Intro to AI and Machine Learning

Syllabus and Outline

The MacLea Project
https://maclea.mit.edu/

Summary

This is a recommended outline for instructors teaching introductory artificial intelligence and machine learning classes. This document was designed around use of the MacLea educational tool.

Contents

Content Guide and Notes

This is for instructors to follow along material covered in the MacLea tutorials. It includes descriptions of the content along with suggested class questions to ask groups of students and instructions for classrooms to follow along. Links to the relevant tutorials are included for each section. The language and explanations in this section is written in such a way to be understandable for teaching on an elementary school level.

1 What is AI and machine learning?

AI, or artificial intelligence, refers to the ability for machines to solve problems, reason through logic, and represent knowledge. It gives them the ability to think and make decisions, kind of like how humans do.

Just like with humans, machines becomes more intelligent by learning more about the world. Machine learning is a special part of AI. It's the method that teaches computers how to learn from experience. So, while AI is the brain, machine learning is the tool that helps computers learn and get better at tasks over time.

Class Question: What are some examples of AI or machine learning? Can you think of some technology that you interact with that is learning from your interactions? (Some examples may include: smart search engines, recommendation systems, self-driving cars.)

Machine learning is very important in AI because it's what makes computers smart. It's like the brain's "thinking process." Instead of telling the computer exactly what to do for every little thing, we use machine learning to let it figure things out on its own by learning from examples.

Tutorial: Go to and read the "What is it?" tutorial page. [Direct Link]

Machine learning is called the most important part of AI because it's the secret sauce that makes computers intelligent. It allows them to learn and adapt, getting better and smarter as they encounter new information and tasks. Without machine learning, AI wouldn't be as powerful or as capable of making smart decisions on its own.

2 Motivation

AI can do cool things like creating art, playing games, recognizing objects, driving, and even handling tasks without being told exactly what to do. Knowing how to use AI gives you a powerful tool that can save you a ton of effort.

Let's take a look at a simple example!

Tutorial: Go to the "Reasons to Learn" tutorial page. [Direct Link]

Take a look at the five handwritten numbers on the page.

Tutorial: Write down in the blue boxes what the numbers above them are.

If it's just a few, like in the blue boxes, it's easy for you to label them.

Tutorial: Scroll down to the large set of images.

But what if you have a whopping 3750 of them (like in the big list of pictures)? That's a lot of work. If you had to label all those 3750 numbers on paper, it would be very hard and time-consuming!

Here's where AI and machine learning come to the rescue! With machine learning, you can teach the computer to recognize and label them quickly. It can do the annoying work for you!

So, learning AI and machine learning makes doing a lot of repeated tasks easy. And, it's not just for numbers; it can help with many different things which we'll learn about in MacLea!

Tutorial: Go to and read the "What We'll Do" tutorial page. [Direct Link]

In this class, we'll use the power of AI to teach our computer to tell us what's in different types of pictures, ranging from simple handwritten numbers to more special images. Rather than directly telling the computer what's in each picture, we'll show it many examples so that it learns on its own. This is what will let our computer understand whats in an image. This process not only makes our computer a helpful assistant in understanding images but also opens up exciting possibilities for understanding and working with many other types of things other than pictures!

3 Architecture

Machine learning architecture is the design that organizes the different parts of machine learning. The architecture guides the flow of information in the machine, ensuring that the AI learns from the data and can make accurate predictions. Essentially, the machine learning architecture provides the order needed to create intelligent systems that can learn from examples and generalize that learning to new, unseen situations.

Loading

Loading data is the first step in every machine learning architecture.

> **Tutorial:** Go to and read the "Loading" tutorial page. [Direct Link]

In many cases, this process is very difficult and time consuming. Data used in machine learning can come in various formats such as text, images, or audio. Handling these different types of data and converting them into a format suitable for machine learning models can be complex. In addition, machine learning often requires large datasets for effective training. Managing massive amounts of data efficiently, especially when dealing with big data, can be challenging due to memory constraints and processing limitations.

But, thankfully for us, it is just a single block!

> **Tutorial:** Drag and drop the "load" block into the workspace. Put this block after the "start" block.

Model

> **Tutorial:** Go to and read the "Model" tutorial page. [Direct Link]

In machine learning, a model is the brain that remembers patterns and relationships from what we teach it. It is what has the ability to make predictions on new, unseen things based on what it learned from when it was being taught. To put it simply, it is the brain for our AI.

> **Tutorial:** Drag and drop the "model" block into the workspace. Put this block after the "load" block.

After we teach our brain, we'll want to do something with it!

Note: A "brain" block will be temporarily inserted within the "model" block. We will look into actually designing the brain in later steps. We are just using this for now to see it in action.

Classification

Classification refers to the task of assigning categories or labels to pictures based on patterns that were previously learned. This is what we were doing manually on the "Reasons to Learn" tutorial page. This is the task we will focus on having our AI try to learn throughout these tutorials.

Tutorial: Go to and read the "Classify" tutorial page. [Direct Link]

The classify block acts as the final decision-making step, where the AI applies what is learned to label each picture. For example, in our case where the AI will be trained to recognize numbers, the classify block would determine whether the image contains a "three", "seven", or any other label.

Tutorial: Drag and drop the "classify" block into the workspace. Put this block after the "model" block.

Essentially, classification is when the AI labels the input, making it an important component in applications such as image recognition, text categorization, and more.

Tutorial: Go to and run the "Let's Try Running It!" tutorial page. [Direct Link]

Class Question: How is it doing? Is it good? Is it accurate?

The provided model on this tutorial page shouldn't be very accurate. In the next section, we'll see what actually goes on in the "brain" block.

4 Neural Networks

A neural network is a type of model inspired by the functioning of the human brain. It is designed to allow computers to learn and make decisions based on information.

Tutorial: Go to and read the "Creating the Brain" tutorial page. [Direct Link]

The term "neural network" comes from how it is formed behind the scenes. Neural networks are made up of neurons that are formed into layers. Each neuron will store the importance of a particular feature. As a whole, an entire layer of neurons will then be able to understand patterns and features in a picture.

This is how neural networks can generalize their learning to make predictions or classifications on new, unseen pictures in the future.

Construction of the neural network will be inside of the "model" block.

Flatten

Flatten is an operation that rearranges the structure of the pixels in pictures. When you flatten a picture, you're taking the 2-D grid of pixels (the rows and columns in a picture) and transforming it into a 1-D sequence. This means the neural network will process the image pixel by pixel in a line instead of trying to understand the entire image all at once. It's kind of like unrolling a scroll of pixels so that the neural network can consider each pixel individually. This simplifies the learning process for the neural network because dealing with one pixel at a time is more manageable and helps it learn patterns effectively.

Tutorial: Go to and read the "Flatten" tutorial page. [Direct Link]

The flatten process is what gets the picture ready for the neural network to study it one pixel at a time.

Tutorial: Drag and drop the "flatten" block into the workspace. Put this block within the "model" block.

Fully Connected Layer

A fully connected layer connects every neuron from the previous layer to every neuron in the current layer. It's having each neuron in one layer talk to every neuron in the next layer. This means that information from each neuron in the previous layer is considered when making predictions or decisions in the current layer.

Tutorial: Go to and read the "Fully Connected Layer" tutorial page. [Direct Link]

The number of "things it's learning" in the fully connected layer block refers to the number of neurons within that layer. In an abstract way, the number of neurons within that layer represents the number of independent things the layer will learn. The more "things it's learning", the more complex patterns the layer can learn.

Tutorial: Drag and drop the "fully connected layer" block into the workspace. Put this block within the "model" block and after the "flatten" block.

In the example, the fully connected layer is needed for the last part of the neural network, where it learns to recognize and differentiate between 10 different types of numbers (0 to 9). Each neuron in this layer is responsible for learning to recognize a specific number. By connecting all the neurons, the fully connected layer enables the neural network to combine the information learned from earlier layers and make a final decision about which number is present in the input data.

Tutorial: Go to and run the "Let's Run Our Neural Network!" tutorial page. [Direct Link]

We are now where we left off the last time we ran our model! The blocks that are currently inside of the model are the same as what was inside of the temporary "brain" block.

Activation Functions

An activation function is a math operation applied to the output of each neuron in a layer. It is what actually allows the neural network as a whole to learn and represent complex patterns in the data. Activation functions help the network make better decisions by determining whether a neuron should be used or not based on its input.

Tutorial: Go to and read the "Activation (Part 1)" tutorial page. [Direct Link]

An activation function influences how much the neuron should be used based on the information it has learned. This activation is what allows the neural network to capture intricate relationships in the data and make smarter decisions.

Tutorial: Drag and drop the “activation” block into the workspace. Put this block within the “model” block and after the “fully connected layer” block. Try running it.

This activation function alone shouldn’t make much of a difference. There are different types of activation functions which use different types of math operations. Choosing the right one is essential for the network’s learning ability and overall performance.

Note: It is possible to apply multiple activation functions after a single layer. This is not very common in practice, and it may produce surprising results.

The “sort” activation block is representative of the softmax function. Its result gives us a probability distribution that contains the likelihood (or probability) of each label being correct, and the label with the highest probability is automatically considered to be the neural network’s prediction.

Note: The actual math operation works as follows: the function takes an input vector (like the flattened sequence), exponentiates each element, and then normalizes the values to make sure they sum up to 1 (the programming equivalent of making sure they add up to 100%). This normalization allows the output to be interpreted as probabilities.

This is commonly used in the output layer of neural networks for classification (which is what we are using it for now in the tutorial).

Neural networks can have multiple layers. This allows the neural network to learn more intricate features and relationships within the data. Each layer in a neural network captures different aspects of the input.

Tutorial: Go to the ”Activation (Part 2)” tutorial page. [Direct Link] Drag and drop another “fully connected layer” block and “activation” block (using the “boost” activation) into the workspace. Put these blocks within the “model” block and after the “flatten” block. Try running it.

Now, it should be somewhat better. Still, it is not perfect, but it’s better than doing it manually! In the provided example, adding a new fully connected layer allows the neural network to learn additional patterns and relationships in the data.

The “boost” activation represents the ReLU (or Rectified Linear Unit) function. This activation function helps the neural network learn about more complex features in the data.

Note: The actual math operation works by outputting the same value for any positive values and zero for any negative values from the input.

In the example, adding a ReLU activation after the new fully connected layer allows the network to identify patterns in the pictures, such as curves, straight lines, and simple shapes.

Note: There is a helpful hint contained in the tutorial that can explain why layering works through a metaphor.

Tutorial: Go to and run the "Let's Run Our Fully Complete Neural Network!" tutorial page. [Direct Link]

It should be much more accurate. Still, it's not perfect, but for a model we've created in a short period of time, it's performance isn't too bad.

5 Convolutional Neural Networks

A Convolutional Neural Network (CNN) is a specialized type of neural network designed specifically for pattern recognition tasks. Unlike traditional fully connected neural networks, CNNs are particularly effective in understanding layout and local patterns within images.

> **Tutorial:** Go to and read the "Creating a Super Smart Brain" tutorial page. [Direct Link]

CNNs are applied before flattening the image to leverage their ability to understand layout and local patterns. By examining small parts of the image at a time, the network can identify specific features, edges, and textures, enabling it to learn representations that are unaffected by placement, rotation, and size changes. This type of learning makes CNNs highly effective in image-related tasks.

After the CNN, the output is typically flattened before passing through the fully connected layers for final decision-making. This combination of the CNN and fully connected neural network allows CNNs to efficiently learn and recognize complex patterns in images, making them powerful tools in computer vision applications.

Loading

The pictures of clothes are more complex for neural networks to understand compared to the original pictures of numbers. This is due to the diverse shapes, styles, and textures of clothing items. Unlike the relatively consistent and distinct patterns of digits, clothing items have fine details and complex textures, making it more challenging for models to generalize effectively. Successfully classifying clothing items requires models to capture a broader range of features, making the task more complex than recognizing handwritten digits.

> **Tutorial:** Go to and read the "Loading Clothes" tutorial page. [Direct Link] Change the "load" block dropdown to "clothes".

Convolutional Layer

A convolutional layer is responsible for scanning pictures to identify important patterns and features. The convolutional layer consists of filters, which act as the components looking at the image to uncover distinct elements. These filters systematically move across the image, capturing local features and spatial layout. Smaller filter sizes focus on fine details, enabling the network to recognize intricate features, while larger filter sizes look for more prominent features in the images.

Tutorial: Go to and read the “Convolution” tutorial page. [Direct Link] Drag and drop the “convolution” block into the workspace. Put this block within the “model” block and before the “flatten” block.

Operations

An operation is a technique applied after convolutional layers. The purpose of operations is to keep track of the most important features and change how the CNN focuses on them. Pooling helps the model become better for when there are new changes and allows the network to concentrate on essential information for subsequent layers.

Combining activation functions with operations enhances the CNN’s ability to understand layout patterns and make more effective decisions during the learning process.

Tutorial: Go to and read the “Operations (Part 1)” tutorial page. [Direct Link] Drag and drop the “activation” block (with “boost” selected) into the workspace. Put this block within the “model” block and after the “convolution” block.

The “pooling” block contains the max pooling operation. This is an operation commonly used after convolutional layers that divides the picture into small sections and selects the most important feature from each region, ignoring the rest. This process captures the most important features while maintaining only the essential information. It helps make the network more efficient and allows the CNN to focus on the most relevant aspects of the image.

Tutorial: Drag and drop the “pooling” block into the workspace. Put this block within the “model” block and after the “activation” block.

Tutorial: Go to and run the “Let’s Run Our Fully Complete CNN!” tutorial page. [Direct Link]

Now, the model should be able to label some more of the clothes accurately! However, let’s learn how we can make it significantly more accurate!

Note: An additional layer was automatically added to the CNN. CNNs can also have multiple layers in the same way that fully connected neural networks can.

6 Model

So far, much of the focus has been on neural networks. However, we can also see notable benefits when we make changes to the way the model is ran directly.

Optimizers

Optimizers are algorithms that control the learning process of a model by adjusting its own settings while it's learning. The goal for this algorithm is to minimize the error between what the model predicts and the actual correct answers. Optimizers play an important role in determining how quickly and effectively a model learns.

> **Tutorial:** Go to and run the "Optimizer" tutorial page. [Direct Link] Change the model optimizer from "quick" to "smart".

With this change, we should see a notable difference in the quality of the model's labelling.

The "quick" optimizer uses Stochastic Gradient Descent (SGD). SGD adjusts its understanding step by step, taking steps in the direction that helps it improve towards the right answer. It considers what it has learned so far and adjusts its approach based on that. It's quick but may have a less accurate solution.

The "smart" optimizer uses Adam (short for Adaptive Moment Estimation). Adam is an adaptive learner. Adam also adjusts its understanding step by step but it is adaptable. This adaptability allows Adam to take larger steps in the early stages of training and smaller steps as it approaches towards solutions. It is often considered a slower optimizer in comparison to SGD.

Epochs

An epoch refers to a single pass through of all of the pictures to learn. It is how many times the model will study the pictures its given before looking at a new set of pictures. Increasing the number of epochs allows the model to see the data more times, potentially improving its ability to learn and generalize patterns.

> **Tutorial:** Go to and run the "Epochs" tutorial page. [Direct Link] Change the amount of epochs from "7" to "15".

This will take longer to run, but it will give us much more accurate results. However, it's essential to find a balance because too many epochs can lead to overfitting, where the model becomes too used to the training data and performs poorly on new, unseen data.

Loss Functions

Loss functions are metrics that tell the model how well the model is doing while it's learning. They measure the difference between the model's predictions and the actual correct answers, providing feedback for the model to adjust and improve its performance. Simply, it's like getting a score after taking a test, helping the model learn from its mistakes.

Tutorial: Go to and run the "Loss Function" tutorial page. [Direct Link] Change the loss function from "category comparer" to "difference checker".

This change will likely not be better than the "category comparer". However, it may be interesting to see if there are patterns in the results!

The "category comparer" loss function uses categorical cross-entropy. Categorical cross-entropy tells the model if its guess for a category is right or wrong and provides information on how far off it was from the correct answer. It's commonly used for classification tasks where the goal is to assign an input to one of multiple predefined categories, like recognizing digits or types of clothing in images.

The "difference checker" loss function uses mean absolute error. Mean absolute error measures how far every single guess the model made is from the correct answer and gives an average error. It's often used for tasks where the goal is to predict a continuous value, like predicting the price of a house or the temperature.

Tutorial: Go to and run the "Let's Run Our Very Smart CNN!" tutorial page. [Direct Link]

We've successfully made a very accurate CNN! At this point, it should be getting most of its predictions right.

Class Question: Try changing the inner model to be just the fully connected neural network. Is it better or worse? How does changing the other model values impact this? How does changing the type of pictures being loaded impact this?

7 Advanced Topics

Additional Activation Functions

Tutorial: Go to and read the "Activation (Part 3)" tutorial page. [Direct Link]

The "squeeze" activation function uses sigmoid. Sigmoid squashes the input values between 0 and 1. It is commonly used in problems where the goal is to output probabilities.

Class Question: Do you see any patterns with using the "squeeze" activation function? How does it compare with the other activation functions we've been using? Based on it's description, do you think it might work better for something else?

The sigmoid function is powerful for "squeezing" the output into a range, making it useful when the network needs to emphasize or suppress certain features based on their importance.

Tutorial: Go to and read the "Activation (Part 4)" tutorial page. [Direct Link]

The "balance" activation function uses tanh. Tanh (short for hyperbolic tangent) maps input values to a range between -1 and 1.

Class Question: Do you see any patterns with the "balance" activation function? How does it compare with the other activation functions we've been using? Based on it's description, do you think it might work better for something else?

Similar to the sigmoid, tanh is used in scenarios where the network needs to output values within a specific range. It is particularly useful when dealing with data that has negative and positive components.

Additional Operations

Upsampling is an operation that increases the size of a picture in a neural network. Upsampling is used to increase the size of the picture. It's making the picture larger to allow the model to examine finer details.

Tutorial: Go to and run the "Operations (Part 2)" tutorial page. [Direct Link]

Class Question: Do you see any patterns with the "upsampling" operation? How does it compare with the "pooling" operation we've been using?

When the size is changed for upsampling, it determines how much the image is enlarged or how detailed the projected image becomes. The operation is commonly used in various tasks where preserving tiny details is important for the model's performance.

Tutorial: Go to and read the "Summary and Conclusion" tutorial page. [Direct Link]

Class Question: Can you make a model that gets EVERYTHING correct in the slider based off the tools we've given you?

Note: Please fill out the survey and please encourage your students to do so as well! This information helps us develop the MacLea project! [Direct Link]

Recurrent Neural Networks

Recurrent Neural Networks (RNNs) are a specialized type of neural network architecture designed to handle sequences of data. Typically, they are used for tasks involving sequences like sentences that may have different lengths. RNNs have a unique ability to consider information from previous steps when processing the current one.

Tutorial: Go to and read the "Recurrent Neural Networks (experimental)" tutorial page. [Direct Link] Can you figure out how to make it work?

In the context of image classification, applying an RNN might seem unconventional, as images are usually represented as fixed-shape grids. However, the RNN can be adapted for image processing.

In this case, the RNN is treating each row of pixels in the image as a sequence, as if it were processing a sequence of words in a sentence. The RNN scans through the rows one by one, finding patterns and comparing them to what it learned from previous rows. This approach allows the network to capture patterns that extend across multiple rows, potentially learning relationships between parts of the image.

Quizzes

This following section contains a set of quizzes that may be useful as handouts to students after each topic.

Intro Quiz

Question 1: What is Artificial Intelligence (AI)?

a) A type of computer game
b) A field focused on making computers do things on their own
c) A programming language
d) A cooking technique

Answer: B

Question 2: What is Machine Learning (ML)?

a) A type of art created by computers
b) A part of AI that teaches computers how to learn from data
c) A type of music played by machines
d) A method for fixing broken computers

Answer: B

Question 3: What is machine learning's significance in AI?

a) Making computers think like humans
b) Teaching computers to learn from experience
c) Creating a smart assistant

Answer: B

Question 4: What is the relationship between AI (artificial intelligence) and machine learning (ML)?

a) AI is a part of machine learning
b) Machine learning is a subset of AI
c) AI and machine learning are unrelated
d) AI is the process of training computers using examples

Answer: B

Question 5: Why is machine learning considered the "thinking process" of AI?

a) It allows computers to make decisions
b) It is responsible for creating art
c) It helps computers learn from experience
d) It enables computers to play games

Answer: C

Architecture Quiz

Question 1: What role does a machine learning architecture play?

a) It loads data into the system
b) It guides the flow of information and organizes the learning process
c) It focuses on classifying images
d) It handles the brain's thinking process

Answer: B

Question 2: Why is loading data challenging in machine learning?

a) Machine learning models don't require large datasets
b) Data comes only in one format, making it easy to handle
c) Handling different data types and converting them for machine learning is complex
d) Loading data is the final step in machine learning architecture

Answer: C

Question 3: What is the "model" in machine learning?

a) The block used for loading data
b) The final decision-making step in classification
c) The brain that remembers patterns and relationships learned
d) The tutorial page guiding the classification process

Answer: C

Question 4: What is the task of "classification" in machine learning?

a) Assigning categories or labels to images based on learned patterns
b) Loading data efficiently
c) Designing the brain for the AI
d) Running the "Let's Try Running It!" tutorial page

Answer: A

Neural Networks Quiz

Question 1: What is a neural network inspired by?

a) Trees
b) The human brain
c) Computers
d) Rivers

Answer: B

Question 2: What does the "flatten" operation do in a neural network?

a) Increases image complexity
b) Rearranges the structure of pictures
c) Adds more layers to the network
d) Reduces the number of neurons

Answer: B

Question 3: What does the "fully connected layer" do in a neural network?

a) Connects every neuron from one layer to every neuron in the next
b) Separates neurons to work independently
c) Increases image resolution
d) Adds more layers to the network

Answer: A

Question 4: What does an activation function do in a neural network?

a) Changes the color of pixels
b) Rearranges neurons in a layer
c) Applies a math operation to the output of each neuron
d) Adds more layers to the network

Answer: C

Question 5: Why can neural networks have multiple layers?

a) To confuse the learning process
b) To make the network slower
c) To model more complex functions and learn intricate patterns
d) To reduce the accuracy of predictions

Answer: C

CNN Quiz

Question 1: What is a Convolutional Neural Network (CNN) specialized for?

a) Solving mathematical equations
b) Understanding patterns in images
c) Playing video games
d) Writing essays

Answer: B

Question 2: What does a convolutional layer do in a CNN?

a) Identifies important patterns and features
b) Connects neurons in a layer
c) Applies math operations to pixels
d) Rearranges the structure of pictures

Answer: A

Question 3: Why is the convolutional layer effective in image-related tasks?

a) It focuses on globa
b) It looks at the entire image at once
c) It captures local patterns and spatial hierarchies
d) It reduces the size of input images

patterns

Answer: C

Question 4: What is the purpose of the "pooling" block in a CNN?

a) Increases image complexity
b) Divides the picture into small sections and selects important features
c) Connects every neuron from one layer to every neuron in the next
d) Rearranges the structure of pictures

Answer: B

Question 5: What is the purpose of operations in a CNN?

a) What is the purpose of operations in a CNN?
b) To reduce the size of the input while keeping important features
c) To create complex patterns in images
d) To confuse the learning process

Answer: B

Model Quiz

Question 1: What do optimizers do in a machine learning model?

a) Make the model colorful
b) Adjust the learning process by controlling settings
c) Increase the size of the model
d) Add new features to the model

Answer: B

Question 2: What does an epoch represent in machine learning?

a) A single pass through all of the pictures to learn
b) A type of neural network
c) The learning speed of the model
d) A measure of model accuracy

Answer: A

Question 3: Why is it important to find a balance when setting the number of epochs?

a) To speed up the learning process
b) To avoid running out of memory
c) To prevent the model from becoming too specialized
d) To increase the model's size

Answer: C

Question 4: What do loss functions measure in a machine learning model?

a) The model's color accuracy
b) The learning speed of the model
c) The difference between the model's predictions and actual target values
d) The number of epochs completed

Answer: C

Advanced Quiz

Question 1: Why is the sigmoid function powerful for the “squeeze” activation?

a) It enhances image resolution
b) It squashes input values between 0 and 1
c) It maps input values to a range between -1 and 1
d) It increases the size of a picture

Answer: B

Question 2: What does the upsampling operation do in a neural network?

a) Reduces the size of a picture
b) Enhances image resolution
c) Removes fine details
d) Zooms out on a picture

Answer: B

Question 3: In upsampling, what does the size determine?

a) The color of the image
b) How much the image is enlarged
c) The activation function used
d) The learning speed of the model

Answer: B

RNN Quiz

Question 1: What is a unique ability of Recurrent Neural Networks (RNNs)?

a) Handling fixed-shape grids
b) Ignoring information from previous steps
c) Processing sequences of data
d) Scanning through columns of pixels

Answer: C

Question 2: Why might applying an RNN for image classification seem unconventional?

a) RNNs are not effective for image data
b) Images are usually represented as fixed-shape grids
c) RNNs cannot handle sequences
d) Image processing does not involve sequences

Answer: B

Question 3: In the context of image classification, how does the RNN treat each row of pixels?

a) As a fixed-shape grid
b) As a sequential input
c) Ignoring patterns in each row
d) Scanning through columns one by one

Answer: B

Question 4: What does the RNN do when processing each row of pixels in image classification?

a) Ignores previous rows
b) Finds patterns and compares them to the next row
c) Focuses on individual pixels
d) Treats rows as independent entities

Answer: B

Additional Resources

These are additional resources that may be beneficial to show in class or to watch on your own to get a better understanding of the fundamentals.

- **MIT OpenLearningLibrary course for ”6.390 (6.036) Introduction to Machine Learning”:** There are many additional resources that include videos and notes. The content covered in this goes into far more detail than what is currently covered here. [Direct Link]
- **IBM Video on ”AI vs Machine Learning”:** This video gives a great high level overview of the differences between AI and machine learning. [Direct Link]
- **KDnuggets article on ”Demystifying Machine Learning”:** This article gives a great overview of what machine learning is. [Direct Link]
- **3Blue1Brown video on ”But what is a neural network? — Chapter 1, Deep learning”:** This video gives a great explanation of neural networks and their parts for beginniners. [Direct Link]

Machine Learning Definitions

This is a quick reference guide of many of vocabulary words used throughout this course.

- **Artificial Intelligence (AI)** — A field of computer science focused on creating machines capable of intelligent behavior, often involving learning and problem-solving.
- **Machine Learning (ML)** — A subset of AI that allows machines to automatically learn and improve from experience without being explicitly programmed.
- **Loading** — The process of bringing data into a program for processing and analysis.
- **Model** — A representation used in machine learning to make predictions or decisions based on input data.
  - **Optimizer** — An algorithm used during model learning to minimize the difference between predicted and actual outcomes.
    - **Quick (SGD (Stochastic Gradient Descent))** — A popular optimization algorithm in machine learning that iteratively searching for the correct label.
    - **Smart (Adam)** — An adaptive optimization algorithm commonly used in machine learning.
  - **Epoch** — One complete pass through the entire training dataset during model learning.
  - **Loss Function** — A function used to measure the difference between predicted and actual values used to update and improve the model.
    - **Category Comparer (Categorical Crossentropy)** — A loss function used in classification tasks to measure the difference between predicted and true class labels.
    - **Difference Checker (Mean Absolute Error)** — A loss function used to quantify the average difference between predicted and true values.
- **Classify** — The model's process of assigning input data to predefined categories or classes.

- **Neural Networks** — A type of model inspired by the human brain. These are used in machine learning for learning and decision-making.
  - **Neuron** — A unit in a neural network that processes information.
  - **Flatten** — A layer in neural networks that converts 2D pictures into 1D data.
  - **Fully Connected Layer** — A layer in a neural network where each neuron is connected to every neuron in the previous and subsequent layers.
  - **Activation Functions** — A function applied to the output of a neuron to allow for the learning of complex patterns.
    - **Sort (Softmax)** — An activation function used in the output layer of a neural network for multi-class classification.
    - **Boost (ReLU (Rectified Linear Unit))** — An activation function commonly used in middle layers of neural networks to learn complex patterns.
    - **Squeeze (Sigmoid)** — An activation function that squashes output values between 0 and 1.
    - **Balance (Tanh)** — An activation function that squashes output values between -1 and 1.
- **Convolutional Neural Networks (CNN)** — A type of neural network designed for processing images
  - **Convolution** — A layer used in CNNs to extract features from input pictures.
  - **Operation** — A technique in CNNs that involves focusing information in specific regions of pictures.
    - **Pooling** — An operation in CNNs that shrinks input data by selecting the maximum or average from a group of pixels.
    - **Upsampling** — An operation that increases the size of input data.
- **Recurrent Neural Networks (RNN)** — A type of neural network model designed to handle sequential data by maintaining a memory of past inputs.