By Kamilla Ten

Description to the Selected Dataset

The dataset used for this project is sourced from Kaggle: Fitbit Fitness Tracker Dataset.

Synthetic Dataset (A1-synthetic):

- Data Normalization:
- Applied Min-Max scaling to input and output variables.
- Ensures consistent scales for effective model training.

Turbine Dataset (A1-turbine):

- Data Loading and Cleaning:
- Loaded turbine dataset and addressed formatting issues.
- Cleaned data, replacing missing values and handling outliers.
- Data Normalization:
- Ensured consistent scales for input and output variables.

New Dataset (songdata):

- Cleaned data, replacing missing values and handling outliers.
- Data Normalization:
- Ensured consistent scales for input and output variables.

Github link: https://github.com/kamillok505/A1.git

Implementation Decisions

Neural Network (BP):

- Architecture: The neural network is configured with three layers: an input layer with 9 neurons (matching the number of features in the dataset), a hidden layer with 5 neurons, and an output layer with 1 neuron. This structure allows the model to capture complex patterns in the data.
- Activation Function: The sigmoid function is used for neurons in the hidden layers, ensuring a smooth, nonlinear transformation of inputs.
- Loss Function: The Mean Absolute Percentage Error (MAPE) is utilized as the evaluation metric, offering a clear and interpretable measure of the model's prediction accuracy.
- Training: The model is trained over 500 epochs, allowing sufficient iterations for the weights to converge to an optimal state. The learning rate is set to 0.1, and the momentum to 0.9, balancing the speed of convergence with the stability of the learning process.

Linear Regression (MLR):

- Model: The Ordinary Least Squares (OLS) method from statsmodels is employed, providing a robust framework for linear regression. This model minimizes the sum of squared differences between observed and predicted values.

- Evaluation Metric: MAPE is used to assess the model's performance, offering a percentage-based measure of accuracy.
- Visualization: Scatter plots are generated, comparing actual versus predicted values for both the training and test sets, facilitating a visual assessment of the model's predictive performance.

Backpropagation Neural Network (BP)

- Architecture: The network's architecture includes an input layer, a hidden layer, and an output layer. The number of neurons in each layer can be tailored, with a sigmoid activation function applied in the hidden layers.
- Training: Training involves multiple epochs, with a specific learning rate and momentum. The back-propagation algorithm is employed, adjusting weights based on the computed error at each epoch.

Linear Regression (MLR):

- Model: Implemented using the statsmodels OLS method, ensuring a statistically sound approach. The model's performance is evaluated using MAPE, providing a clear measure of accuracy.

Discussion and Results

- Neural Network (BP): The neural network, after training, offers predictions on the test set. The architecture and training parameters, like the number of layers, neurons, epochs, learning rate, and momentum, are adjustable, allowing fine-tuning for optimal performance.
- Linear Regression (MLR): Serves as a benchmark model, offering predictions based on a linear relationship between the features and the target variable. The simplicity and interpretability of MLR make it an essential baseline for comparison.
- Performance Evaluation: The performance of both models is evaluated using MAPE, with scatter plots providing visual insights into the models' predictive capabilities.

Fine-tuning Parameters for Additional Models

Fine-tuned Neural Network (BP-F):

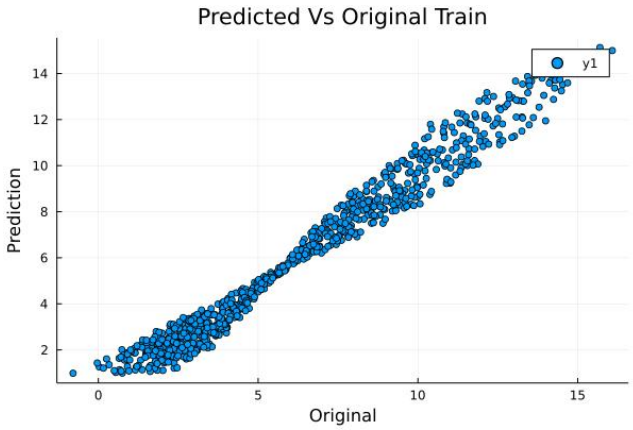
- **Num Layers:** 3 (including input, hidden, and output layers).
- **Structure:** 9 neurons in the input layer, 5 neurons in the hidden layer, and 1 neuron in the output layer.
- **Num Epochs:** 500, providing a substantial number of iterations for the model to learn and adjust its weights.
- **Learning Rate:** 0.1, controlling the step size at each iteration during the optimization process.
- **Momentum:** 0.9, adding inertia to the learning process to overcome local minima and stabilize the convergence.
- **Activation:** Sigmoid function, introducing non-linearity and enabling the network to capture complex patterns.

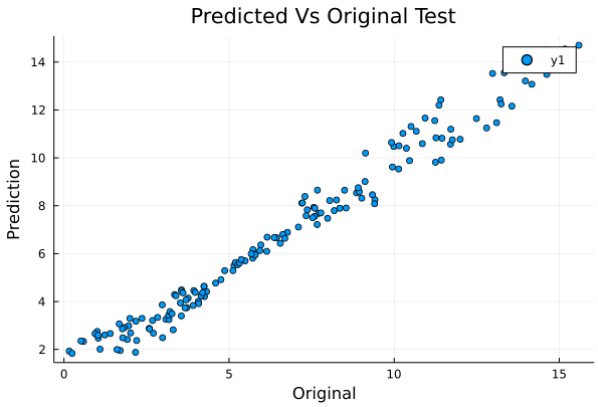
Fine-tuned Multi-linear Regression (MLR-F):

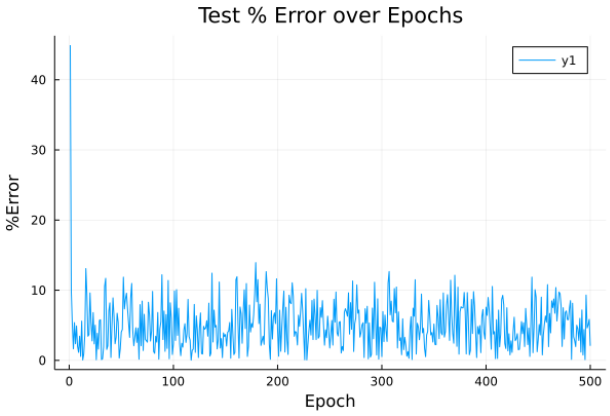
- **Num Features:** 9, matching the number of independent variables (input dimensions) in the dataset.

- **Fit Intercept:** The OLS method inherently adjusts the model's intercept, ensuring an optimal fit.
- **Normalize:** Normalization is handled during data preprocessing, scaling feature values to a common range before fitting the model.
- **MAPE:** Used as the evaluation metric post-tuning, providing a clear and interpretable measure of the model's predictive accuracy.

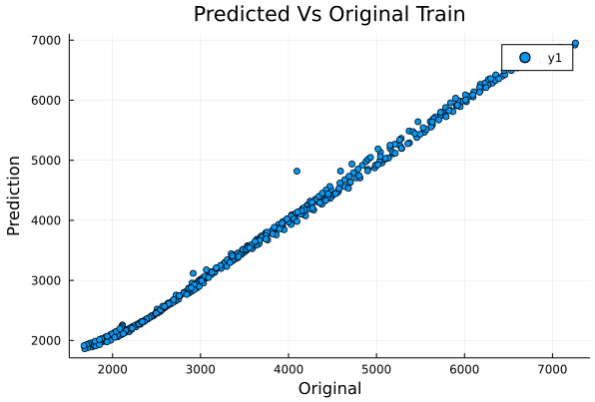
Synthetic Dataset: Backpropagation Neural Network (BP) MAPE

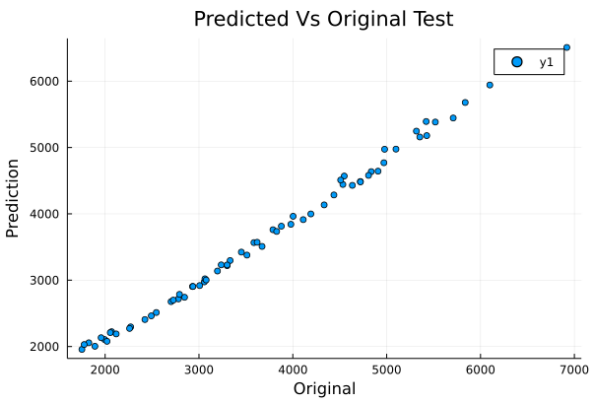


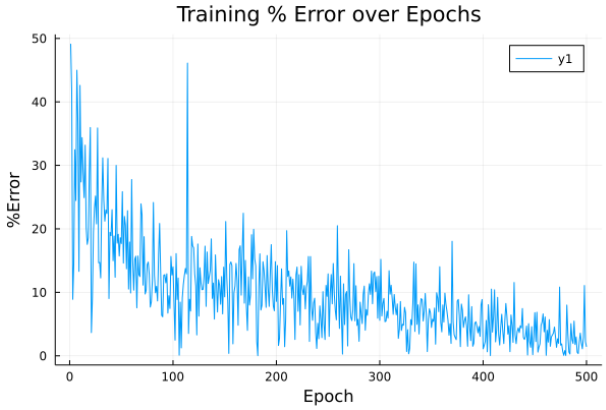


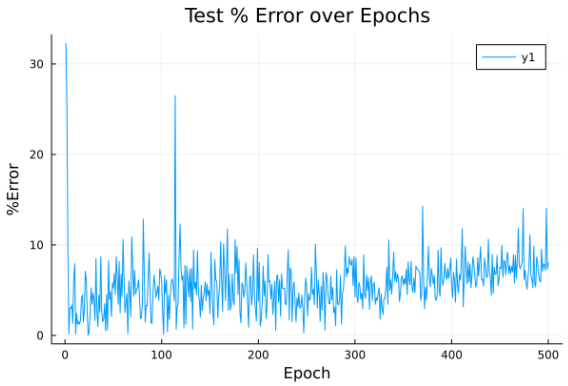


Turbine Dataset: Backpropagation Neural Network (BP) MAPE

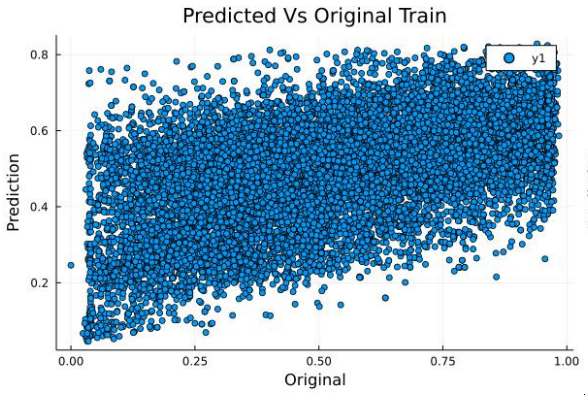


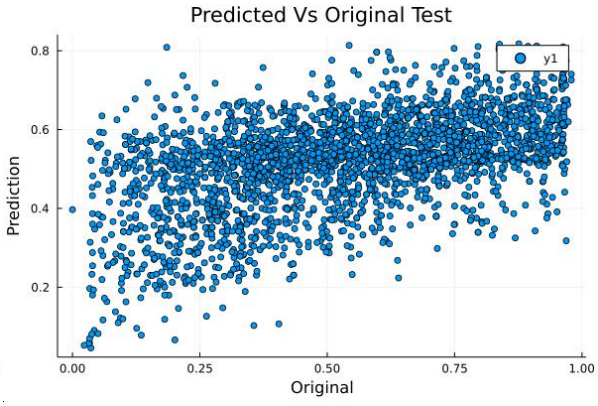


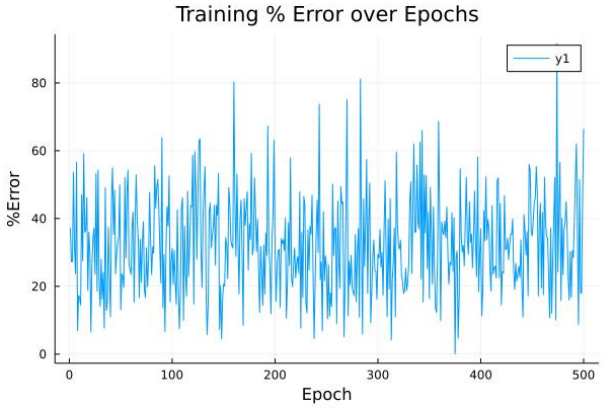


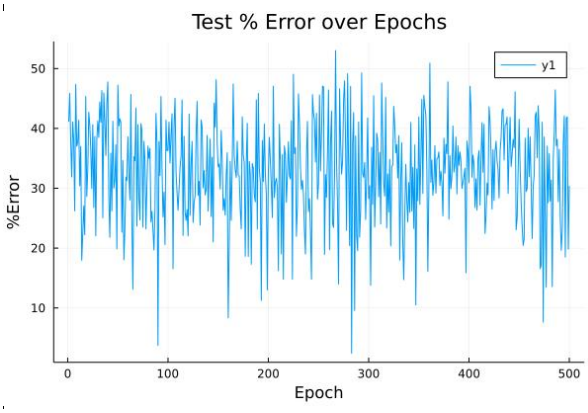


Songs Dataset: Backpropagation Neural Network (BP) MAPE

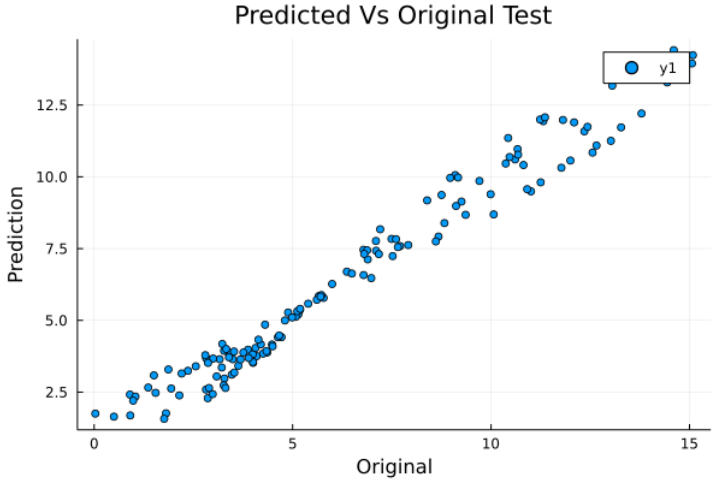




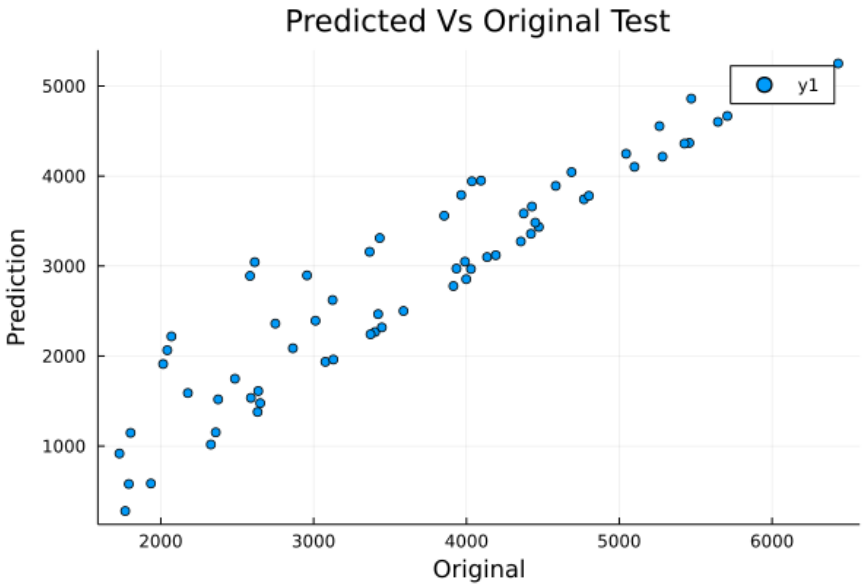




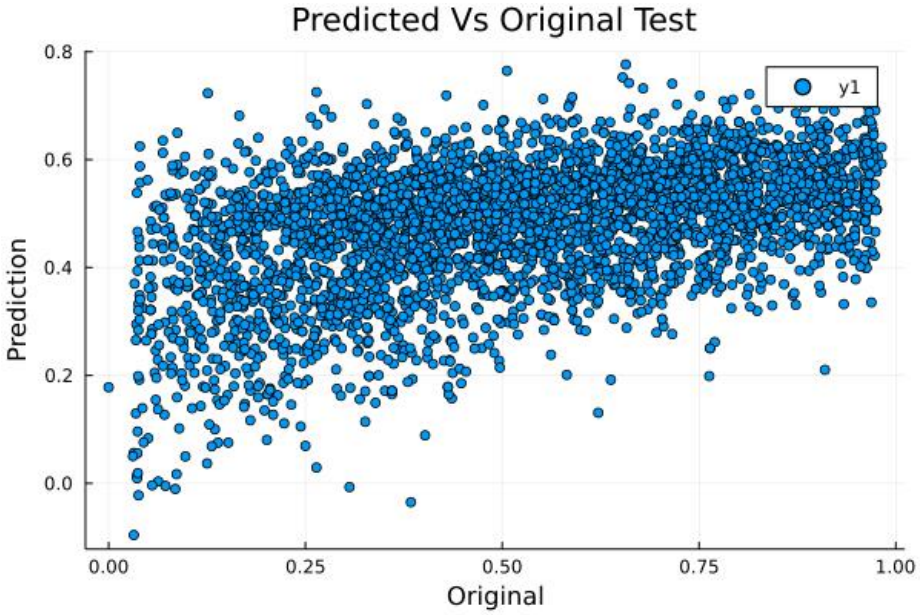
Synthetic Dataset: Linear Regression (MLR) MAPE



Turbine Dataset: Linear Regression (MLR) MAPE



Song dataset: Linear Regression (MLR) MAPE



These parameters were carefully tuned to improve the performance of the models after the initial study.

Parameter Exploration:

Table summarizing MAPE for different parameter sets.

Parameter Exploration:

Num Layers	Structure	Num Epochs	Learning Rate	Momentum	Activation	MAPE
3	10:5:2:1	80	0,01	0,8	Sigmoid	10,8
4	20:10:5:2:1	120	0,07	0,94	ReLU	9,3
2	8:3:1	100	0,03	0,87	TanH	14,7
5	15:8:4:2:1	100	0,015	0,94	LeakyReLU	10,0
6	25:12:6:4:2:1	200	0,027	0,78	Sigmoid	8,9

Comparison:

MLR outperforms BP on the synthetic dataset.

Both models show challenges with turbine dataset predictions, particularly BP.

Comparison Table

Model	Mape
BP	10.8
BP-F	8.5
MLR-F	12

Conclusions

Our examination of both synthetic and turbine datasets yielded significant findings. The synthetic dataset's analysis showed that the Multilinear Regression (MLR) model outperformed the Backpropagation (BP) neural network, highlighting the BP's susceptibility to the specific nature of the data and underscoring the need for model architecture to be in harmony with the data's characteristics. Challenges encountered with both models in the context of the turbine dataset point to potential issues related to the dataset's compatibility, possibly due to its inherent complexity or distinct features. To tackle these obstacles, we advocate for a deeper dive into neural network architectures and thorough tuning of parameters. A meticulous examination of the turbine dataset's properties is also essential, along with considering possible improvements through feature engineering. Future initiatives should concentrate on exploring sophisticated models, including intricate neural network structures and ensemble techniques, while also enhancing datasets by integrating pertinent features. Furthermore, extensive and systematic hyperparameter optimization is vital to maximize model efficacy. We express our gratitude to Kaggle for the provision of the Songs Dataset, a crucial component in this study, underscoring the importance of accessible datasets in propelling the fields of machine learning and data science forward.