Groundwater Level Prediction Using MLR and ANN

Study Area: Western Uttar Pradesh Sugarcane Belt (Muzaffarnagar, Meerut, Baghpat, Bulandshahar, Hapur, Gautam Buddha Nagar, Ghaziabad)

I. Scenario

Groundwater is a crucial resource for irrigation in the sugarcane belt of Western Uttar Pradesh. Monitoring and predicting its fluctuations help in sustainable water management and planning. The **Central Groundwater Board of India** aims to assess groundwater dynamics across the region.

As a data scientist, the goal is to develop predictive models capable of estimating groundwater levels at unsampled locations using climatic (rainfall, temperature), soil (texture, moisture), and temporal (lagged groundwater levels, time index) parameters. Accurate predictions can guide irrigation schedules, crop planning, and water conservation strategies.

II. Objective

The main objectives of this study are:

1. To **build and evaluate Multiple Linear Regression (MLR) and Artificial Neural Network (ANN)** models for groundwater level (GWL) prediction.

2. To **compare model performance**, including predictive accuracy, handling of non-linearity, interpretability, and computational efficiency.

3. To identify the **key factors affecting groundwater levels** in the study region.

Theory:

- **MLR** assumes a linear relationship between predictors and the target variable. It is interpretable but limited in handling complex non-linear patterns.

- **ANN** (specifically Multi-Layer Perceptron, MLP) can capture non-linear dependencies between input variables and the target. Dropout and early stopping help control overfitting.

III. Dataset and Preprocessing

Dataset sources:

- Groundwater levels from **Central Groundwater Board**.
- Climate variables from **Copernicus Climate Data Store** (temperature, precipitation, soil moisture, runoff).
- Soil texture data from **NICES Portal**.
- District boundaries and spatial mapping from **India WRIS Geospatial layers**.

Preprocessing steps:

1. Cleaning missing or inconsistent data.
2. Imputing missing values using interpolation or mean values.
3. Creating **lag features** to capture temporal dependencies.
4. Scaling all features with **MinMaxScaler** for ANN training.
5. Splitting dataset into **training (70%)**, **validation (15% of training data)**, and **testing (30%)** sets.

Theory:

- Scaling ensures that variables with different units do not disproportionately influence the ANN.
- Lag features help in capturing **autocorrelation** in time series, which is essential for groundwater predictions.

IV. MLR Model Results

Model Performance:

Model	AIC	BIC	Adj. R^2
Full Model	5900.88	5954.35	0.398
Climate Only	6287.83	6317.00	0.092
Soil Only	6309.77	6329.21	0.069

Theory:

- **Adj. R^2** measures how well the model explains variance while adjusting for the number of predictors.
- **AIC/BIC** are criteria for model selection; lower values indicate better fit.
- Climate-only or soil-only models show poor performance, suggesting that **both types of features are necessary**.

V. ANN Architecture and Training

Layer Type	Units	Activation	Dropout
Dense	128	ReLU	0.2
Dense	64	ReLU	0.2
Output Dense	1	Linear	-

Hyperparameters:

- Optimizer: **Adam**
- Learning Rate: 0.001

- Batch Size: 64
- Epochs: 200
- Early Stopping: Patience = 10
- Validation Split: 15%
- Runtime: 2.34 seconds (CPU)

Theory:

- **ReLU activation** helps in modeling non-linear relationships efficiently.
- **Dropout** prevents overfitting by randomly ignoring neurons during training.
- **Adam optimizer** adaptively adjusts learning rates to converge faster.

VI. Model Evaluation and Comparison

Model	RMSE	MAE	R^2	Training Time (s)
MLR	5.110	3.776	0.363	-
ANN (MLP)	4.250	2.362	0.559	2.34

Statistical Test:

Test	t-statistic	p-value	Result
Paired t-test	12.5947	0.0000	Statistically Significant

Theory:

- **RMSE** (Root Mean Squared Error) penalizes larger errors more than MAE.
- **Paired t-test** confirms whether ANN significantly outperforms MLR.

VII. Model Explainability

- **Permutation Feature Importance** and **SHAP analysis** highlight:
    - Previous groundwater level (**GWL_lag1**)
    - Time index
    - Floamy soil percentage
    - Rainfall
- **Partial dependence plots** show strong non-linear impact of rainfall on groundwater levels.

Theory:

- SHAP values allow interpreting complex models like ANN by showing contribution of each feature to predictions.
- Partial dependence plots visualize marginal effect of a feature while averaging out others.

VIII. Discussion

Aspect	MLR	ANN (MLP)
Interpretability	High	Moderate (via SHAP)
Non-linearity Handling	Limited	Excellent
Overfitting Control	Simple	Dropout + Early Stopping
Computation Time	Fast	2.34s
Scalability	Moderate	High

Theory:

- MLR is simpler and interpretable but fails in capturing non-linear dependencies.
- ANN performs better due to its flexibility but requires careful tuning.

IX. Conclusion

- **ANN outperformed MLR**: $R^2 = 0.559$ vs 0.363.
- Improvement confirmed via **paired t-test** ($p<0.001$).
- **Key predictors**: lagged groundwater levels, soil texture (floamy), and rainfall.
- **Future Work:**
    - Implement **LSTM/GRU networks** for better temporal dependency capture.
    - Integrate **spatial validation** for regional generalization.

Theory:

- Accurate groundwater prediction supports efficient water resource management.
- Machine learning models, especially ANN, can handle complex, non-linear environmental relationships more effectively than traditional linear models.

Limitations of ANN:

While ANNs can be very powerful, in this case, several limitations led to their underperformance compared to the Multiple Linear Regression (MLR) model.
Small Dataset Size
The primary limitation here is the **small size of the dataset**. Our dataset has 954 total records. After splitting, the ANN was trained on only **566 samples** and validated on 101.
ANNs are "data-hungry" and typically require thousands or tens of thousands of records to effectively learn the complex, non-linear patterns they are designed to find.
With a small dataset, the ANN is prone to simply memorizing the training data (overfitting) without generalizing well, as evidenced by its poor test set performance (R2 of 0.16).