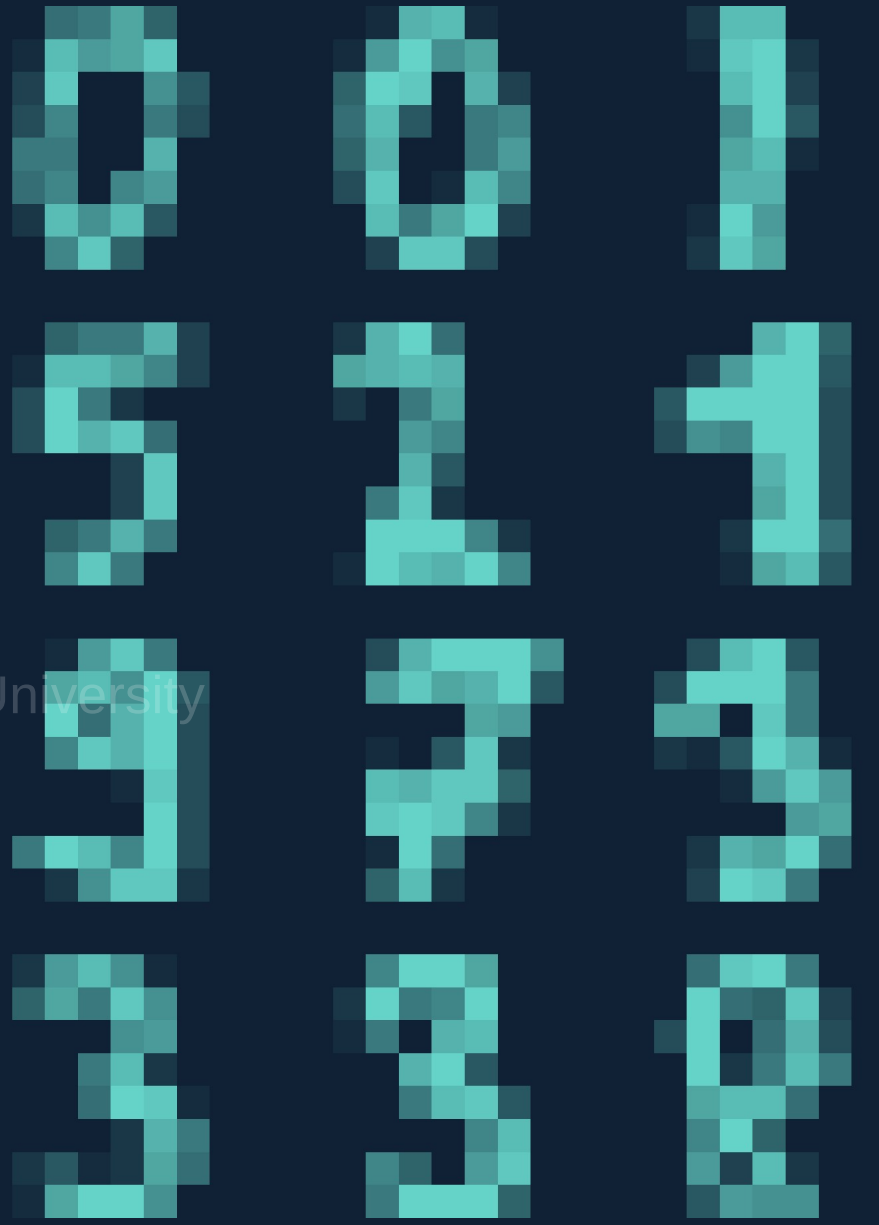
Training a classifier

Mohammad Sabokrou | New Uzbekistan University

From prediction errors to parameter updates

Mohammad Sabokrou
New Uzbekistan University



Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

COURSE CONNECTION

From scores to learning

Week 5 taught us how to predict. Week 6 explains how a classifier improves.

ALREADY KNOW

An image becomes a feature vector x.

Scores are computed as z = Wx + b.

Softmax converts scores into probabilities.

TODAY WE LEARN

Compare predictions with targets.

Measure errors with MSE or cross-entropy.

Adjust W and b to reduce the loss.

Mohammad Sabokrou | New Uzbekistan University

Big idea: a correct label tells us what the model should have predicted.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

LEARNING PATH

Four steps from prediction to training

Each section builds on the previous one; work through the numerical examples in order.

01

REGRESSION & MSE

Learn loss with simple subtraction and squaring.

Slides 5-12

02

BINARY & MULTICLASS LOSS

Understand labels 0/1, BCE and cross-entropy.

Slides 13-28

Mohammad Sabokrou | New Uzbekistan University

03

GRADIENTS & UPDATES

Calculate a derivative, then change one weight.

Slides 30-47

04

TRAINING & EVALUATION

Choose a learning rate; inspect mistakes and limits.

Slides 48-59

Appendices: NumPy implementations, numerical gradient checks, and formula reference.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

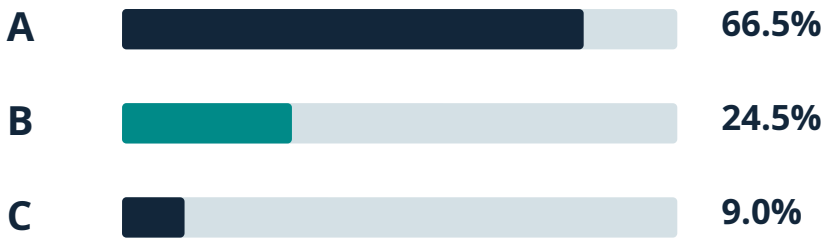
The model predicts A, but the answer is B

One small three-class example anchors the rest of the lecture.

MODEL OUTPUT

Scores z = [2, 1, 0]

Probabilities after softmax:



Mohammad Sabokrou | New Uzbekistan University

PREDICTED CLASS: A

TRUE CLASS: B

Question: how can we describe how wrong this prediction is, then improve the model?

Start with easier numerical prediction errors (regression and MSE); then return to class probabilities.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Predicting a numerical measurement

Regression predicts a quantity whose numerical distance matters.

Example i	Input feature x_i	Observed length y_i
1	0	1 cm
2	1	3 cm
3	2	5 cm

Mohammad Sabokrou | New Uzbekistan University

y is the observed target. $\hat{y}$ is the model's numerical prediction.

These three invented examples keep the arithmetic small.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Step 1: calculate the model predictions

$\hat{y} = wx + b$ Model A: $w = 1, b = 1$

Input x	Substitution into $\hat{y} = wx + b$	Prediction $\hat{y}$
0	$1 \times 0 + 1$	1 cm
1	$1 \times 1 + 1$	2 cm
2	$1 \times 2 + 1$	3 cm

Mohammad Sabokrou | New Uzbekistan University

w controls the slope. b is the prediction when x = 0.

The same w and b produce a prediction for every example.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Step 2: subtract the observed targets

Prediction error $e_i = \hat{y}_i - y_i$

Example	Prediction $\hat{y}_i$	Target y_i	Error $\hat{y}_i - y_i$
1	1	1	$1 - 1 = 0$
2	2	3	$2 - 3 = -1$
3	3	5	$3 - 5 = -2$

Mohammad Sabokrou | New Uzbekistan University

A negative error means underprediction. A positive error means overprediction.

Example 3 underpredicts the length by 2 cm.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Step 3: square each prediction error

One-example loss: $L_i = e_i^2 = (\hat{y}_i - y_i)^2$

Error e_i	Multiplication	Squared error L_i
0	0×0	0
-1	$(-1) \times (-1)$	1
-2	$(-2) \times (-2)$	4

Mohammad Sabokrou | New Uzbekistan University

Sum of squared errors = 0 + 1 + 4 = 5 cm²

Doubling the error magnitude from 1 to 2 makes its square four times larger.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Step 4: average the squared errors

MSE means Mean Squared Error.

$$J = (1/N) \sum_i (\hat{y}_i - y_i)^2$$

N = 3 examples, so J = (0 + 1 + 4) / 3

Mohammad Sabokrou | New Uzbekistan University

$$J = 5/3 \approx 1.6667 \text{ cm}^2$$

L_i is one example's loss. J is the mean loss across examples.

A perfect fit on these examples has MSE = 0. Lower MSE means a closer fit here.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A second model has lower MSE

Model B: w = 1, b = 2, so ŷ = x + 2

x	Target y	Prediction ŷ	Error e	Square e²
0	1	2	+1	1
1	3	3	0	0
2	5	4	−1	1

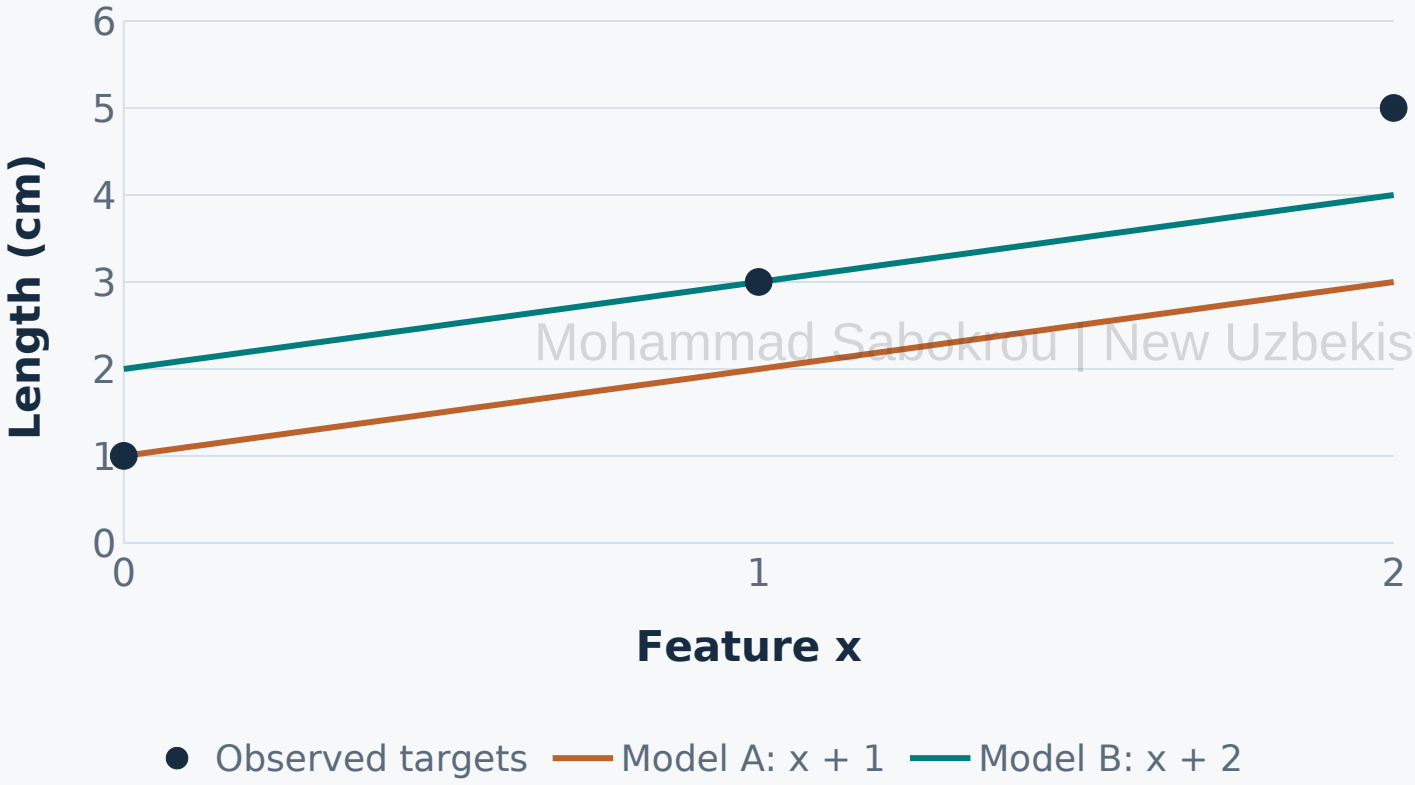
Mohammad Sabokrou | New Uzbekistan University

MSE = (1 + 0 + 1) / 3 = 2/3 ≈ 0.6667 cm²

Mean signed error is 0, yet two predictions are wrong. Squaring avoids cancellation.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The graph and the loss tell the same story



Mohammad Sabokrou | New Uzbekistan University

Model A

MSE = 1.6667

Model B

MSE = 0.6667

Compare vertical errors at the same x values.

For these toy points, w = 2 and b = 1 gives an exact fit and MSE = 0.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The target determines what a prediction means

Task	Meaning of target	Model output
Regression	A numerical measurement, such as length	A number $\hat{y}$
Classification	A category, such as digit 3	Probability for each class

Mohammad Sabokrou | New Uzbekistan University

Digit labels name categories. Predicting 8 for a 3 is not a measured error of 5 cm.

Next, loss will evaluate the probability assigned to the observed class.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Binary means two possible classes

Example task: does this image contain the digit 3?

Outcome	Encoded target y	Meaning
Digit 3	1	The positive class
Any other digit	0	The negative class

Mohammad Sabokrou | New Uzbekistan University

The numbers 0 and 1 are category codes. Positive and negative are names.

Binary describes the number of outcomes, not the number of input features.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A probability and a predicted label

$p = P(y = 1 \mid x)$

Read this as: the model’s probability of digit 3, given image features x.

If the model outputs p = 0.80	Probability
Digit 3, y = 1	p = 0.80
Another digit, y = 0	1 − p = 0.20

Mohammad Sabokrou | New Uzbekistan University

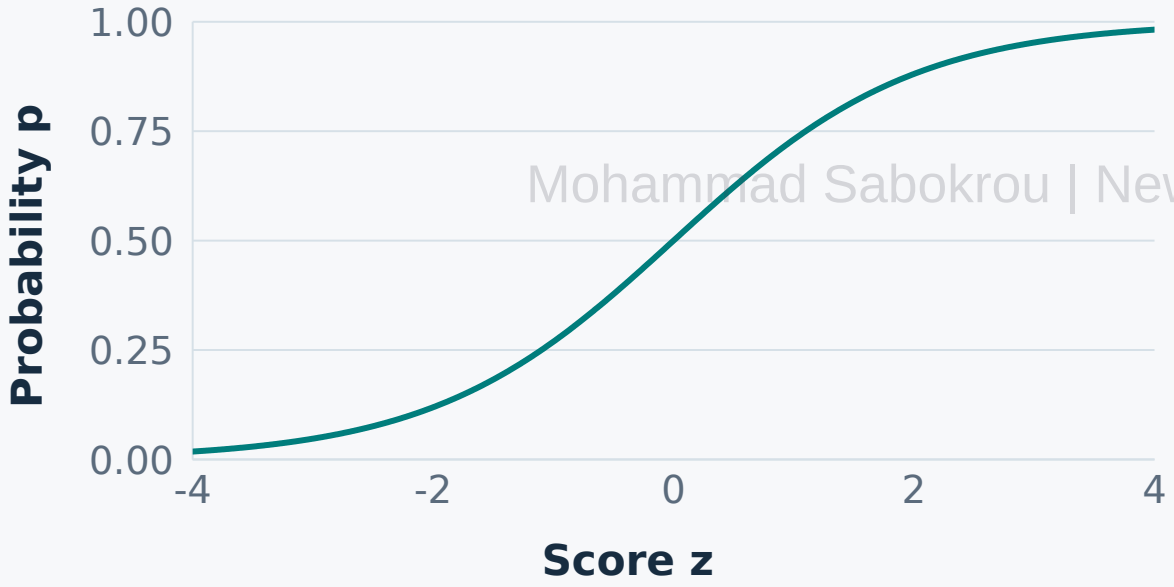
Example decision rule: predict 1 when p ≥ 0.5; otherwise predict 0.

The target y comes from the data. A high model probability can still be wrong.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

One score becomes a binary probability

$$z = w^T x + b \qquad p = \text{sigmoid}(z) = 1 / (1 + e^{-z})$$



Mohammad Sabokrou | New Uzbekistan University

Example: z = 2

$e^{-2} \approx 0.135335$

$p = 1 / 1.135335$

$p \approx 0.8808 = 88.08\%$

At z = 0, p = 0.5. Larger scores give larger probabilities for class 1.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The correct outcome selects the probability

Return to the p = 0.80 prediction. Now read the observed target.

Observed target	Correct outcome	Probability of that outcome
y = 1	Digit 3	p = 0.80
y = 0	Another digit	1 − p = 0.20

Mohammad Sabokrou | New Uzbekistan University

The model is more consistent with y = 1 than with y = 0 for this prediction.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A small true-class probability gives a large loss

Loss = −ln(probability of the correct outcome)

Correct-outcome probability q	ln(q)	Loss −ln(q)
0.80	−0.2231	0.2231
0.50	−0.6931	0.6931
0.20	−1.6094	1.6094
0.01	−4.6052	4.6052

Mohammad Sabokrou | New Uzbekistan University

ln is log base e. Since exp(−0.2231) ≈ 0.80, ln(0.80) ≈ −0.2231.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A positive example: y = 1

Observed target: digit 3. Model prediction: p = 0.80.

1. Select the correct-outcome probability: q = p = 0.80

2. Take its natural logarithm: ln(0.80) ≈ −0.2231

3. Change the sign: L = −(−0.2231) = 0.2231

Mohammad Sabokrou | New Uzbekistan University

When y = 1, L = −ln(p).

If p fell to 0.20, the loss would rise to −ln(0.20) ≈ 1.6094.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A negative example: y = 0

Observed target: another digit. The model still outputs p = 0.80.

1. Select the complement: q = 1 − p = 1 − 0.80 = 0.20

2. Take its natural logarithm: ln(0.20) ≈ −1.6094

3. Change the sign: L = −(−1.6094) = 1.6094

Mohammad Sabokrou | New Uzbekistan University

When y = 0, L = −ln(1 − p).

A zero target still contributes a loss. Here the model favours the wrong outcome.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

One formula contains both target cases

$$L = -[y \ln(p) + (1 - y) \ln(1 - p)]$$

Target	Substitution	Remaining loss
$y = 1$	$-[1 \times \ln(p) + 0 \times \ln(1 - p)]$	$-\ln(p)$
$y = 0$	$-[0 \times \ln(p) + 1 \times \ln(1 - p)]$	$-\ln(1 - p)$

Mohammad Sabokrou | New Uzbekistan University

Binary: two outcomes. Cross-entropy: score predicted probabilities against the target.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Mean binary cross-entropy for three examples

Example	Target y	Output p	True-class probability q	Loss −ln(q)
1	1	0.80	0.80	0.2231
2	0	0.30	1 − 0.30 = 0.70	0.3567
3	1	0.40	0.40	0.9163

Mohammad Sabokrou | New Uzbekistan University

Sum = 0.2231436 + 0.3566749 + 0.9162907 = 1.4961092

Mean BCE = 1.4961092 / 3 ≈ 0.4987

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Binary, multiclass and multiple labels

Task	Possible targets for one image	Output and loss
Binary	Digit 3 or another digit	One sigmoid probability Binary cross-entropy
Multiclass	Exactly one of A, B, C	Softmax probabilities Categorical cross-entropy
Multiple labels	A person and a bicycle can both be present	One sigmoid per label Binary cross-entropy per label

Mohammad Sabokrou | New Uzbekistan University

The next example has exactly one correct class among A, B and C.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A one-hot target identifies the correct class

There are K = 3 mutually exclusive classes. The observed class is B.

Class	A	B	C
One-hot target t	0	1	0

Mohammad Sabokrou | New Uzbekistan University

y = B t = [0, 1, 0]ᵀ

One-hot means exactly one 1. It marks the target, independently of the prediction.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Multiclass cross-entropy, step by step

$$L = -\sum_k t_k \ln(p_k) = -\ln(p_y)$$

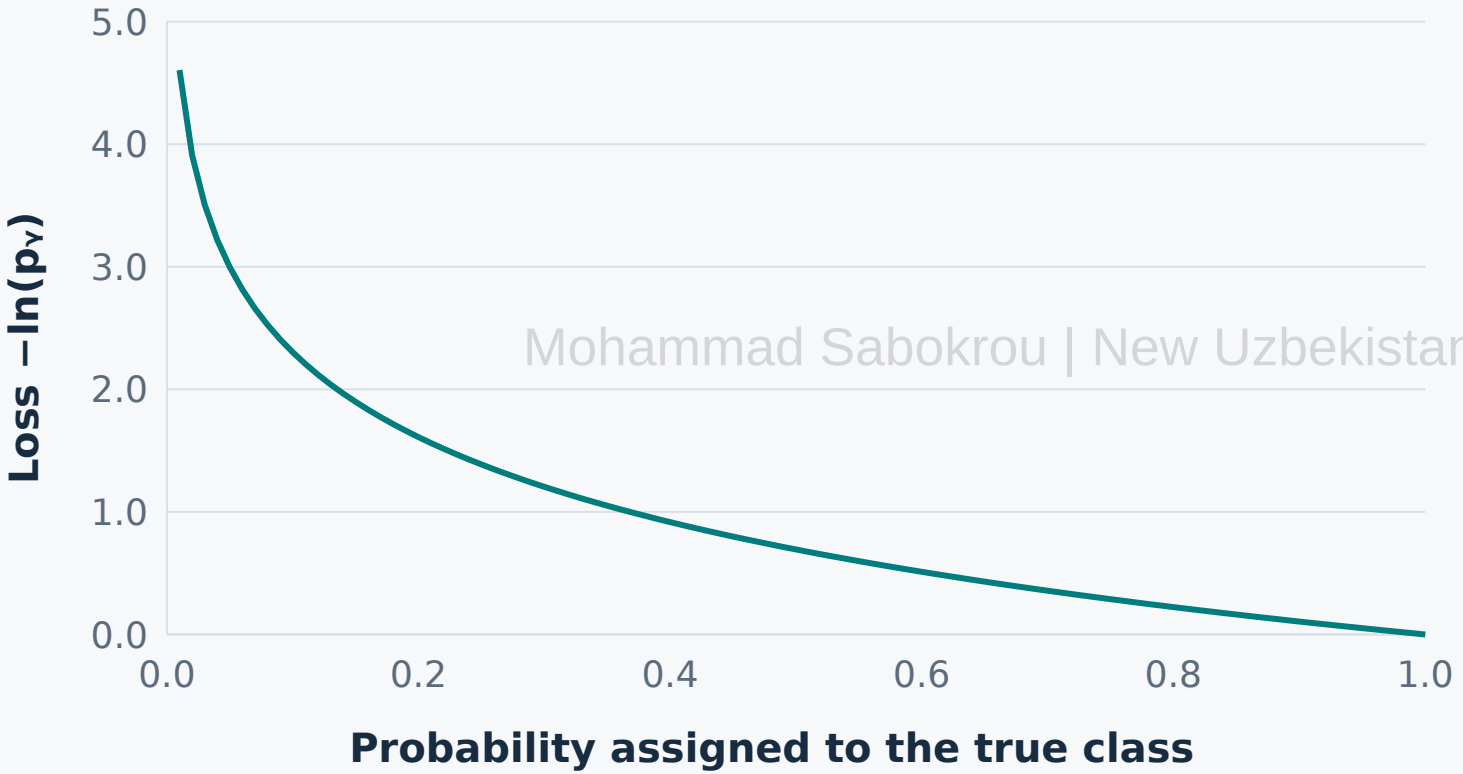
Class	Probability p_k	Target t_k	Term $-t_k \ln(p_k)$
A	0.665241	0	0
B	0.244728	1	$-\ln(0.244728\ldots)$
C	0.090031	0	0

Mohammad Sabokrou | New Uzbekistan University

$$L = 0 + 1.407606 + 0 \approx 1.4076$$

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Confident mistakes receive a large loss



Mohammad Sabokrou | New Uzbekistan University

p_Y	Loss
0.90	0.105
0.50	0.693
0.10	2.303
0.01	4.605

Loss uses natural-log units, not percent. The probability of the true class determines it.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Accuracy can stay fixed while loss improves

Probabilities [A, B, C]	Prediction	Correct?	Loss
[0.35, 0.40, 0.25]	B	Yes	0.9163
[0.20, 0.60, 0.20]	B	Yes	0.5108
[0.05, 0.90, 0.05]	B	Yes	0.1054

Mohammad Sabokrou | New Uzbekistan University

Correct label: B for every row

All three predictions are correct. Increasing p(B) still reduces cross-entropy.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Mean loss over the training examples

$$J(W, b) = (1/N) \sum_i L_i$$

N = examples averaged. Training changes W and b to reduce J.

Mohammad Sabokrou | New Uzbekistan University

At zero weights and biases	Digits classifier
Every logit	0
Probability for every class	1/10 = 0.1
Initial loss per example	$-\ln(0.1) = \ln(10) \approx 2.303$

Zero initialization works for this linear model; hidden neural layers need different care.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Three short calculations to check understanding

Task	Calculation	Answer
MSE: targets [2,4], predictions [1,6]	$[(1-2)^2 + (6-4)^2] / 2$	2.5
Binary: y = 0 and p = 0.10	$-\ln(1-0.10)$	0.1054
Multiclass: true B, p = [0.6,0.3,0.1]	$-\ln(0.30)$	1.2040

Mohammad Sabokrou | New Uzbekistan University

Try the calculation before reading the answer column.

For classification, select the true class’s probability before taking −ln.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

FIVE-MINUTE BREAK

05:00

Mohammad Sabokrou | New Uzbekistan University

How should a wrong prediction change the weights?

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

This gradient measures change in the loss

EARLIER WEEKS

$$\partial I / \partial u$$

Change intensity I
by moving across the image.

THIS WEEK

$$\partial L / \partial W_{kj}$$

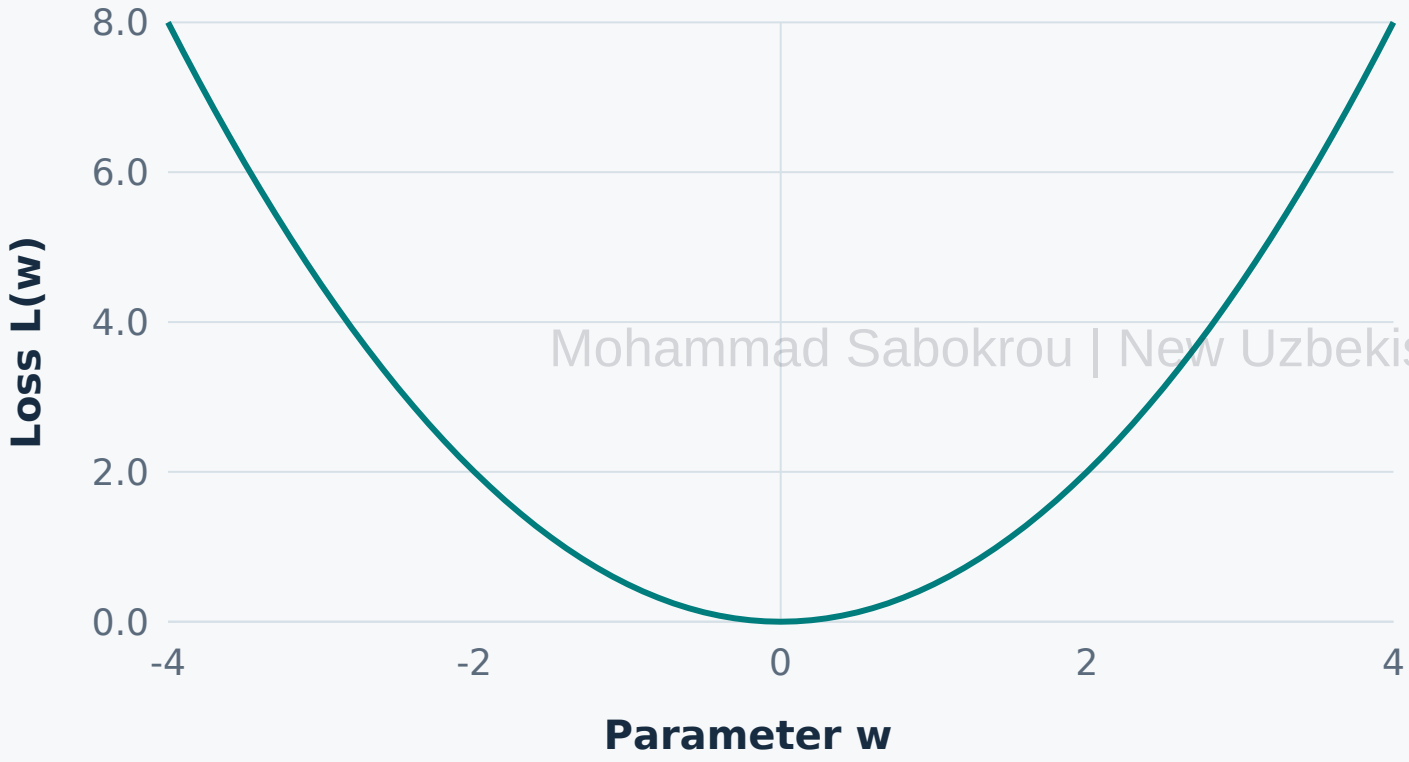
Change the loss L
by changing a weight.

Mohammad Sabokrou | New Uzbekistan University

Same calculus idea. Different quantities and variables.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A derivative gives a local direction



Mohammad Sabokrou | New Uzbekistan University

$$L(w) = \tfrac{1}{2}w^2$$

$$dL/dw = w$$

At $w = +2$
Positive slope
Move w left

At $w = -2$
Negative slope
Move w right

Here the factor ½ cancels the 2 from differentiating w^2. This is a separate toy loss.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A gradient collects partial derivatives

Term	Meaning
Derivative dL/dw	A local rate of change for a function of one variable
Partial derivative ∂L/∂w	Change w while keeping the other parameters fixed
Gradient ∇L	The collection of all parameter partial derivatives

Mohammad Sabokrou | New Uzbekistan University

Local estimate: ΔL ≈ (∂L/∂w) × Δw

If ∂L/∂w = −2 and Δw = +0.01, then ΔL ≈ −0.02.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The gradient-descent update

$$\theta' = \theta - \eta \nabla L(\theta)$$

Term	Meaning
θ, θ'	Parameters W and b, before and after the step
$\nabla L(\theta)$	How the loss changes with each parameter
η	Learning rate: how far to step

The minus sign moves against the gradient. A step that is too large can increase loss.

Mohammad Sabokrou | New Uzbekistan University

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The chain rule with squared error

One point from Model A: x = 2, y = 5, w = 1, b = 1.

$\hat{y} = 1 \times 2 + 1 = 3$ $e = 3 - 5 = -2$ $L = (-2)^2 = 4$

Link in the calculation	Local derivative	Value here
Loss $L = e^2$	$\partial L/\partial e = 2e$	-4
Error $e = \hat{y} - y$	$\partial e/\partial \hat{y} = 1$	1
Prediction $\hat{y} = wx + b$	$\partial \hat{y}/\partial w = x$	2

Mohammad Sabokrou | New Uzbekistan University

$\partial L/\partial w = (-4) \times 1 \times 2 = -8$

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The chain rule links loss to weights

Forward calculation	Backward question
$z = Wx + b$	How does a weight change its class score?
$p = \text{softmax}(z)$	How does a score change the distribution?
$L = -\ln(p_y)$	How does the true-class probability affect loss?

Mohammad Sabokrou | New Uzbekistan University

$$\partial L/\partial W_{kj} = (\partial L/\partial z_k) \times x_j$$

The input x and target y stay fixed while we differentiate the model parameters.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Changing one logit changes every probability

Quantity	Original scores	Increase only B’s score
Logits [A,B,C]	[2,1,0]	[2,2,0]
Probabilities [A,B,C]	[0.6652,0.2447,0.0900]	[0.4683,0.4683,0.0634]

Mohammad Sabokrou | New Uzbekistan University

B’s numerator grows, and the denominator shared by every class also grows.

The compact gradient p − t already includes these effects on all classes.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

WORKED EXAMPLE

The logit gradient is p − t

$$\partial L/\partial z_k = p_k - t_k$$

For one example with softmax + categorical cross-entropy + a one-hot target.

Class	Probability p	Target t	Gradient p − t
A	0.6652	0	+0.6652
B	0.2447	1	−0.7553
C	0.0900	0	+0.0900

Mohammad Sabokrou | New Uzbekistan University

B’s gradient is negative. Increasing zB lowers this example’s loss locally.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

WORKED EXAMPLE

One weight update, with every number visible

CLASS B, FEATURE 1: W[B,1] IS MATHEMATICAL NOTATION

$$\partial L/\partial W[B,1] = (p(B) - 1) \times x_1$$

$$= (-0.7552715\ldots) \times 2 \approx -1.510543\ldots$$

Mohammad Sabokrou | New Uzbekistan University

$$W'[B,1] \approx 0 - 0.1 \times (-1.510543)$$

$$\approx \mathbf{0.1511}$$

In NumPy with classes A=0, B=1, C=2, this entry is W[1,0].

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

WORKED EXAMPLE

The same rule updates all weights and biases

$$W' = W - \eta(p - t)x^T$$

Compute every gradient from the same original W and b, then update all entries.

Class	Old weights	New weights	Old bias	New bias
A	[1, 0]	[0.8670, −0.0665]	0	−0.0665
B	[0, 1]	[0.1511, 1.0755]	0	0.0755
C	[−1, 1]	[−1.0180, 0.9910]	1	0.9910

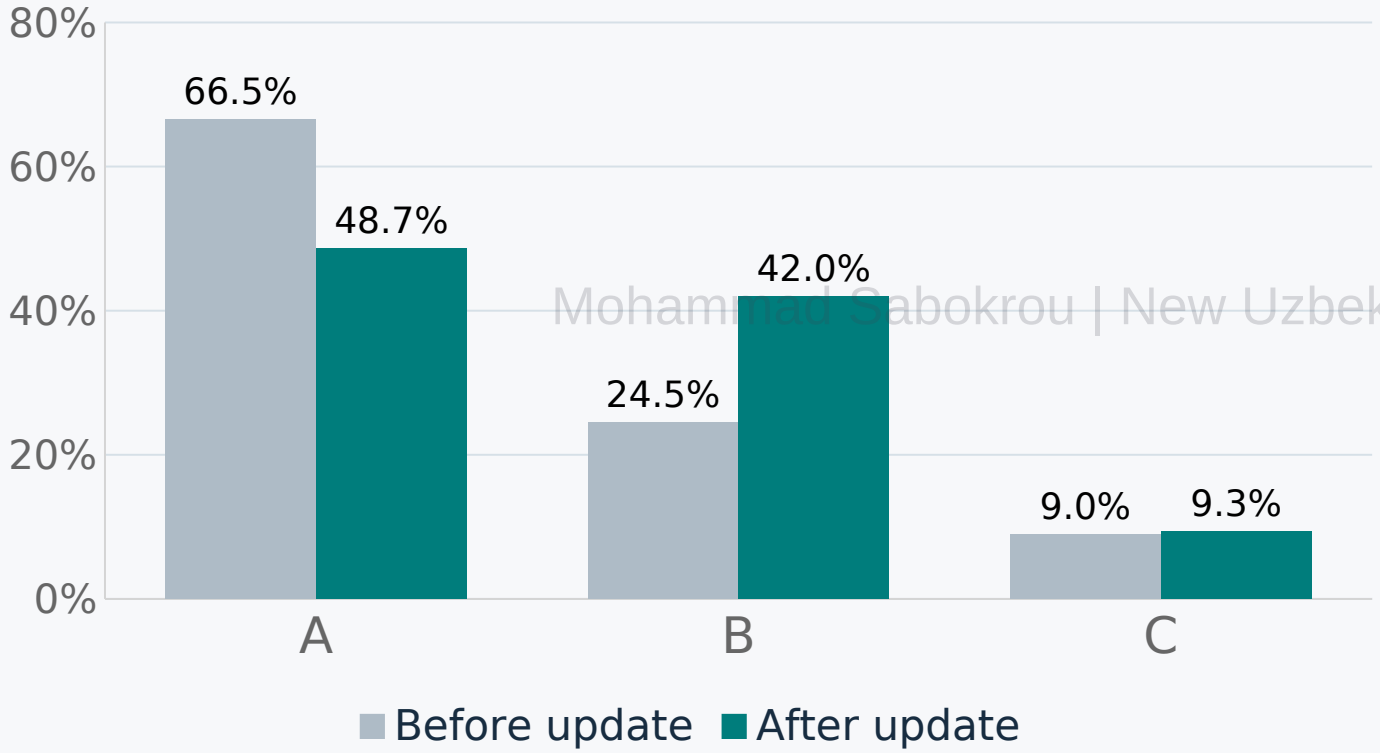
Mohammad Sabokrou | New Uzbekistan University

$b' = b - \eta(p - t)$ **Learning rate $\eta = 0.1$**

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

WORKED EXAMPLE

Loss can improve before accuracy changes



Mohammad Sabokrou | New Uzbekistan University

TRUE-CLASS PROBABILITY

24.47% → 42.00%

CROSS-ENTROPY

1.4076 → 0.8674

Prediction stays A.

The probabilities above use the complete W and b update on the preceding slide.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The mean gradient balances examples

Toy model $\hat{y} = w$, with targets 0 and 2. Start at $w = 0$.

Target y	Loss $(w - y)^2$ at w=0	Gradient $2(w - y)$
0	0	0
2	4	−4

Mohammad Sabokrou | New Uzbekistan University

Mean gradient = (0 − 4) / 2 = −2

$\eta = 0.25$: $w' = 0 - 0.25 \times (-2) = 0.5$

New losses: 0.25 and 2.25. Mean falls from 2 to 1.25; one example gets worse.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Gradients for a batch of examples

$$G = (P - T) / B$$

P: probabilities. T: one-hot targets. dW and db: parameter gradients.

$$dW = G^T X$$

$$db_k = \sum_i G_{ik}$$

Mohammad Sabokrou | New Uzbekistan University

X	P, T, G	W and dW
B × D	B × K	K × D

Here B is a count, not the earlier class named B. Divide by batch size once.

B = current batch size. Rows are examples; columns of P and T are classes.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The training loop

01 Initialize the model parameters W and b.

02 Compute logits and probabilities on training images.

03 Measure mean cross-entropy using the labels.

Mohammad Sabokrou | New Uzbekistan University

04 Compute all gradients from the current W and b.

05 Update every parameter using its own gradient.

06 Recompute after each update. Evaluate on validation images.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Examples, batches, updates and epochs

Term	Meaning
Example	One input together with its target
Batch	Examples used together to calculate one mean gradient
Update / iteration	One parameter step, followed by a new forward calculation
Epoch	One complete pass through the training examples

Mohammad Sabokrou | New Uzbekistan University

With 1,000 examples and batches of 100: 10 updates make one epoch.

The digits experiment uses the full training set in each update.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The learning rate controls the step size

$$L(w) = \tfrac{1}{2}w^2 \qquad w_0 = 4$$

$$w_1 = 4 - \eta \times 4$$

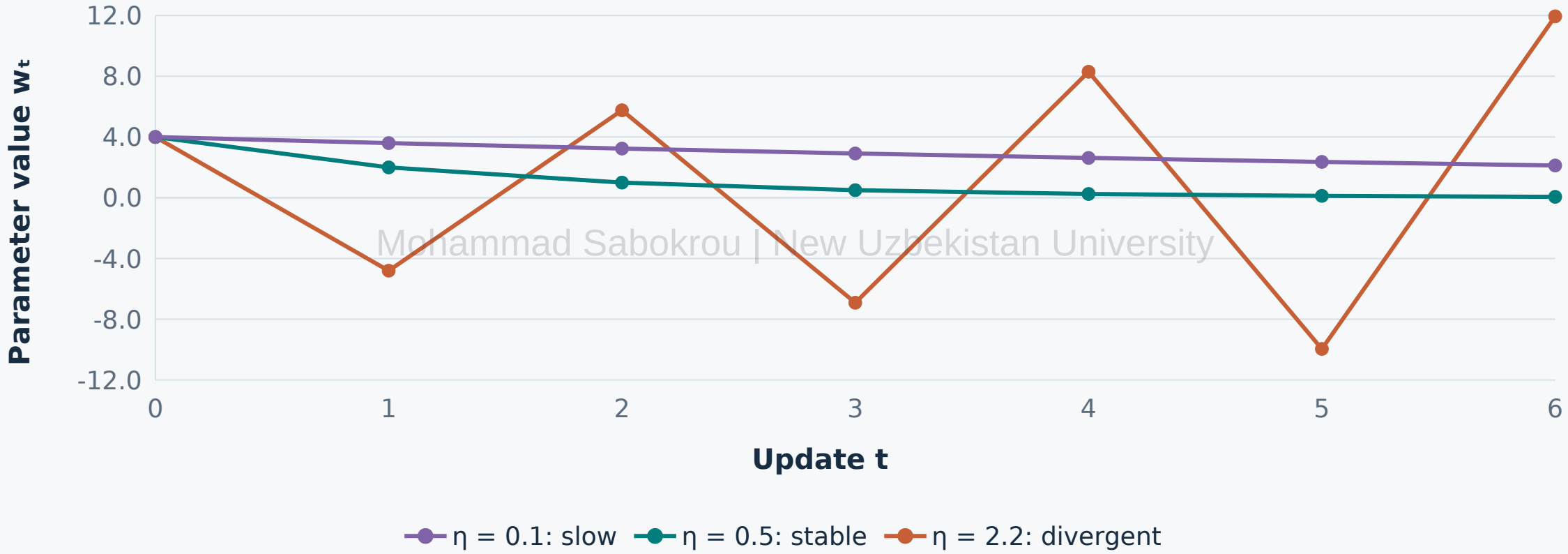
Mohammad Sabokrou | New Uzbekistan University

Learning rate η	Next parameter w_1	Next loss $L(w_1)$
0.1	3.6	6.48
0.5	2.0	2.00
2.2	−4.8	11.52

Initial loss is 8. This example isolates the effect of the step size.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Slow, stable and divergent updates



$w_t = 4(1 - \eta)^t$ **Divergence means the magnitude grows.**

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Oscillation can still converge

LEARNING RATE η = 1.5

Update t	0	1	2	3	4
Parameter w_t	4	−2	1	−0.5	0.25

Mohammad Sabokrou | New Uzbekistan University

The sign alternates. The magnitude shrinks.

For $L(w) = \frac{1}{2}w^2$: convergence when $0 < \eta < 2$

The threshold depends on curvature and scaling. It is not a universal bound of 2.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A controlled experiment on real digit images

Training	Validation	Test
1,077 images	360 images	360 images
Compute gradients	Choose learning rate	Final evaluation

Mohammad Sabokrou | New Uzbekistan University

Same split, seed 42, and pixels divided by 16.

Zero initialization. No regularization. 500 full-batch updates.

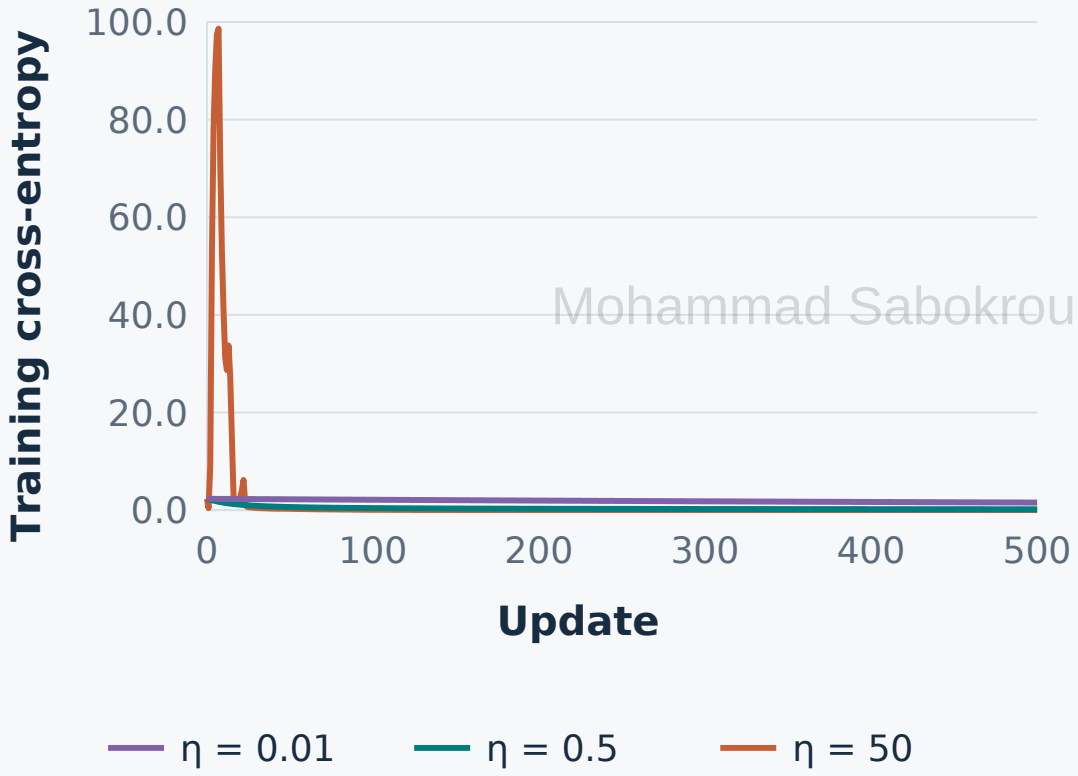
Candidate learning rates: 0.01, 0.5, 50.

Stratified image-level split of 1,797 sklearn digits. Not a test on independent writers.

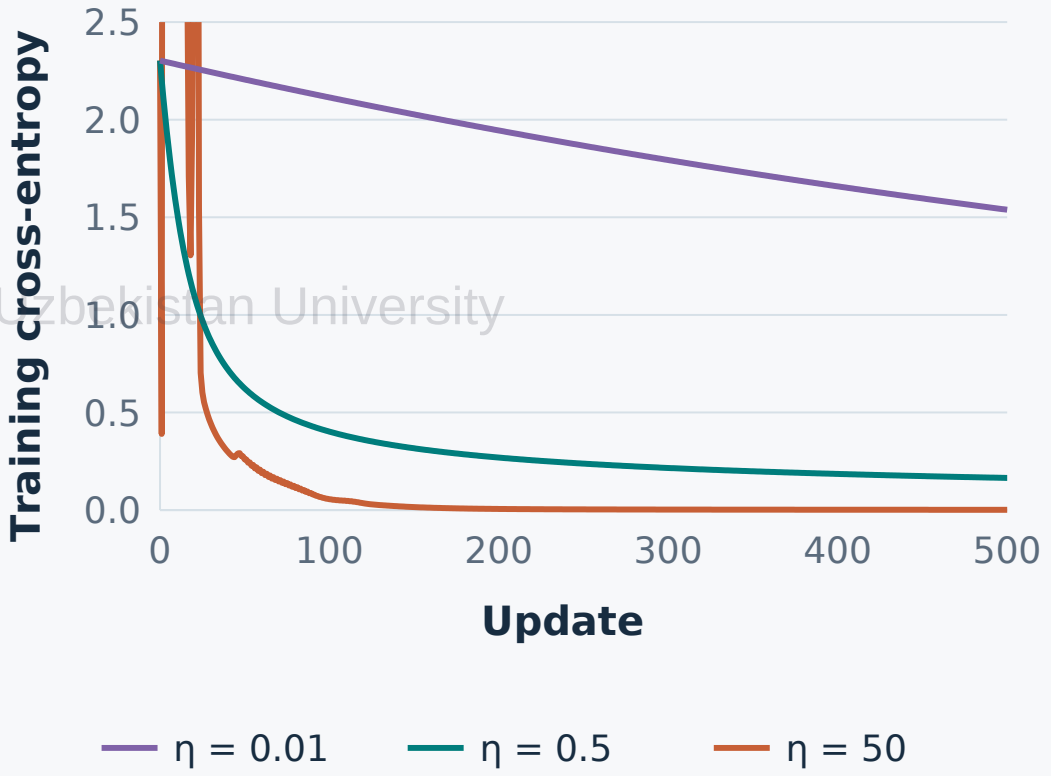
Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Learning rates on real digit images

FULL LOSS SCALE



CLOSE-UP: LOSSES FROM 0 TO 2.5



Mohammad Sabokrou | New Uzbekistan University

η = 50 spikes to 98.63 at update 7, then recovers. All losses stay finite.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Validation chooses the learning rate

Learning rate	Train loss	Train accuracy	Validation loss	Validation accuracy
0.01	1.5387	89.97%	1.5495	88.61%
0.5	0.1647	97.12%	0.1909	95.56%
50	0.0014	100.00%	0.4838	94.17%

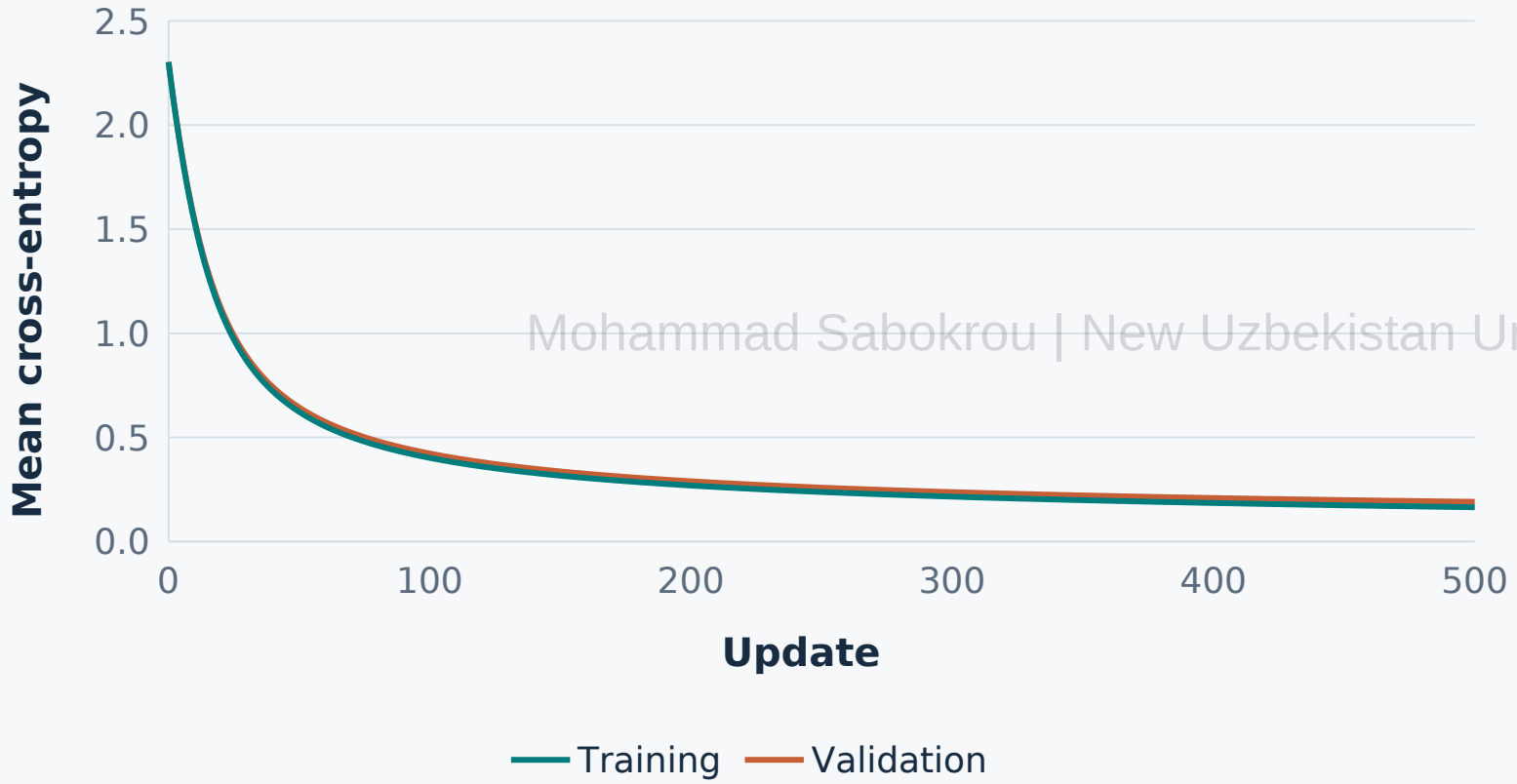
Mohammad Sabokrou | New Uzbekistan University

Selected $\eta = 0.5$

Best among these three rates: lowest validation loss after 500 updates.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Training and validation loss



Mohammad Sabokrou | New Uzbekistan University

AT UPDATE 500

Training
0.1647

Validation
0.1909

Validation images support model selection without contributing gradients.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Interpreting training and validation curves

Observed pattern	What to investigate
Both losses remain high	More optimization, better inputs, or a more suitable model
Training loss falls while validation loss rises	Possible overfitting: compare checkpoints using validation
Loss is erratic or becomes nonfinite	Learning rate, numerical stability, and input data

Mohammad Sabokrou | New Uzbekistan University

A curve is evidence to investigate. One pattern alone does not identify its cause.

Keep the test set for final evaluation of the selected procedure.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Test result: 339 correct out of 360

94.17%

339 correct
21 mistakes

Test cross-entropy
0.2198

ROWS: TRUE LABEL / COLUMNS: PREDICTION

y	0	1	2	3	4	5	6	7	8	9
0	34	0	0	0	1	0	0	0	0	0
1	0	31	0	0	1	1	2	0	0	1
2	0	2	33	0	0	0	0	1	0	0
3	0	0	0	36	0	0	0	0	0	1
4	0	0	0	0	35	0	0	1	1	0
5	0	0	0	0	0	34	0	0	0	2
6	0	0	0	0	0	0	35	0	1	0
7	0	0	0	0	0	0	0	36	0	0
8	0	4	0	0	0	0	0	0	31	0
9	0	1	0	0	0	0	0	1	0	34

Mohammad Sabokrou | New Uzbekistan University

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Reading the digit 1 results

Rows are true labels. Columns are predictions. Diagonal cells are correct.

Quantity from the matrix	Calculation	Meaning
True digit 1: 36 images	31 correct / 36 true 1s	Recall = 86.1%
Predicted digit 1: 38 images	31 correct / 38 predicted 1s	Precision = 81.6%

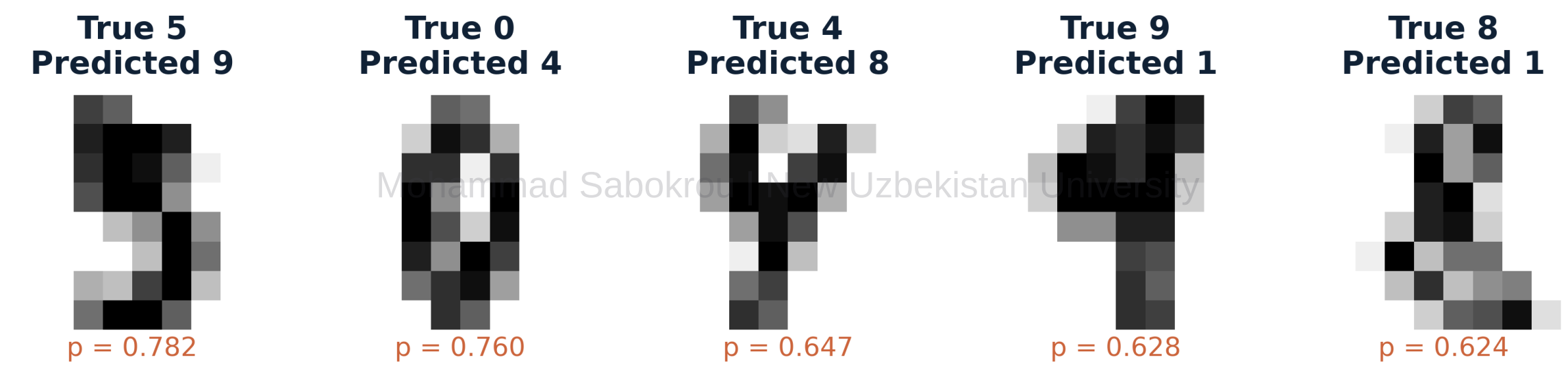
Mohammad Sabokrou | New Uzbekistan University

The four true 8s predicted as 1 appear in row 8, column 1.

An off-diagonal cell identifies a specific error type. These entries are counts.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Confident mistakes on real test images



Mohammad Sabokrou | New Uzbekistan University

p is the model’s probability for the predicted class.

Examples selected from the 21 mistakes for inspection. They do not show the overall error frequency.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

What the learned weights show



Mohammad Sabokrou | New Uzbekistan University

Each map is a learned weight vector, reshaped to 8 × 8.

A positive centered weight is above the class average at that pixel. The next slide explains.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

What a centered weight map means

At pixel j, subtract the average weight across the K classes.

Centered weight = $w_{kj} - (1/K) \sum_a w_{aj}$

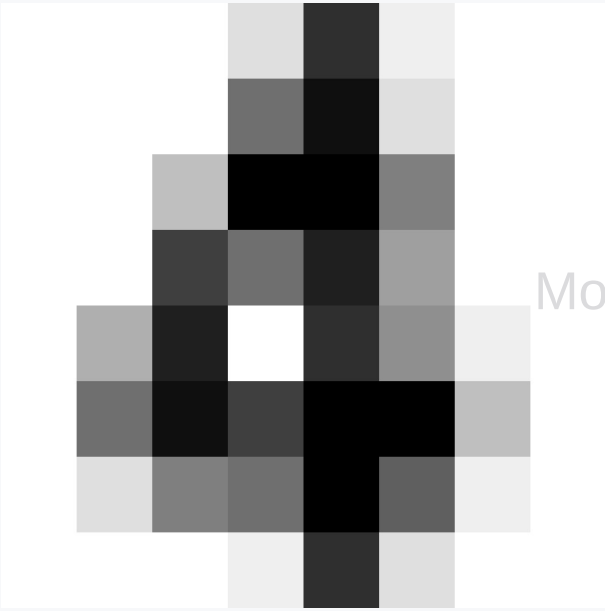
Feature of the map	Interpretation
Positive centered coefficient	This class’s coefficient exceeds the class average at that pixel
Contribution on a particular image	Weight × input value; the score also includes the bias
Equal shift across every class	Adds the same offset to all logits and preserves softmax

A weight map alone does not establish causal importance or guarantee a probability change.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

A one-pixel shift can change the prediction

Original



Predicts 4

p = 81.74%

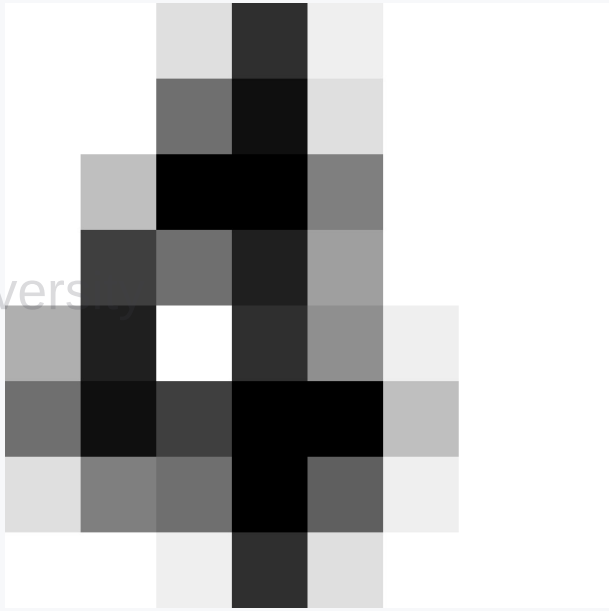
1 pixel right



Predicts 1

p = 53.32%

1 pixel left



Predicts 4

p = 85.72%

Mohammad Sabokrou | New Uzbekistan University

Same true digit 4. Zero-filled shifts, no ink cropped. One example, not an aggregate robustness test.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

NEXT LECTURE

Next topic: convolutional neural networks

Linear classifier	Convolutional network
Fixed pixel features	Learned local filters and feature maps
Learn W and b over those features	Learn the representation and classifier
Absolute pixel positions matter	Shared filters reuse local patterns

Mohammad Sabokrou | New Uzbekistan University

The final class scores can still use softmax and cross-entropy.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

The complete multiclass learning step

Calculation	Purpose
$z = Wx + b$	Class scores
$p = \mathrm{softmax}(z)$	Model class probabilities
$L = -\ln(p_Y)$	Loss for the true class
$g = p - t$	Gradient with respect to scores
$dW = gx^T, \quad db = g$	Gradients of the parameters
$W' = W - \eta dW$	A training update, likewise for b

Mohammad Sabokrou | New Uzbekistan University

One example, one-hot target, no regularization. Average gradients for batch training.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Practice and implementation

HAND CALCULATION, THEN NUMPY

Compute MSE and both binary cross-entropy target cases.

Reproduce the A/B/C softmax, loss and full weight update.

Mohammad Sabokrou | New Uzbekistan University

Implement stable loss, check a gradient, and compare rates.

TEXTBOOK READING

Szeliski, 2nd edition: §5.1.3, §5.3.4, §5.3.5, §5.3.6

Logistic regression, loss functions, backpropagation, training and optimization.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Reading for the mathematical foundations

Szeliski (2022), second edition

Computer Vision: Algorithms and Applications

https://szeliski.org/Book/

Stanford CS231n

Linear Classification

Mohammad Sabokrou | New Uzbekistan University

https://cs231n.github.io/linear-classify/

Stanford CS231n

Optimization: Stochastic Gradient Descent

https://cs231n.github.io/optimization-1/

Stanford CS231n

A Neural Network Case Study

https://cs231n.github.io/neural-networks-case-study/

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Data credits and experiment details

DIGIT IMAGES

E. Alpaydin and C. Kaynak (1998)
Optical Recognition of Handwritten Digits
UCI Machine Learning Repository, CC BY 4.0

Mohammad Sabokrou | New Uzbekistan University

https://doi.org/10.24432/C50P49

THIS CLASSROOM EXPERIMENT

1,797-image sklearn subset. Stratified image split, seed 42.
Pixels / 16. Zero initialization. No regularization.
500 full-batch updates. Learning rate selected on validation loss.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Reading for regression and cross-entropy

Source	Relevant explanation
Google Machine Learning Crash Course	Linear regression: loss Logistic regression: sigmoid function
PyTorch documentation	BCEWithLogitsLoss CrossEntropyLoss

Mohammad Sabokrou | New Uzbekistan University

developers.google.com/machine-learning/crash-course

docs.pytorch.org/docs/stable/nn.html#loss-functions

The numerical examples in these slides are synthetic teaching examples.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Appendix: stable forward pass in NumPy

```
import numpy as np

def forward(X, y, W, b):
    z = X @ W.T + b
    s = z - z.max(axis=1, keepdims=True)
    logp = s - np.log(
        np.exp(s).sum(axis=1, keepdims=True)
    )
    loss = -logp[np.arange(len(y)), y].mean()
    p = np.exp(logp)
    return loss, p
```

Mohammad Sabokrou | New Uzbekistan University

X: B × D W: K × D b: K y: B integer class labels

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Appendix: batch gradients and updates

```
loss, p = forward(X, y, W, b)

g = p.copy()
g[np.arange(len(y)), y] -= 1
g /= len(y)             # mean loss: divide once

dW = g.T @ X            # shape: K by D
db = g.sum(axis=0)  # shape: K

W -= learning_rate * dW
b -= learning_rate * db
```

Mohammad Sabokrou | New Uzbekistan University

Use argmax for evaluation. The loss supplies the training gradient.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Appendix: a numerical gradient check

$$\partial L/\partial w \approx [L(w + h) - L(w - h)] / (2h)$$

Check for W[B,1]	Value
Perturbation h	0.000001
Analytic gradient	−1.510543058
Numerical gradient	−1.510543058

Perturb one parameter, keep the others fixed, and restore it afterwards.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

OPTIONAL DERIVATION

Appendix: why the logit gradient is p − t

$$L = -z_y + \ln\left(\sum_j \exp(z_j)\right)$$

Term	Derivative with respect to z_k
$-z_y$	$-t_k$
$\ln(\sum_j \exp(z_j))$	$\exp(z_k) / \sum_j \exp(z_j) = p_k$

$$\partial L/\partial z_k = p_k - t_k$$

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

OPTIONAL CONNECTION

Appendix: two classes give the sigmoid model

$$p_1 = \exp(z_1) / [\exp(z_0) + \exp(z_1)]$$

$$= 1 / [1 + \exp(-(z_1 - z_0))]$$

$$\mathbf{= \text{sigmoid}(z_1 - z_0)}$$

The score difference gives the familiar binary logistic-regression model.

Mohammad Sabokrou | New Uzbekistan University

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

Loss functions expect different inputs

PyTorch loss	Pass to the loss	Targets
MSELoss	Numerical predictions ŷ	Numerical values, matching shape
BCEWithLogitsLoss	Raw binary logits z	Float labels 0 or 1, matching shape
CrossEntropyLoss	K raw class logits per example	One integer class index per example

Mohammad Sabokrou | New Uzbekistan University

BCEWithLogitsLoss includes sigmoid. CrossEntropyLoss includes log-softmax.

Use probabilities for interpretation. Pass raw logits to these two classification losses.

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

OPTIONAL DISCUSSION

Appendix: three explanations to practise

01 Why do [2, 1, 0] and [1002, 1001, 1000] give the same softmax?

02 How can a weight update lower loss while the predicted class stays wrong?

03 Why can one oscillating learning curve converge while another diverges?

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

QUICK REFERENCE

Appendix: vocabulary and notation

Term	Meaning
Feature x_j	One numerical input coordinate
Parameters W, b	Values that training updates
Logit z_k	An unnormalized class score
Probability p_k	Model probability for a class, from sigmoid or softmax
Loss L	Penalty for a prediction; its definition depends on the task
Gradient ∇L	Partial derivatives with respect to parameters
Learning rate η	Scale of the parameter update
Epoch	One pass through the training examples

Mohammad Sabokrou | New Uzbekistan University

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026

QUICK REFERENCE

Loss formulas and the meaning of their symbols

Setting	Formula	Meaning
Numerical regression	$L = (\hat{y} - y)^2$	Squared prediction error
Binary classification	$L = -[y \ln(p) + (1-y) \ln(1-p)]$	p = probability of class 1
Multiclass classification	$L = -\sum_k t_k \ln(p_k) = -\ln(p_Y)$	p_Y = probability of the true class
Mean objective	$J = (1/B) \sum_i L_i$	Average over B examples

Mohammad Sabokrou | New Uzbekistan University

y = observed target $\hat{y}$ = numerical prediction t = one-hot target ln = natural log

Copyright Mohammad Sabokrou | New Uzbekistan University | 2026