**ASSIGNMENT - 2**

**Traditional Machine Learning vs. Basic Neural Networks**

**1. Overview**

Both traditional ML algorithms and neural networks aim to make predictions or decisions based on data. However, they differ significantly in structure, data requirements, feature handling, and application domains.

**2. Key Differences**

| **Aspect** | **Traditional ML Algorithms** | **Basic Neural Networks** |
| --- | --- | --- |
| **Examples** | Linear Regression, Decision Trees, SVM, KNN | Multi-layer Perceptron (MLP), Perceptron |
| **Structure** | Rule-based, often linear or tree-based | Layers of interconnected nodes (neurons) |
| **Feature Engineering** | Manual (requires domain expertise) | Automatic (learns features from raw data) |
| **Data Requirements** | Performs well on small to medium datasets | Requires large datasets for good performance |
| **Complexity Handling** | Limited to simpler patterns | Good at capturing complex, non-linear patterns |
| **Interpretability** | High (e.g., decision trees are transparent) | Low (acts like a black box) |
| **Training Time** | Generally fast and resource-light | Slower and resource-heavy |
| **Generalization** | May overfit if data is high-dimensional | Can generalize better with large data |
| **Hardware Requirement** | Runs on standard CPU | Often requires GPUs for efficiency |

**3. How They Work**

* **Traditional ML**:
  + Learns patterns using mathematical models or heuristics.
  + Requires selecting and preprocessing the most informative features manually.
  + Works well with **structured/tabular data** (e.g., spreadsheets).
* **Basic Neural Networks**:
  + Composed of an input layer, one or more hidden layers, and an output layer.
  + Each neuron in a layer processes input and passes it through an activation function.
  + Uses **backpropagation** and **gradient descent** for learning.
  + Ideal for **unstructured data** like images, audio, and text.

**4. Use Case Comparison**

| **Task** | **Traditional ML Works Well** | **Neural Networks Work Better** |
| --- | --- | --- |
| Predicting housing prices | Linear Regression | Overkill |
| Spam detection | Naive Bayes, SVM | But NN may be overkill |
| Image recognition | Requires manual feature extraction | Learns image features automatically |
| Sentiment analysis | Needs word count/TF-IDF | With embeddings and deep learning |
| Fraud detection (large data) | With engineered features | Handles high-dimensional complex data |

**1. Image Recognition & Computer Vision**

* **Tasks**: Face recognition, object detection, medical image analysis, self-driving car vision.
* **Why DL Wins**: Convolutional Neural Networks (CNNs) automatically learn spatial features from raw pixels, removing the need for manual feature extraction.

**Example**: Identifying tumors in X-ray or MRI scans using CNNs.

**2. Natural Language Processing (NLP)**

* **Tasks**: Machine translation, sentiment analysis, chatbots, summarization.
* **Why DL Wins**: Recurrent Neural Networks (RNNs), LSTMs, and Transformers understand context and semantics better than bag-of-words or TF-IDF approaches.

**Example**: Google Translate, ChatGPT, or spam filtering in emails.

**3. Speech Recognition & Audio Processing**

* **Tasks**: Voice assistants (Siri, Alexa), speech-to-text, speaker verification.
* **Why DL Wins**: Neural networks like RNNs and CNNs learn time-based patterns and noise tolerance directly from audio signals.

**Example**: Real-time transcription or voice command systems in smart devices.

**4. Autonomous Systems & Robotics**

* **Tasks**: Path planning, object avoidance, sensor fusion in robotics or drones.
* **Why DL Wins**: Combines vision, motion, and decision-making in real-time using end-to-end models.

**Example**: Tesla’s self-driving software using neural networks.

**5. Anomaly Detection in High-Dimensional Data**

* **Tasks**: Fraud detection, network intrusion, equipment failure prediction.
* **Why DL Wins**: Neural networks detect subtle, non-linear patterns across many variables, which traditional ML may miss.

**Example**: Credit card fraud detection from millions of transactions.

**6. Generative Tasks**

* **Tasks**: Creating images, music, or text (text-to-image, deepfakes, AI art).
* **Why DL Wins**: Generative Adversarial Networks (GANs) and Transformers produce realistic content by learning complex data distributions.

**Example**: Midjourney, DALL·E, or AI-generated music.

**7. Gaming and Reinforcement Learning**

* **Tasks**: Game playing, robotics control, trading bots.
* **Why DL Wins**: Deep Reinforcement Learning (DRL) models learn strategies and decisions through trial and error in complex environments.

**Example**: AlphaGo defeating human champions in Go.

**Summary**

| **Domain** | **Deep Learning Advantage** |
| --- | --- |
| Image & Video Analysis | Automatic feature learning (CNNs) |
| Language Understanding | Context-aware processing (Transformers) |
| Audio & Speech | Time-series pattern capture |
| Complex Decision Systems | End-to-end learning capability |
| Generative Applications | High-fidelity data generation |

* SKANDAMAHIMA V M