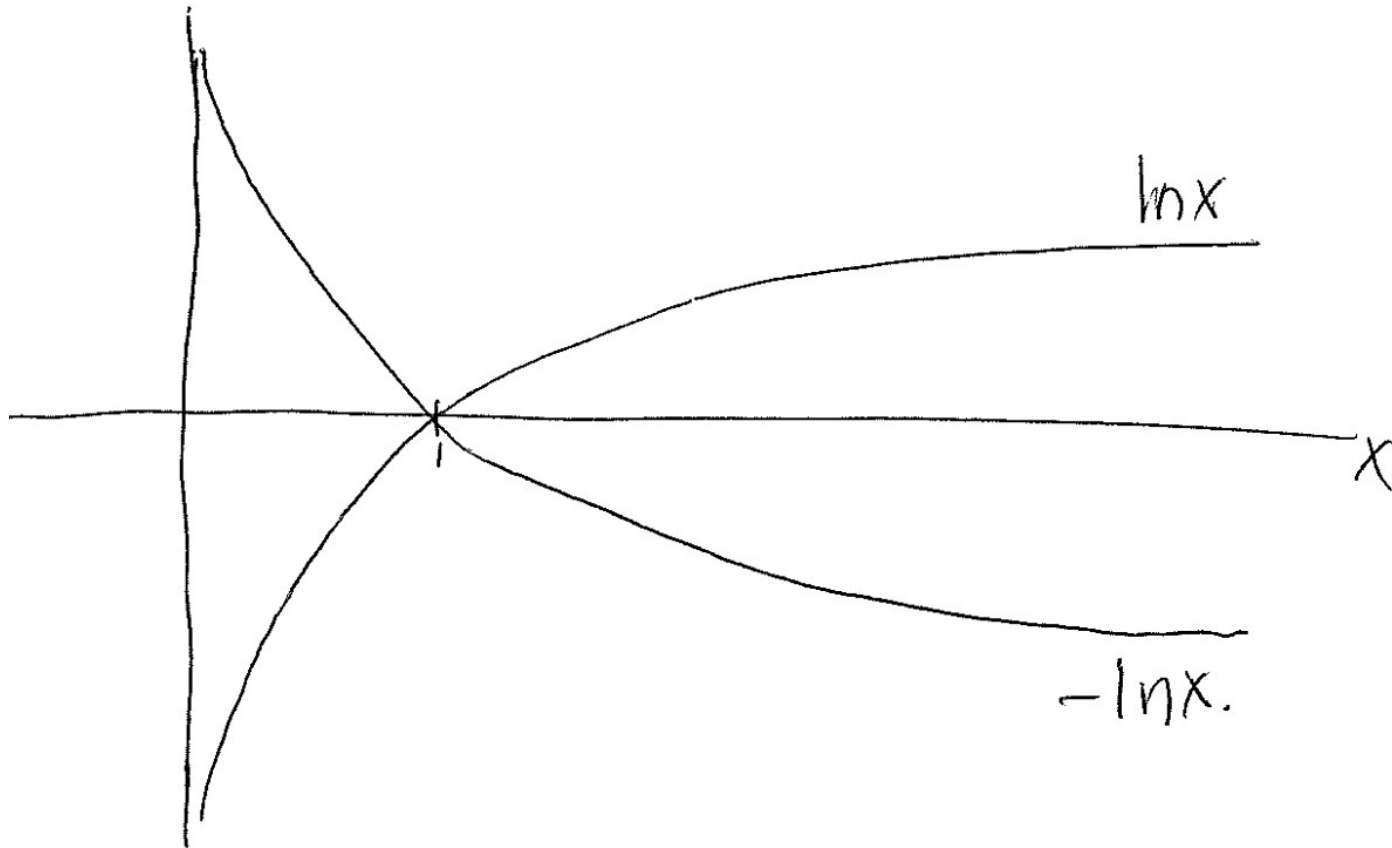
Logistic Regression

- We want a convex loss function if possible, so that gradient descent is guaranteed to converge to the minimum (as long as the learning rate is not too large of course)
- Here's the loss function we'll use:

$$\text{COST}(\hat{y}) = -y \ln(\hat{y}) - (1-y) \ln(1-\hat{y})$$

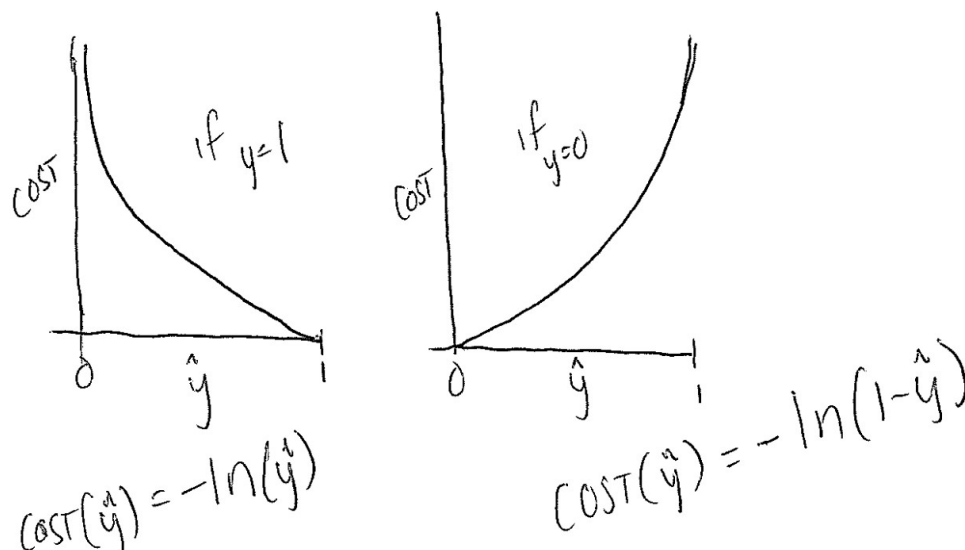
Logistic Regression

- Recall the graph of the log (and -log)



Logistic Regression

- Let's look at graphs of the two parts of the loss function



- Notice: if predicted y and actual y match, loss is 0. If the predicted y and actual y do not match, loss is high

Logistic Regression

- The loss function for Logistic Regression over all the training examples is:

COST OVER ALL TRAINING EXAMPLES
FOR LOGISTIC REGRESSION IS:

$$J(w) = -\frac{1}{m} \sum_{i=1}^m \left(y^{(i)} \ln(\hat{y}^{(i)}) + (1 - y^{(i)}) \ln(1 - \hat{y}^{(i)}) \right)$$

Logistic Regression

- The gradient of the loss function is similar to linear regression gradient except the sigmoid of $w^T x$ is substituted for y rather than $w^T x$

For all j 's $0 \dots n$

$$\frac{\partial J}{\partial w_j} = \frac{1}{m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)}) x_j^{(i)}$$

WHERE $\hat{y}^{(i)} = 1 / (1 + e^{-w^T x^{(i)}})$

NOTE: j is index into w & x $0 \dots n$

i is " " training examples $1 \dots m$.

Logistic Regression

- We do gradient descent using the gradient from last slide to get the weights (learned parameters) that minimize our loss function.
- That is how logistic regression works for 2-class classification.
- What if we want to do multiclass classification (that is, more than 2 classes)? e.g. differentiate among 5 classes: cat, dog, frog, elephant, fish

Logistic Regression

- To do multiclass classification we can use a technique called one-vs-all. Say we have 5 classes 0,1,2,3,4
- We train 5 logistic regression 2-class classifiers:
 - train 0 vs. all (1,2,3,4 all considered the same non-0 class)
 - train 1 vs. (0,2,3,4)
 - train 2 vs. (0,1,3,4)
 - train 3 vs. (0,1,2,4)
 - train 4 vs. (0,1,2,3)
- To predict, we run all five and whichever gives highest probability we go with it