**Introduction to Artificial Neural Network**

Summary

The document provides a comprehensive introduction to Artificial Neural Networks (ANN), explaining their structure and function, as well as the different types of learning used in these networks. It also discusses the implementation and training of ANN, and provides an example of their application in predicting pressure distribution in hydrodynamic journal bearings. The document serves as a comprehensive guide to understanding and utilizing ANN for analysis and decision-making purposes.

Introduction to ANN.

This section of the document provides an introduction to Artificial Neural Networks (ANN). It explains that ANN is inspired by the way biological nervous systems process information and consists of multiple layers of neurons. The section discusses the use of ANN for analysis and decision-making in systems, as well as the different types of learning for adjusting the weights of the network. It also covers the mathematical model of the network and terminology associated with neurons and connections. Additionally, the section provides an overview of feed-forward networks and neural learning, including supervised, unsupervised, and reinforced learning.

Types of learning.

This section of the document discusses different types of learning in neural networks, including supervised learning, unsupervised learning, and reinforced learning. It also mentions the back propagation algorithm used in layered feed-forward ANNs. The section goes on to discuss the development and implementation of artificial neural networks, as well as training and prediction quality. The section concludes by discussing the use of artificial neural networks for prediction of pressure distribution in hydrodynamic journal bearings. The section includes references to various research papers on the topic.

**Communication in Neuronal Networks**

Summary

The document discusses the efficiency and communication in neuronal networks. It explains that the brain is efficient and capable of computation, and that cortical networks follow design principles similar to electronic networks. Neuronal networks transfer large amounts of information between brain areas and can process synaptic inputs to implement various operations. The design of neuronal networks reduces the resources required for tasks, and the brain is organized to reduce wiring costs. Energy supply limits signal traffic in the brain, and miniaturization affects energy consumption and noise in neuronal networks. Additionally, the document explores the relationship between noise and energy efficiency in the nervous system, suggesting that improvements in information transfer are expensive due to noise. Distributing signals appropriately in time and space and using sparse coding and redundancy reduction can improve energy efficiency. Regulating signal traffic at the level of individual synaptic connections and neural plasticity may also contribute to energy efficiency. Overall, the document emphasizes the precision and efficiency of the brain's structure and function.

Neuronal network communication efficiency

This section of the document discusses the communication in neuronal networks. It states that brains are efficient, capable of computation, and excellent at communication. It explains that the structure and function of cortical networks follow design principles similar to those used in electronic networks. The document also mentions that neuronal networks serve as communication networks for transferring large amounts of information between brain areas. It explains that neurons can receive and deliver signals at a high number of synapses and that they can process synaptic inputs to implement various operations. The section also talks about the efficiency of the design of neuronal networks, stating that they reduce the resources required for tasks. It mentions that brains have evolved to operate efficiently and that economy and efficiency are guiding principles in physiology. The section then goes on to discuss the geometrical and biophysical constraints on wiring in the brain, explaining that the brain is organized to reduce wiring costs. It also mentions the role of energy consumption in neural communication, stating that energy supply limits signal traffic in the brain. The section concludes by mentioning the effects of miniaturization on energy consumption and noise in neuronal networks.

Noise impacts efficiency.

This section of the document discusses the relationship between noise and energy efficiency in the nervous system. It suggests that noise limits the information transfer down single neurons at high rates, making improvements expensive. The document also mentions the importance of distributing signals appropriately in time and space to improve efficiency. It explains that sparse coding and redundancy reduction can help in achieving energy efficiency in the nervous system. The section also touches on the idea that the energy efficiency of the brain is improved by regulating signal traffic at the level of individual synaptic connections between neurons. It concludes by highlighting the precision and efficiency of the brain's structure and function, and the potential role of neural plasticity in directing the brain's resources.

**Training Neural Networks**

Summary

The document discusses the use of the neuralnet package in R for training neural networks in regression analyses. It provides an overview of the package and its capabilities, including training algorithms and the ability to handle multiple covariates and hidden layers. It also explains how to specify the desired number of hidden layers and neurons and provides an example of training a neural network using a dataset. The document introduces additional functions for tracing signals and calculating predictions, as well as calculating confidence intervals for the weights. Overall, the document serves as a comprehensive guide to using the neuralnet package for regression analyses.

Artificial neural networks in regression analysis

This section is part of a larger document discussing the use of artificial neural networks in regression analyses. Specifically, it focuses on the neuralnet package in the R programming language. The section provides an introduction to multi-layer perceptrons (MLP) and the training algorithms implemented in neuralnet. It explains how neural networks can be used to approximate complex functional relationships between covariates and response variables. The neuralnet package allows for flexible training of neural networks with multiple covariates, response variables, and hidden layers. The section also discusses the backpropagation and resilient backpropagation algorithms used in training neural networks. The resilient backpropagation algorithm is particularly efficient in training neural networks for regression analyses. The section concludes by mentioning two other packages for artificial neural networks in R, nnet and AMORE, but notes that neuralnet was specifically built for regression analyses. Overall, this section provides an overview of neuralnet and its capabilities for training neural networks.

R neuralnet package

This section of the document discusses the usage of the neuralnet package in R for training neural networks. It explains how to specify the required number of hidden layers and hidden neurons, as well as the important arguments for the neuralnet function. It provides an example of training a neural network using the infert dataset and describes the important values saved in the result object. It also compares the results of the neuralnet function with the backpropagation algorithm and the nnet package. Finally, it discusses how to visualize the results of the training process using plots of the neural network and generalized weights.

Neural network package.

This section of the document introduces the package neuralnet, which can be used to model functional relationships between covariates and response variables using multi-layer perceptrons. It discusses the compute function, which can be used to trace signals passing through the neural network and calculate predictions for new covariate combinations. It also introduces the confidence.interval function, which calculates confidence intervals for the weights of the neural network. The section concludes with a summary of the paper and acknowledgements.