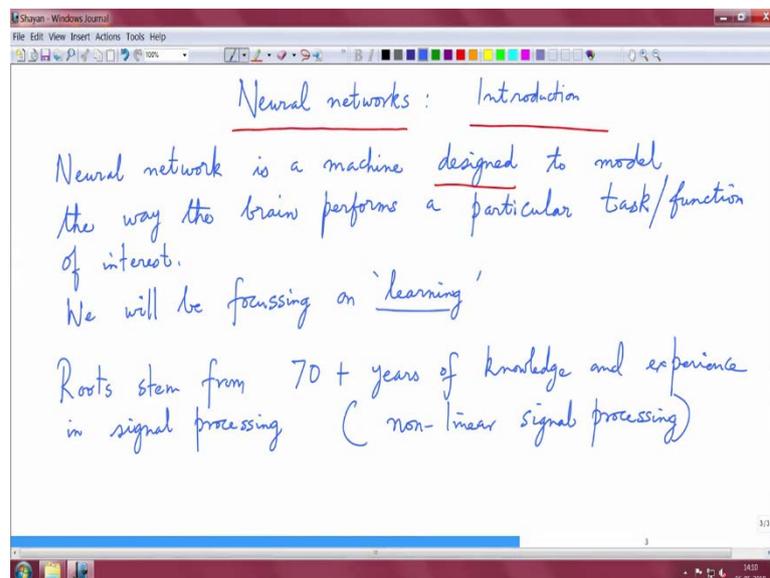


Neural Networks for Signal Processing – I
Prof. Shayan Srinivasa Garani
Department of Electronics Systems Engineering
Indian Institute of Science, Bengaluru

Lecture – 02
Introduction to Neural Networks

Let's begin with an introduction to neural networks. We will explore what neural networks are, why we need them, the tasks they can naturally perform, and why they are considered a basis for artificial intelligence (AI).

(Refer Slide Time: 00:47)

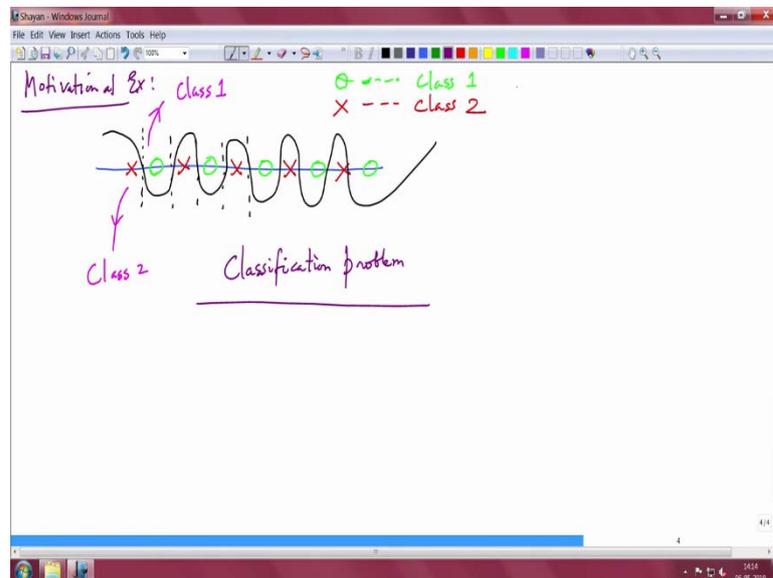


A neural network is a machine designed to model the way the brain performs a particular task or function of interest. We have seen the motivation behind how brain computation happens, and we aim to build engineering systems or machines that can mimic the brain's operations. This is our motivation: to create neural networks designed to model the brain's performance on specific tasks. Throughout this course, our primary focus will be on learning.

Our focus in this course will be on learning. If you consider the motivation behind this subject, it stems from the traditional area of signal processing. Traditional signal processing is linear in nature, whereas neural networks represent non-linear signal

processing. Our foundation is built on over 70 years of knowledge and experience in signal processing. Neural networks provide a means for non-linear processing of signals and efficient ways to realize tasks.

(Refer Slide Time: 03:58)



Let's delve into a motivating example. Imagine we have points arranged along a line as follows. I'll distinguish them using two different colors. Here, we have crosses belonging to one class. Now, using a different color, let's say green, we have circles interspersed among these crosses, belonging to another class. This represents class 1 and class 2, respectively.

Now, consider this question: how many lines are needed to separate these classes? Imagine extending this scenario infinitely along the real line, where crosses represent all odd integers and circles represent all even integers. To effectively separate these classes with lines, how many such lines would suffice?

The answer lies in needing a line between each pair of points to distinctly classify them. Attempting to separate them with just one line is inadequate. Visualizing this, we'd require boundaries like these, several segments where different regions can be identified to classify between crosses and circles.

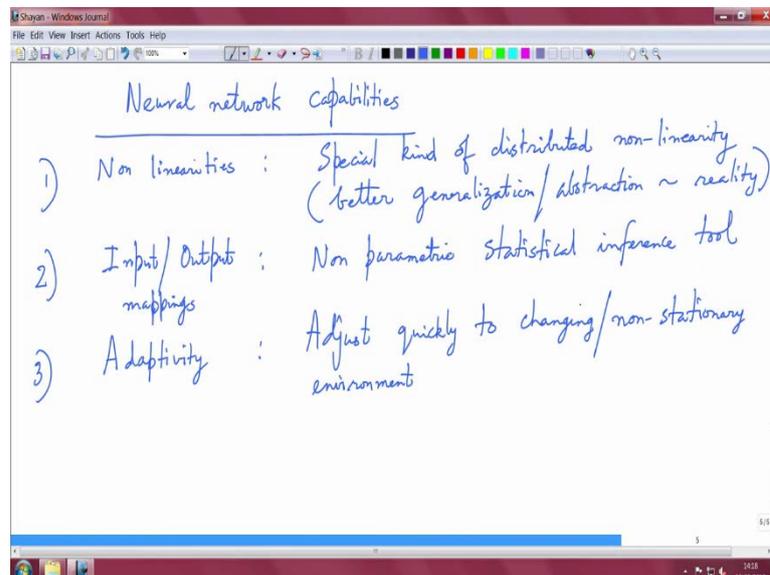
But why can't we approach this differently? Consider a curve, illustrated again with this black marker. Suppose we have a curve that looks like this. If we designate everything on

one side of the curve as circles and everything on the other side as crosses, then this side represents class 2 and the other class 1. Here, unlike with linear boundaries, we encounter non-linearity. A single line fails to separate these points effectively, but a curve proves capable of distinguishing them.

This simple example underscores everyday scenarios where linear boundaries are insufficient to discern patterns. To address such non-linear challenges, neural networks emerge as a solution. They excel in learning and leveraging these connections, constructing curves that effectively differentiate between classes.

This serves as a motivational example. Consider if we were to ask, "What can a neural network achieve?" It operates as a non-linear, distributed machine capable of generalizing and abstracting real-world objects and entities. Neural networks learn to map inputs to outputs and adapt themselves accordingly, showcasing their versatility and adaptability.

(Refer Slide Time: 08:25)



Let me summarize this. Moving to the next slide, firstly, a neural network excels in learning non-linearities. It represents a unique form of distributed non-linearity, capable of achieving superior generalization and abstraction that closely mirrors reality. This stands out as one of its core capabilities.

Next, consider its ability in input/output mappings. Traditional pattern recognition systems often rely on parametric or non-parametric forms. If you have prior knowledge of the

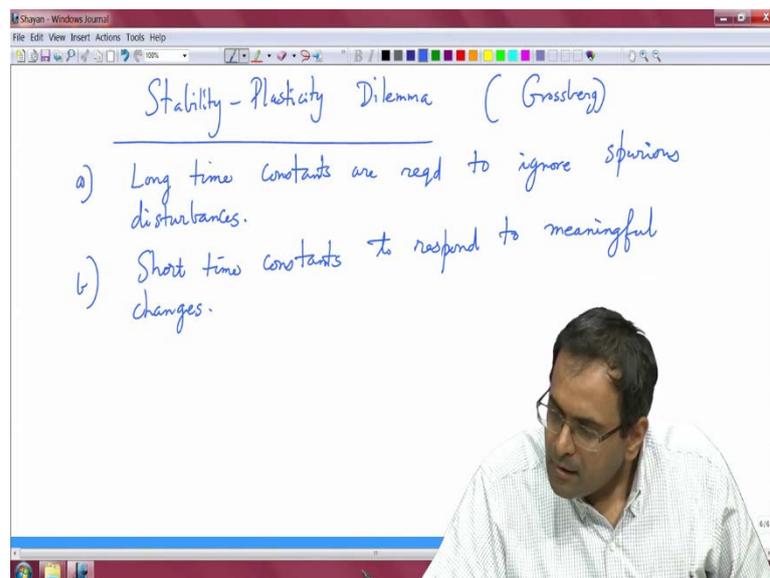
distribution and its parameters, statistical techniques like maximum likelihood estimation can yield meaningful descriptions in parametric forms.

However, neural networks offer a significant advantage by being non-parametric. They learn directly from input data without assuming a specific parametric form of the underlying distribution. While neural network models do incorporate parameters within their architecture, they operate without a predefined parametric structure, marking this as a pivotal capability. In summary, neural networks serve as powerful non-parametric tools for statistical inference.

One of the most crucial aspects is adaptability. The ability to swiftly adapt to changes in a dynamic or non-stationary environment is a critical trait of any learning machine. Traditional systems are designed for stationary environments where parameters can be estimated and adjusted accordingly. However, in non-stationary environments, statistics vary over time, presenting a challenge.

A learning machine must possess the capability to rapidly adjust to these changing conditions, a feature inherently embedded within neural networks. With these capabilities, we move beyond conventional linear signal processing techniques towards robust non-linear approaches. These techniques excel in handling input nonlinearities, learning complex input-output mappings, and exhibiting adaptability to varying conditions.

(Refer Slide Time: 12:48)



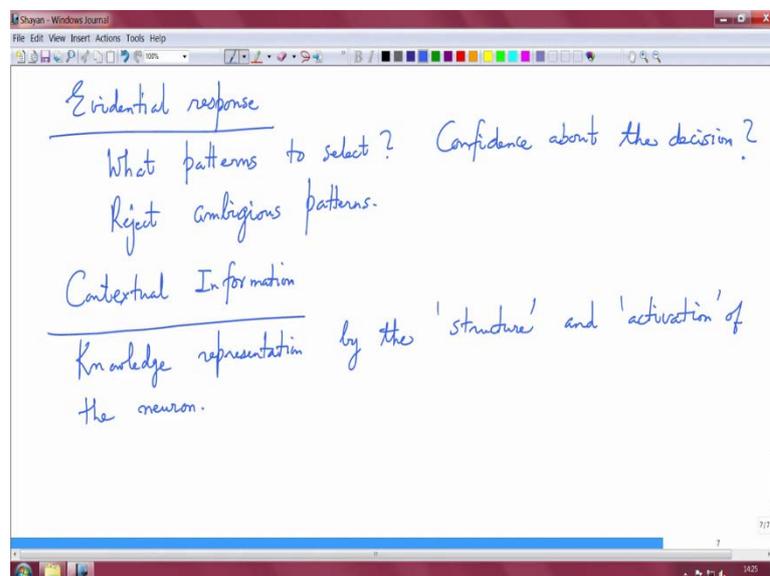
Now, let's discuss the stability-plasticity dilemma, initially proposed by Grossberg. The essence of this dilemma revolves around a fundamental concept: long time constants are essential to disregard irrelevant disturbances, while short time constants are necessary to promptly adapt to significant changes.

To illustrate, consider a scenario where a learning machine, such as a bot tasked with understanding a classroom lecture, must navigate this dilemma. The short time constant corresponds to the ability to swiftly transition between topics. It enables the machine to detect and respond to meaningful changes in topic without being distracted by irrelevant signals or arbitrary time constraints.

This refers to handling spurious disturbances, where a long time constant is crucial for effectively filtering out such distractions. There exists a delicate balance between employing a short time constant for local responsiveness within a limited timeframe and adopting a longer time constant to disregard irrelevant disturbances. This interplay is a critical capability that any learning machine must possess.

In essence, stability involves the ability to ignore irrelevant disturbances, while plasticity ensures responsiveness to significant changes over short durations. Determining what qualifies as important versus trivial requires careful mathematical modeling to guide the learning machine's understanding. With these concepts in mind, we must identify other essential properties that such a learning machine should possess.

(Refer Slide Time: 15:44)



An essential requirement is the ability to provide an evidential response. This entails selecting meaningful patterns with confidence in decision-making while avoiding ambiguous ones. For instance, consider an AI-equipped license plate recognition system focused on tracking specific plates. Its attention is squarely on identifying and tracking the intended plate amidst potential distractions and ambiguous patterns in its vicinity.

Imagine an application scenario where a drone is tasked with tracking a specific truck bearing a unique license plate in a dynamically changing environment. To effectively learn and perform this task, the system must discern relevant patterns while disregarding ambiguous ones. It must also maintain confidence in its decisions and continuously improve its performance—an embodiment of evidential response.

Contextual information plays a pivotal role in learning systems, particularly in knowledge representation. In neural networks, knowledge is represented and processed through the structured activation of neurons. Neurons serve as nonlinear processing elements crucial for capturing and utilizing contextual information effectively.

So, what exactly do we mean by contextual information? When we consider a local window, within this window lies a specific context. This can be likened to distinguishing between short-term and long-term memory. Contextual information restricts the inclusion of events too far in the past or future, emphasizing connectivity within a relevant timeframe.

During conversations or sequences of meaningful patterns over time, there exists a crucial context that must be integrated into learning machines. Therefore, the structural layout of information and the activation of neurons play vital roles in capturing and utilizing this contextual data effectively.

An excellent illustration is automated dialogues with bots. When interacting with a bot, I issue commands and expect responses based on the current context—both short-term and long-term considerations. Whether I'm instructing it to activate a specific device, adjust room lighting, or perform other tasks, the bot must interpret and respond to contextual cues in real-time, continually learning and improving its responses.

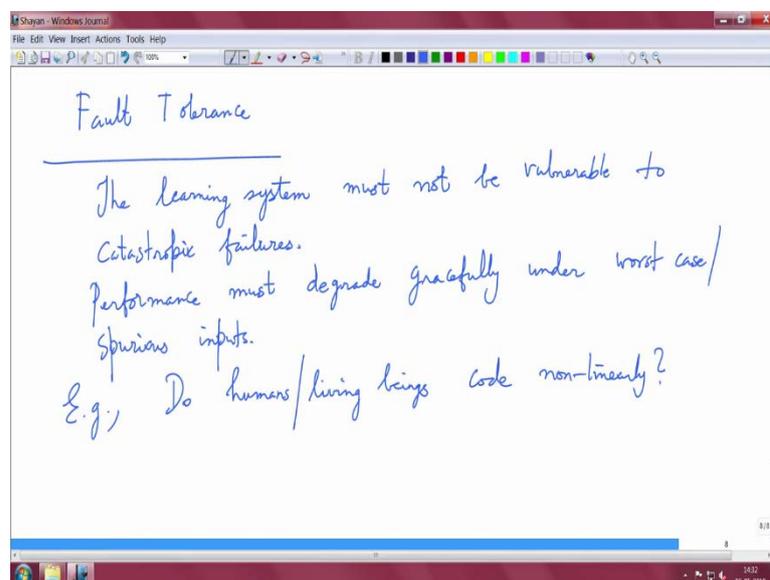
Consider the distinction between short-term and long-term consequences, such as the impact of leaving lights on for several days. Even in such scenarios, a bot should be

capable of adapting and understanding these nuances. This underscores the critical role of knowledge representation in artificial intelligence. To truly automate tasks as humans do, embedding this level of intelligence into machines becomes imperative, establishing a crucial connection.

Another example highlighting contextual information can be found in movies. Imagine watching a three-hour film where specific dialogues within ten or fifteen-minute segments hold localized contextual significance. However, the overarching theme of the story isn't solely reliant on any single dialogue. Understanding the entire context of the three-hour narrative requires synthesizing these individual elements into a coherent whole, capturing both the overarching storyline and the nuanced dialogues.

Now, consider the challenge: can we engineer a machine capable of comprehending and responding to such complex information? Developing intelligence in machines to handle these tasks presents a fascinating and significant endeavor.

(Refer Slide Time: 21:36)



Next, let's consider fault tolerance, a crucial property for any system, whether artificial or natural. In biological systems, fault tolerance is inherent, starting from the selection of genes, which includes mechanisms for immunity against mutations and other disruptions.

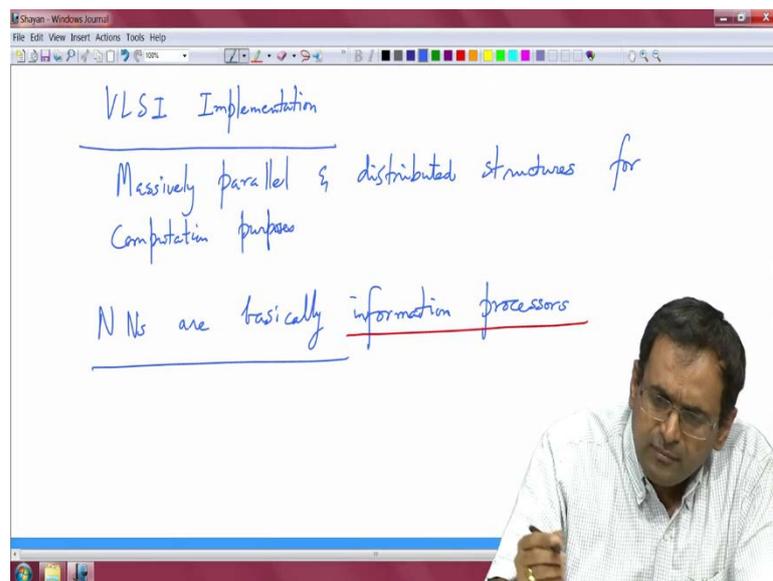
Similarly, in artificial systems, fault tolerance is essential. It ensures that even in the presence of spurious signals or adverse inputs, the system continues to perform reliably

without succumbing to catastrophic failures. Essentially, a learning system must gracefully degrade its performance under worst-case scenarios and remain resilient.

This concept can be exemplified in error correction learning. For instance, if I've learned that the sun rises in the east but encounter an error stating it rises in the west, context helps correct this misconception promptly. Another example involves understanding sentences like "the dog chased the rat" versus "the cat chased the dog," where context dictates the meaningful sequence.

To achieve this robustness, systems must possess error correction learning capabilities, thereby ensuring fault tolerance. This capability mirrors the non-linear nature of learning and fault immunity observed in humans, animals, and other living beings. Understanding and implementing these mappings are crucial steps in imbuing machines with intelligence and ensuring their reliability.

(Refer Slide Time: 25:21)



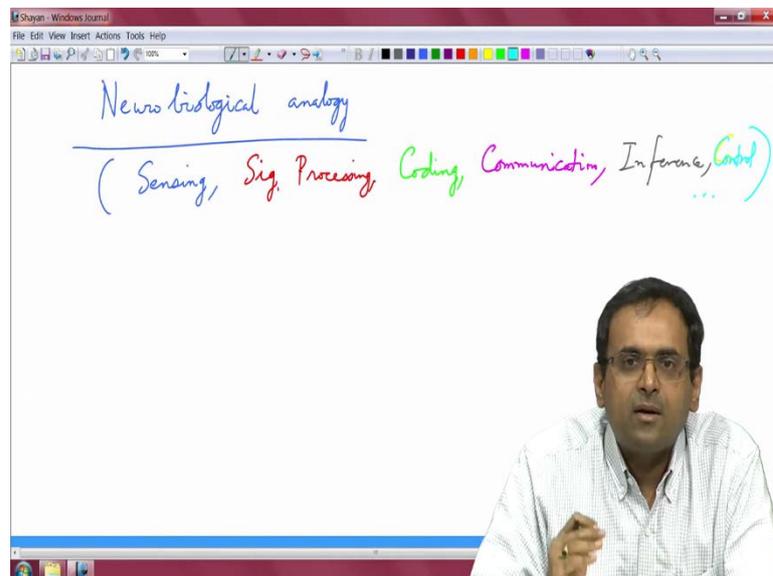
Now, shifting to a more engineering perspective, when we contemplate theory, algorithms, and beyond, we enter the realm of mathematical abstraction where ideas are conceived and models are abstracted. Once these concepts are formalized into algorithms, the next step involves practical implementation, transforming them into hardware, software, or hybrid systems ready for real-world deployment.

When algorithms are translated into practical systems, particularly in terms of VLSI (Very Large Scale Integration) implementation, the focus turns to their implementability. This necessitates designing massively parallel and distributed structures for computational efficiency. Parallel processing enhances throughput significantly, while distributed computing allows tasks to be divided among multiple subprocessors, resulting in substantial computational savings, albeit at the cost of increased hardware footprint.

When considering neuromorphic systems, whether they operate in analog or digital domains, their ability to process signals is crucial. Moving from theoretical algorithms to practical systems involves evaluating the VLSI implementation capabilities of these algorithms. Utilizing the parallel and distributed nature of tasks becomes essential in optimizing their performance.

In essence, neural network systems function as sophisticated information processors. This distinction is significant—they are not merely DSP processors or computational units but are specifically designed to process information. Handling data nonlinearly, these systems learn complex mappings, memorize information, and execute tasks such as pattern association and recall, all integral to artificial intelligence (AI).

(Refer Slide Time: 28:52)



Let's take a moment to consider the neurobiological analogy. When comparing biological and artificial systems, it's essential to understand what we aim to construct. We can

delineate basic tasks such as sensing, signal processing, coding, communication, inference, control, and numerous others.

In human biology, our sensory capabilities encompass touch, smell, vision, and hearing—each providing crucial cues for understanding signals. Signal processing involves extracting features from these sensory inputs, a critical step in interpreting information. Non-linear signal processing, crucial for pattern association and other cognitive tasks, further enhances our cognitive abilities.

Encoded in our biological neurons, knowledge is represented in a compact, parallel, and distributed manner. This spatiotemporal coding occurs naturally within our neural networks. Communication plays a vital role as we respond to various stimuli by associating learned patterns and adapting our responses accordingly. This comprehensive framework enables us to effectively interact with and respond to our environment.

Next, let's consider inference, a critical aspect as it entails decision-making capabilities. Inference is pivotal in guiding the choices we make. Additionally, control and regulation are essential, whether in open-loop or closed-loop systems, where closed-loop feedback mechanisms play a predominant role. These functions are seamlessly integrated into our daily routines, encompassing tasks like signal processing, communication, control, and sensing, albeit often without conscious awareness.

However, when imparting these capabilities to machines through learning, the approach differs significantly from innate human abilities. Machines require structured models and intelligence infused with foundational principles, mathematics, and intricate details to function effectively. This holistic integration of these disciplines is imperative for developing sophisticated machine intelligence.

We will stop here.