```
\documentclass{article}
\usepackage{amsmath}
\usepackage{amsfonts}
\usepackage{amssymb}
\usepackage{graphicx}

\title{Neural Networks: A Primer}
\author{}
\date{}

\begin{document}

\maketitle

\section{Introduction}

Neural networks, inspired by the biological neural networks that constitute animal brains, are computational models composed of interconnected nodes (neurons) organized in layers. These networks are capable of learning complex patterns from data and making predictions or classifications based on learned representations. Their ability to learn from data without explicit programming makes them a powerful tool in various fields, including image recognition, natural language processing, and time series forecasting.

\section{Architecture}

A typical neural network consists of three main types of layers:

\begin{itemize}
\item \textbf{Input Layer:} Receives the initial data as input. Each node represents a feature of the input.
\item \textbf{Hidden Layers:} Perform complex transformations on the input data. Multiple hidden layers allow for the learning of hierarchical representations.
\item \textbf{Output Layer:} Produces the network's prediction or classification. The number of nodes in this layer depends on the task.
\end{itemize}
```

Each connection between nodes has an associated weight, representing the strength of the connection. The nodes in a layer receive weighted sums of the outputs from the previous layer, apply an activation function, and then pass the result to the next layer.

\section{Activation Functions}

Activation functions introduce non-linearity into the network, enabling it to learn complex relationships. Common activation functions include:

\begin{itemize}
\item \textbf{Sigmoid:} $\sigma(x) = \frac{1}{1 + e^{-x}}$
\item \textbf{ReLU (Rectified Linear Unit):}(\text{ReLU}(x) = \max(0, x))
\item \textbf{tanh (Hyperbolic Tangent):}(\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}})
\end{itemize}

The choice of activation function depends on the specific application and network architecture.

\section{Learning Process}

Neural networks learn through a process called \textbf{training}. This involves adjusting the connection weights to minimize the difference between the network's predictions and the actual values (the loss function). This is typically achieved using \textbf{backpropagation}, an algorithm that calculates the gradient of the loss function with respect to the weights and uses this gradient to update the weights iteratively. Common optimization algorithms used in this process include:

\begin{itemize}
\item \textbf{Stochastic Gradient Descent (SGD)}
\item \textbf{Adam}
\item \textbf{RMSprop}
\end{itemize}

\section{Conclusion}

Neural networks are a powerful and versatile class of machine learning models. Their ability to learn complex patterns from data makes them applicable to a wide range of problems. However, understanding their architecture, activation functions, and training process is crucial for effectively utilizing their potential. Further research into novel architectures and training techniques continues to push the boundaries of what neural networks can achieve.

\end{document}