Yield Prediction Model Architecture (Deep Neural Network Regressor)

This document visualizes the structure of the Deep Neural Network (DNN) built for crop yield prediction. The model is a regression network designed to estimate the **Yield in kg/ha** based on a combination of numerical (scaled) and categorical (one-hot encoded) input features.

Preprocessing Pipeline (Input Preparation)

The `ColumnTransformer` is used to prepare the 12 input features before they enter the model:

1. **Numerical Features (8):** `Area` , `N` , `P` , `K` , `temperature` , `humidity` , `ph` , `rainfall`
    - **Transformation:** `MinMaxScaler` is applied to scale these values between 0 and 1.
2. **Categorical Features (4):** `State_Name` , `District_Name` , `Season` , `Crop`
    - **Transformation:** `OneHotEncoder` is applied to convert these nominal features into a sparse matrix, which is then forced into a dense NumPy array (`sparse_output=False`).

The final input shape to the model is large and dynamic, depending on the total number of unique categories (States, Districts, Seasons, Crops) found in the training data, plus the 8 numerical features.

Architecture Breakdown (DNN Regressor)

Component	Layer Type	Node Count	Activation Function	Purpose
Input Layer	Dense (Implicit)	Dynamic*	N/A	Receives the fully processed (scaled and one-hot encoded) feature vector.
Hidden Layer 1	Dense	256	ReLU (`relu`)	Initial large layer for learning complex patterns across the combined numerical and categorical inputs.
Hidden Layer 2	Dense	128	ReLU (`relu`)	Intermediate abstraction layer.
Hidden Layer 3	Dense	64	ReLU (`relu`)	Final compact layer for feature extraction.
Output Layer	Dense	1	Linear (`linear`)	**Regression Output:** Produces a single, continuous numerical value representing the predicted **Yield (kg/ha)**.

**The size of the input layer is `8 + (Total unique categories)` .*

Training and Optimization Details

- **Model Type:** Deep Neural Network Regressor

- **Target Variable (`y`):** `Yield_kg_per_ha` (A continuous numerical value)
- **Optimizer:** `Adam` (Adaptive Moment Estimation)
- **Loss Function:** `MAE` (Mean Absolute Error)
- **Evaluation Metrics:** `MSE` (Mean Squared Error), `MAE` (Mean Absolute Error)
- **Epochs:** 100
- **Batch Size:** 64