

LECTURE 14

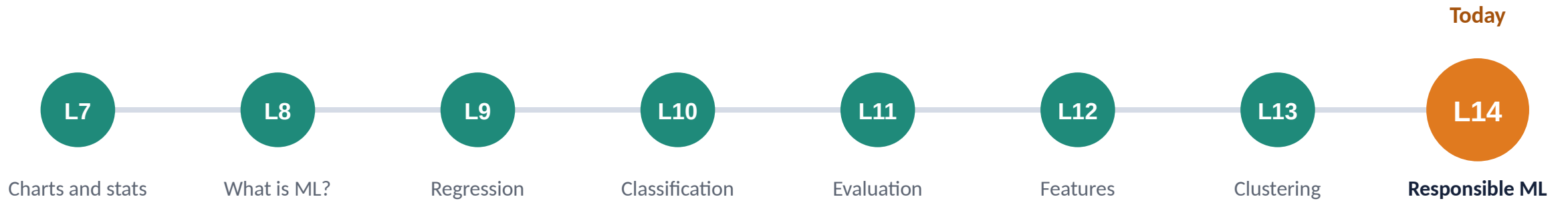
Responsible ML and what's next

Fairness, privacy, and a first look at neural networks

Today's key line of code

```
MLPClassifier(hidden_layer_sizes=(64,))
```

The road ahead: machine learning



Today you will

- 1 Explain how models can affect people unfairly
- 2 Check a model's performance per group
- 3 Protect privacy when collecting data
- 4 Compute one artificial neuron by hand
- 5 Train a small neural network and know where to go next

Models make decisions about people

1

Who gets a loan

A credit model trained on past approvals can repeat past unfairness.

2

Who gets seen first

A health model that works worse for one group puts that group at risk.

3

Who gets filtered out

A spam or CV filter can hide real messages or good candidates without anyone noticing.

Where unfairness comes from

1

The data

Some groups are under-represented, so the model learns less about them. Recall our 2011 UK spam data.

2

The labels

Labels written by people can carry their past decisions and biases.

3

The measurement

A feature may mean different things for different groups, or be missing more often for some.

A simple, essential check: performance per group

`cross_val_predict` gives an honest prediction for every row • Then compare accuracy group by group • Large gaps need explaining before any use • `students.csv` is invented: this shows the method only

```
Code
import pandas as pd
from sklearn.compose import make_column_transformer
from sklearn.impute import SimpleImputer
from sklearn.model_selection import cross_val_predict
from sklearn.neighbors import KNeighborsClassifier
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import StandardScaler
st = pd.read_csv("students.csv")
num = ["hours", "attendance", "prior_gpa"]
prep = make_column_transformer(
    (make_pipeline(SimpleImputer(strategy="median"),
                  StandardScaler()), num),
    (OneHotEncoder(), ["dept"]))
model = make_pipeline(prepare, KNeighborsClassifier())
X = st[num + ["dept"]]
st["pred"] = cross_val_predict(model, X, st["passed"], cv=5)
st["correct"] = st["pred"] == st["passed"]
print(st.groupby("dept")["correct"].mean().round(3))
```

```
Output
dept
CSE      0.911
EEE      0.900
Math     0.867
Name: correct, dtype: float64
```

Privacy and consent

1

Collect only what you need

If a name or phone number does not help the question, do not collect it.

2

Remove identifiers

Delete names, numbers and addresses before analysis. Your SMS homework already did this.

3

Ask permission

People should know and agree when their data is used. Store it safely and delete it when done.

A checklist before trusting any model

Check	Question to ask
Baseline	Does it clearly beat a simple guess?
Honest test	Was the test set used only once, at the end?
Right metric	Which mistake is costly, and is that metric reported?
Groups	Does it work similarly for different groups?
Data match	Does new data look like the training data (time, place, language)?
Privacy	Was the data collected with permission and anonymised?

A first look at neural networks

1

A neuron

Multiplies each input by a weight, adds them up with a bias, and passes the result through a simple function.

2

Layers

Many neurons side by side form a layer; layers stacked together form a network.

3

Learning

Training adjusts the weights a little at a time to reduce the error, like the line search in Lecture 9.

By hand: one neuron

Inputs: hours 6, attendance 0.9, GPA 3.2. Weights 0.5, 2.0, 1.0, bias -5. Output 1 if the total is above 0.

Code

```
# total = 6×0.5 + 0.9×2.0  
#       + 3.2×1.0 - 5  
#       = 3.0 + 1.8 + 3.2 - 5  
#       = 3.0  
# 3.0 > 0, so output 1
```

input	value	weight	value × weight
hours	6	0.5	3.0
attendance	0.9	2.0	1.8
GPA	3.2	1.0	3.2
bias			-5
total			3.0
output			1

Click to reveal the next row. Predict it first!

Watch it: one neuron computing its output

WATCH FOR

- Each input is multiplied by its weight
- The products flow into the neuron and are added
- The bias is added: $8.0 - 5 = 3.0$
- $3.0 > 0$, so the output is 1

NEURAL NETWORKS, STEP BY STEP

One neuron computing its output

Inputs travel along weighted connections into the neuron.
It adds them up, adds the bias, and decides.

Click the video to play, click again to pause. Silent: talk the class through it.

A small neural network on the digits

MLPClassifier: a multi-layer perceptron · One hidden layer of 64 neurons · Same fit / score pattern as every model · Compare with k-NN from Practice 3 on the same split

Code

```
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
from sklearn.neural_network import MLPClassifier
X, y = load_digits(return_X_y=True)
X_tr, X_te, y_tr, y_te = train_test_split(
    X, y, test_size=0.25, random_state=0, stratify=y)
knn = KNeighborsClassifier(n_neighbors=1)
net = MLPClassifier(hidden_layer_sizes=(64,),
                    max_iter=1000, random_state=0)
for name, m in [("kNN k=1", knn), ("neural net", net)]:
    pipe = make_pipeline(StandardScaler(), m)
    pipe.fit(X_tr, y_tr)
    print(name, round(pipe.score(X_te, y_te), 3))
```

Output

```
kNN k=1 0.967
neural net 0.973
```

Watch it: a small network learning

WATCH FOR

- The border starts almost straight
- Each round nudges every weight a little
- The error falls fast, then flattens
- After 200 rounds: 97% on these points

NEURAL NETWORKS, STEP BY STEP

A small network learning

Each training round, the network nudges its weights to reduce the error.
Watch the border bend to fit the data.

Click the video to play, click again to pause. Silent: talk the class through it.

Where to go next

1

Deep learning

Larger networks for images, speech and text. Libraries: PyTorch, TensorFlow.

2

Language and speech

Bangla text and speech processing is an open area with real local impact.

3

Practice

Your project. Then public datasets and competitions, always with honest evaluation.

Interactive lab: Responsible ML Lab

In the lab you can

1 Check every group

Slide 5 per department; a rare group gets worse results.

2 Privacy

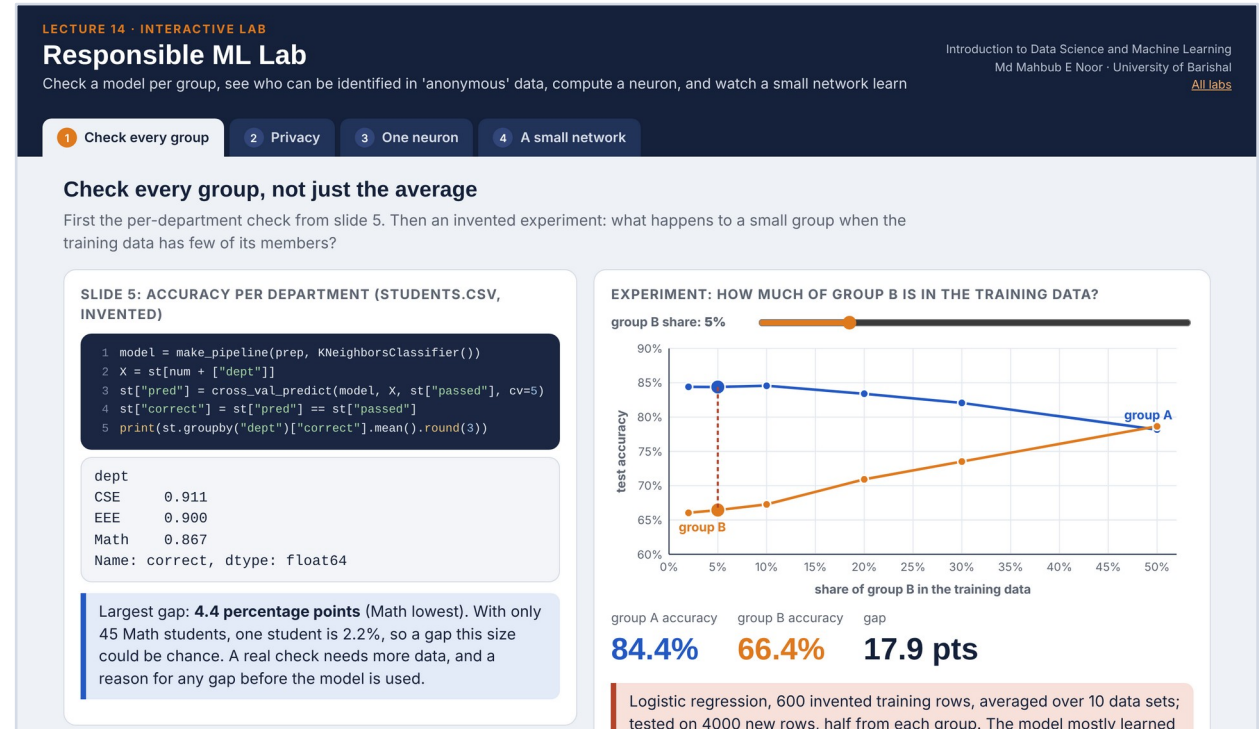
Remove names: how many people are still unique?

3 One neuron

Slide 9 with sliders: weights, bias, step or sigmoid.

4 A small network

Watch the border bend epoch by epoch; 1 to 16 neurons.



Open the lab ►

Works offline in any browser
Labs_open_in_browser/
L14_Responsible_ML_Lab.html

Today's toolkit

cross_val_predict	Honest prediction for every row
groupby(group) ["correct"].mean()	Accuracy per group
anonymise	Remove names, numbers, addresses

neuron	Weighted sum + bias, then a function
MLPClassifier(...)	A small neural network
checklist	Baseline, honest test, metric, groups, data match, privacy

Homework 14

Due before the project kickoff • [date]

1 By hand

For the neuron on the slide, find a student (hours, attendance, GPA) for whom the output is 0. Show the total.

2 Group check

Repeat the per-group check with a decision tree instead of k-NN. Do the groups differ more or less?

3 Network size

Train MLPClassifier on the digits with hidden layers of 8, 32 and 128 neurons. Report test accuracy for each.

4 Responsible use

Pick one real ML system you use. Write two ways it could treat some people unfairly.

5 Project idea

Write one paragraph describing a project you would like to do. Bring it to the kickoff.

Submit on [LMS / Google Classroom]: your notebook file HW14_<your ID>.ipynb, or a Colab link set to “Anyone with the link can view”.

Recap: Lecture 14 in 90 seconds

RECAP

Lecture 14 in 90 seconds

A short recap of today's lecture. Pause any time.

Click to play or pause. Watch it again at home to revise.