The flight data contains different flight segments, noise levels, and behaviors. Working with aerial flight magnetic data, so:

1. Not all flight segments are equally useful.
    - Example: 1003.01 Takeoff and 1003.10 HOLD-OUT TESTING DATA might be excluded from training.
    - Others like Free-Fly or Climb may be cleaner, more consistent signal sources.
2. Each segment represents a different environment (e.g., low altitude, transit, climb, etc.) which impacts noise characteristics.

Parse Flight Segments First

Before you normalize or split data, filter your training set to include only specific segments (e.g., 1003.02 → 1003.09), and exclude:

- 1003.01 Takeoff
- 1003.10 HOLD-OUT (used later for evaluation)
- 1003.11 Transit to base (if not representative of noise profile)

We've gotten some basic information regarding the flight timings and their specifications - but I had a problem loading them. While we have 'tt' it didn't match.

We are also:

- Dropping outliers or obvious NaNs
- Removing extremely short segments

1. Feature Importance

Feature Importance is a technique used to determine which input features (variables) in a model are most influential for predicting the target variable. This can help identify which features are contributing the most to the model's predictions, allowing you to prioritize them and possibly remove irrelevant or redundant features, leading to a simpler and more interpretable model.

- Random Forests: Random Forest is a popular machine learning algorithm that can be used to calculate feature importance. It works by creating multiple decision trees and averaging their predictions. For each tree, it evaluates how well each feature splits the data, and this information can be aggregated to estimate the importance of each feature across all trees.

Evaluating all the features to see which ones hold the most importance/relevance.

-
- Evaluating with features: ['baro'] => MSE: 106994.939, R²: 0.184
- Evaluating with features: ['cur_ac_hi'] => MSE: 126606.672, R²: 0.034
- Evaluating with features: ['cur_ac_lo'] => MSE: 113259.619, R²: 0.136
- Evaluating with features: ['cur_acpwr'] => MSE: 130336.605, R²: 0.006
- Evaluating with features: ['cur_bat_1'] => MSE: 129725.716, R²: 0.010
- Evaluating with features: ['cur_bat_2'] => MSE: 130915.554, R²: 0.001
- Evaluating with features: ['cur_com_1'] => MSE: 121570.137, R²: 0.073
- Evaluating with features: ['cur_flap'] => MSE: 119449.430, R²: 0.089
- Evaluating with features: ['cur_heat'] => MSE: 124118.949, R²: 0.053
- Evaluating with features: ['cur_outpwr'] => MSE: 130418.030, R²: 0.005
- Evaluating with features: ['cur_srvo_i'] => MSE: 131061.557, R²: 0.000
- Evaluating with features: ['cur_srvo_m'] => MSE: 131106.053, R²: -0.000
- Evaluating with features: ['cur_srvo_o'] => MSE: 131038.826, R²: 0.000
- Evaluating with features: ['cur_strb'] => MSE: 115076.563, R²: 0.122
- Evaluating with features: ['cur_tank'] => MSE: 96162.971, R²: 0.266
- Evaluating with features: ['dem'] => MSE: 77153.143, R²: 0.411
- Evaluating with features: ['diurnal'] => MSE: 43959.499, R²: 0.665
- Evaluating with features: ['drape'] => MSE: 84890.894, R²: 0.352
- Evaluating with features: ['dt'] => MSE: 131092.754, R²: -0.000
- Evaluating with features: ['flight'] => MSE: 131092.754, R²: -0.000
- Evaluating with features: ['flux_a_t'] => MSE: 64161.056, R²: 0.511
- Evaluating with features: ['flux_a_x'] => MSE: 113665.922, R²: 0.133
- Evaluating with features: ['flux_a_y'] => MSE: 103639.485, R²: 0.209
- Evaluating with features: ['flux_a_z'] => MSE: 104665.549, R²: 0.201
- Evaluating with features: ['flux_b_t'] => MSE: 71739.173, R²: 0.453
- Evaluating with features: ['flux_b_x'] => MSE: 109659.145, R²: 0.163
- Evaluating with features: ['flux_b_y'] => MSE: 109109.549, R²: 0.168
- Evaluating with features: ['flux_b_z'] => MSE: 104007.522, R²: 0.207
- Evaluating with features: ['flux_c_t'] => MSE: 99129.663, R²: 0.244
- Evaluating with features: ['flux_c_x'] => MSE: 101653.822, R²: 0.224
- Evaluating with features: ['flux_c_y'] => MSE: 104342.918, R²: 0.204
- Evaluating with features: ['flux_c_z'] => MSE: 113587.540, R²: 0.133
- Evaluating with features: ['flux_d_t'] => MSE: 76878.831, R²: 0.413
- Evaluating with features: ['flux_d_x'] => MSE: 104212.203, R²: 0.205
- Evaluating with features: ['flux_d_y'] => MSE: 105420.360, R²: 0.196
- Evaluating with features: ['flux_d_z'] => MSE: 113335.917, R²: 0.135
- Evaluating with features: ['ins_acc_x'] => MSE: 129299.762, R²: 0.014
- Evaluating with features: ['ins_acc_y'] => MSE: 130768.359, R²: 0.002
- Evaluating with features: ['ins_acc_z'] => MSE: 129747.371, R²: 0.010
- Evaluating with features: ['ins_alt'] => MSE: 111014.931, R²: 0.153
- Evaluating with features: ['ins_lat'] => MSE: 23670.470, R²: 0.819
- Evaluating with features: ['ins_lon'] => MSE: 46363.892, R²: 0.646
- Evaluating with features: ['ins_pitch'] => MSE: 127401.034, R²: 0.028
- Evaluating with features: ['ins_roll'] => MSE: 129297.512, R²: 0.014
- Evaluating with features: ['ins_vn'] => MSE: 93447.862, R²: 0.287
- Evaluating with features: ['ins_vu'] => MSE: 130757.870, R²: 0.002
- Evaluating with features: ['ins_vw'] => MSE: 93933.918, R²: 0.283
- Evaluating with features: ['ins_wander'] => MSE: 46340.789, R²: 0.646
- Evaluating with features: ['ins_yaw'] => MSE: 87075.235, R²: 0.336
- Evaluating with features: ['lat'] => MSE: 23231.321, R²: 0.823
- Evaluating with features: ['lgtl_acc'] => MSE: 126420.124, R²: 0.036
- Evaluating with features: ['line'] => MSE: 59916.283, R²: 0.543
- Evaluating with features: ['lon'] => MSE: 46329.770, R²: 0.647
- Evaluating with features: ['ltrl_acc'] => MSE: 128315.094, R²: 0.021
- Evaluating with features: ['mag_1_c'] => MSE: 240.808, R²: 0.998
- Evaluating with features: ['mag_1_dc'] => MSE: 286.748, R²: 0.998
- Evaluating with features: ['mag_1_igrf'] => MSE: 22187.863, R²: 0.831
- Evaluating with features: ['mag_1_lag'] => MSE: 249.335, R²: 0.998
- Evaluating with features: ['mag_1_uc'] => MSE: 243.457, R²: 0.998
- Evaluating with features: ['mag_2_uc'] => MSE: 77824.195, R²: 0.406
- Evaluating with features: ['mag_3_uc'] => MSE: 92505.847, R²: 0.294
- Evaluating with features: ['mag_4_uc'] => MSE: 47119.309, R²: 0.641
- Evaluating with features: ['mag_5_uc'] => MSE: 9899.215, R²: 0.924
- Evaluating with features: ['msl'] => MSE: 110962.038, R²: 0.153
- Evaluating with features: ['nrml_acc'] => MSE: 129615.380, R²: 0.011
- Evaluating with features: ['ogs_alt'] => MSE: 90920.992, R²: 0.306
- Evaluating with features: ['ogs_mag'] => MSE: 5660.868, R²: 0.957
- Evaluating with features: ['pitch_rate'] => MSE: 129636.709, R²: 0.011
- Evaluating with features: ['pitot_p'] => MSE: 128538.503, R²: 0.019
- Evaluating with features: ['radar'] => MSE: 81841.807, R²: 0.376
- Evaluating with features: ['roll_rate'] => MSE: 130146.953, R²: 0.007
- Evaluating with features: ['static_p'] => MSE: 107128.028, R²: 0.183
- Evaluating with features: ['tas'] => MSE: 129078.407, R²: 0.015
- Evaluating with features: ['topo'] => MSE: 80919.327, R²: 0.383
- Evaluating with features: ['total_p'] => MSE: 111396.141, R²: 0.150
- Evaluating with features: ['tt'] => MSE: 18021.446, R²: 0.863
- Evaluating with features: ['utm_x'] => MSE: 46635.094, R²: 0.644
- Evaluating with features: ['utm_y'] => MSE: 22802.319, R²: 0.826
- Evaluating with features: ['utm_z'] => MSE: 110914.999, R²: 0.154
- Evaluating with features: ['vol_acc_n'] => MSE: 126596.962, R²: 0.034
- Evaluating with features: ['vol_acc_p'] => MSE: 128145.214, R²: 0.022
- Evaluating with features: ['vol_acpwr'] => MSE: 129423.569, R²: 0.013
- Evaluating with features: ['vol_back'] => MSE: 130894.462, R²: 0.001
- Evaluating with features: ['vol_back_n'] => MSE: 113499.609, R²: 0.134
- Evaluating with features: ['vol_back_p'] => MSE: 114442.815, R²: 0.127
- Evaluating with features: ['vol_bat_1'] => MSE: 68316.595, R²: 0.479
- Evaluating with features: ['vol_bat_2'] => MSE: 117852.267, R²: 0.101
- Evaluating with features: ['vol_block'] => MSE: 128958.715, R²: 0.016
- Evaluating with features: ['vol_cabt'] => MSE: 130464.848, R²: 0.005
- Evaluating with features: ['vol_fan'] => MSE: 119742.742, R²: 0.086
- Evaluating with features: ['vol_gyro_1'] => MSE: 117818.528, R²: 0.101
- Evaluating with features: ['vol_gyro_2'] => MSE: 126178.471, R²: 0.037
- Evaluating with features: ['vol_outpwr'] => MSE: 129697.132, R²: 0.011
- Evaluating with features: ['vol_res_n'] => MSE: 124444.099, R²: 0.051
- Evaluating with features: ['vol_res_p'] => MSE: 122710.952, R²: 0.064
- Evaluating with features: ['vol_srvo'] => MSE: 118182.222, R²: 0.098
- Evaluating with features: ['yaw_rate'] => MSE: 130105.770, R²: 0.007

We are focusing on those features that have the best balance between predictive power and real-world relevance. From the results you've shared, we can assess which features might be more valuable for your model.

Key Features to Consider:

1. Magnetic Features (mag_1_c, mag_1_dc, mag_1_lag, etc.):
    - These features have extremely high R^2 values close to 1, indicating that they explain a very large portion of the variance in the data. These features are highly correlated with the target variable.
2. Geospatial Features (lat, lon, utm_x, utm_y):
    - These features also have relatively high R^2 values, especially lat and utm_y (around 0.823 and 0.826). This suggests that the geographic location of your observations has a strong relationship with the outcome you're trying to predict.
    - Usage: Data has a geographical component (such as altitude or geographic positioning), these features should be retained. In many environmental or geophysical problems, geographic data can explain large portions of the variation.
3. Flight Dynamics Features (ins_lat, ins_lon, ins_acc_x, etc.):
    - Some flight dynamics features like ins_lat and ins_lon have high R^2 values (e.g., ins_lat has 0.819 R^2). These features seem to be related to the position and orientation of the system, which may be important if the model is related to tracking, navigation, or flight performance.

Final Feature Set Recommendation:

1. Primary Features to keep:
    - mag_1_c, mag_1_dc, mag_1_lag, mag_1_uc (High R^2 values)
    - lat, lon, utm_x, utm_y (Geospatial data with strong correlation)
    - ins_lat, ins_lon, ins_acc_x (Flight dynamics with moderate to high R^2 values)

The cleaning process should be separate for each group because:

- Each group captures different physical phenomena
- They have different error sources (e.g., GPS drift ≠ magnetic noise)
- They need different preprocessing strategies (e.g., Kalman filter for INS, rolling mean for mag, haversine calc for lat/lon)

1. Magnetometer Data

- Applying a **Savitzky-Golay filter** to mag_1_uc to smooth out high-frequency noise while preserving the signal shape.
- For mag_1_lag, a **rolling mean** with a window of 5 is used to smooth sudden variations.
- **Goal:** Reduce sensor noise and keep underlying magnetic trends clear.

2. Geolocation (GPS) Data

- Missing values in lat and lon are filled using **linear interpolation**.
- Differences (lat_diff and lon_diff) are calculated to estimate motion between points.
- **Goal:** Create a continuous and smooth GPS path, essential for calculating velocity or direction.

3. INS (Inertial Navigation System)

- Applying a **rolling average** to smooth out the ins_acc_x signal.
- **Goal:** Reduce jitter and drift in accelerometer data, which helps better capture real movement.

- The code is meant to clean flight sensor data and use a machine learning model to predict a magnetic field value (`mag_1_c`).
- It starts by importing necessary tools for handling data, math, machine learning, and plotting graphs.
- There are several **data cleaning functions**:
    - For **magnetometer data**, it smooths out noise using techniques like rolling average and Savitzky-Golay filter.
    - For **geolocation data** (latitude and longitude), it fills in missing values and calculates how much the drone has moved.
    - For **INS data** (acceleration from internal sensors), it smooths out the values to reduce noise.
- All these cleaning steps are combined into one `preprocess()` function that prepares the full dataset.
- After cleaning, it extracts important columns from the data to use as **features** (inputs for the model), and it picks the **target value** the model is supposed to predict (`mag_1_c`).
- The main script then:
    - Loads raw flight data from a `.h5` file.
    - Cleans the data using the defined cleaning functions.
    - Extracts the features and target values.
    - Loads saved scalers which were used during model training to scale the input and output data.
    - Applies the same scaling to the new data to keep it consistent with the model's training.
    - Loads a pre-trained machine learning model that was previously saved.
    - Uses this model to predict what the magnetic value should be.
    - Converts the scaled prediction back to original units.
    - Measures how accurate the predictions are using standard error metrics.
    - Prints out a few predicted and true values for comparison.

The model

Epoch 1/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6s 1ms/step - loss: 0.0097 - val_loss: 5.0173e-04

Epoch 2/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8s 2ms/step - loss: 3.1984e-04 - val_loss: 1.1061e-04

Epoch 3/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 1.8968e-04 - val_loss: 2.1261e-04

Epoch 4/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 1.3597e-04 - val_loss: 4.3255e-05

Epoch 5/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 1.2459e-04 - val_loss: 7.9963e-05

Epoch 6/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 1.1392e-04 - val_loss: 3.4726e-05

Epoch 7/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 1.0166e-04 - val_loss: 8.8936e-05

Epoch 8/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 1.0761e-04 - val_loss: 4.3398e-04

Epoch 9/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 1.2224e-04 - val_loss: 5.9865e-05

Epoch 10/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 8.7074e-05 - val_loss: 6.8289e-05

Epoch 11/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 7.5455e-05 - val_loss: 5.4168e-05

Epoch 12/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 8.1077e-05 - val_loss: 1.1349e-04

Epoch 13/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 8.4633e-05 - val_loss: 5.5988e-05

Epoch 14/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 5.7847e-05 - val_loss: 1.3635e-05

Epoch 15/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 6.6822e-05 - val_loss: 1.0602e-04

Epoch 16/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 6.4024e-05 - val_loss: 3.7032e-05

Epoch 17/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 5.4513e-05 - val_loss: 2.3425e-05

Epoch 18/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 6.9205e-05 - val_loss: 1.8504e-05

Epoch 19/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 7.5681e-05 - val_loss: 2.7142e-05

Epoch 20/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 5.4852e-05 - val_loss: 7.6608e-05

Epoch 21/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 6.5962e-05 - val_loss: 4.5895e-05

Epoch 22/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 4.6525e-05 - val_loss: 1.5894e-05

Epoch 23/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 1.0255e-04 - val_loss: 2.3568e-05

Epoch 24/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 4.3769e-05 - val_loss: 1.2972e-04

Epoch 25/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 6.6415e-05 - val_loss: 1.2215e-04

Epoch 26/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 6.9135e-05 - val_loss: 4.6957e-05

Epoch 27/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 6.1874e-05 - val_loss: 5.4412e-06

Epoch 28/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 6.7774e-05 - val_loss: 5.3403e-05

Epoch 29/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 4.5296e-05 - val_loss: 6.4440e-06

Epoch 30/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 3.2388e-05 - val_loss: 1.1883e-05

Epoch 31/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 4.7800e-05 - val_loss: 2.2995e-04

Epoch 32/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 5.0166e-05 - val_loss: 1.1269e-05

Epoch 33/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 4.8673e-05 - val_loss: 2.2728e-05

Epoch 34/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 4.8559e-05 - val_loss: 5.3661e-05

Epoch 35/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 5.4760e-05 - val_loss: 1.2987e-04

Epoch 36/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 5.6534e-05 - val_loss: 8.8144e-06

Epoch 37/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 4.4580e-05 - val_loss: 1.6372e-04

Epoch 38/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 6.4039e-05 - val_loss: 2.1692e-05

Epoch 39/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 6.5888e-05 - val_loss: 1.9798e-05

Epoch 40/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 5.2137e-05 - val_loss: 7.3287e-06

Epoch 41/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 3.4588e-05 - val_loss: 9.9722e-06

Epoch 42/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 2.8579e-05 - val_loss: 5.9826e-05

Epoch 43/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 4.7995e-05 - val_loss: 9.7346e-05

Epoch 44/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 4.2054e-05 - val_loss: 8.0080e-06

Epoch 45/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 3.9734e-05 - val_loss: 9.2902e-05

Epoch 46/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 3.6812e-05 - val_loss: 1.0081e-05

Epoch 47/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 4.0109e-05 - val_loss: 1.5071e-05

Epoch 48/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 3.6058e-05 - val_loss: 1.0170e-04

Epoch 49/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 3.7569e-05 - val_loss: 4.6010e-05

Epoch 50/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 4.7298e-05 - val_loss: 3.2248e-05

Epoch 51/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 3.5549e-05 - val_loss: 1.9797e-05

Epoch 52/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 4.7458e-05 - val_loss: 1.4280e-04

Epoch 53/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 4.6987e-05 - val_loss: 9.6025e-06

Epoch 54/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2.8424e-05 - val_loss: 1.3466e-05

Epoch 55/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2.6963e-05 - val_loss: 3.1321e-05

Epoch 56/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 3.4623e-05 - val_loss: 9.8758e-06

Epoch 57/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 3.0849e-05 - val_loss: 1.0188e-05

Epoch 58/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 2.5656e-05 - val_loss: 1.6023e-05

Epoch 59/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2.6042e-05 - val_loss: 2.1398e-05

Epoch 60/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2.5479e-05 - val_loss: 1.1252e-05

Epoch 61/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 15s 1ms/step - loss: 3.6143e-05 - val_loss: 6.5502e-05

Epoch 62/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2.7391e-05 - val_loss: 7.0377e-06

Epoch 63/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 3.3273e-05 - val_loss: 3.6813e-05

Epoch 64/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2.9502e-05 - val_loss: 1.7433e-05

Epoch 65/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 2.7966e-05 - val_loss: 1.4992e-05

Epoch 66/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 4.1148e-05 - val_loss: 2.1589e-05

Epoch 67/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 3.7241e-05 - val_loss: 7.4673e-06

Epoch 68/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 2.6851e-05 - val_loss: 1.4206e-05

Epoch 69/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 14s 3ms/step - loss: 2.7425e-05 - val_loss: 1.8150e-05

Epoch 70/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 2.3704e-05 - val_loss: 1.4242e-05

Epoch 71/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8s 2ms/step - loss: 2.8438e-05 - val_loss: 8.5979e-06

Epoch 72/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 2.6661e-05 - val_loss: 1.9868e-05

Epoch 73/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 3.0172e-05 - val_loss: 3.4909e-05

Epoch 74/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 2ms/step - loss: 3.0959e-05 - val_loss: 1.9787e-05

Epoch 75/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 3.3494e-05 - val_loss: 2.0119e-05

Epoch 76/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 2.3667e-05 - val_loss: 1.2610e-05

Epoch 77/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 9s 2ms/step - loss: 2.3086e-05 - val_loss: 3.8762e-04

Epoch 78/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 3.7180e-05 - val_loss: 2.4900e-05

Epoch 79/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 18s 4ms/step - loss: 3.0104e-05 - val_loss: 1.4701e-05

Epoch 80/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 3.3553e-05 - val_loss: 2.2046e-05

Epoch 81/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 19s 2ms/step - loss: 2.4616e-05 - val_loss: 1.1201e-04

Epoch 82/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 13s 3ms/step - loss: 2.6325e-05 - val_loss: 3.1143e-05

Epoch 83/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 2.7210e-05 - val_loss: 6.3816e-05

Epoch 84/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 3.2530e-05 - val_loss: 6.8043e-05

Epoch 85/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 3.3648e-05 - val_loss: 4.8534e-06

Epoch 86/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 2.3534e-05 - val_loss: 6.5216e-05

Epoch 87/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 22s 5ms/step - loss: 2.9039e-05 - val_loss: 3.4027e-05

Epoch 88/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 19s 5ms/step - loss: 2.7359e-05 - val_loss: 1.1564e-05

Epoch 89/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 2.5828e-05 - val_loss: 9.2429e-06

Epoch 90/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 2.9264e-05 - val_loss: 2.3166e-05

Epoch 91/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 14s 3ms/step - loss: 2.2407e-05 - val_loss: 1.4403e-05

Epoch 92/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 2.9553e-05 - val_loss: 1.7771e-05

Epoch 93/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 18s 5ms/step - loss: 2.4107e-05 - val_loss: 1.5114e-05

Epoch 94/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 20s 5ms/step - loss: 3.1928e-05 - val_loss: 1.3761e-05

Epoch 95/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - loss: 2.3176e-05 - val_loss: 2.5819e-05

Epoch 96/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 20s 5ms/step - loss: 2.7396e-05 - val_loss: 7.4003e-06

Epoch 97/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - loss: 2.7415e-05 - val_loss: 2.5269e-05

Epoch 98/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12s 3ms/step - loss: 2.8844e-05 - val_loss: 8.3137e-06

Epoch 99/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 13s 3ms/step - loss: 2.6126e-05 - val_loss: 1.6307e-05

Epoch 100/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2.7845e-05 - val_loss: 1.0116e-05

Epoch 99/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 13s 3ms/step - loss: 2.6126e-05 - val_loss: 1.6307e-05

Epoch 100/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2.7845e-05 - val_loss: 1.0116e-05

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 13s 3ms/step - loss: 2.6126e-05 - val_loss: 1.6307e-05

Epoch 100/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2.7845e-05 - val_loss: 1.0116e-05

1001/1001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1s 1ms/step

Epoch 100/100

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2.7845e-05 - val_loss: 1.0116e-05

1001/1001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1s 1ms/step

4001/4001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - loss: 2.7845e-05 - val_loss: 1.0116e-05

1001/1001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1s 1ms/step

1001/1001 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1s 1ms/step

Min of true target: 52948.136

Min of true target: 52948.136

Max of true target: 55282.793

Mean of true target: 53711.41069024558

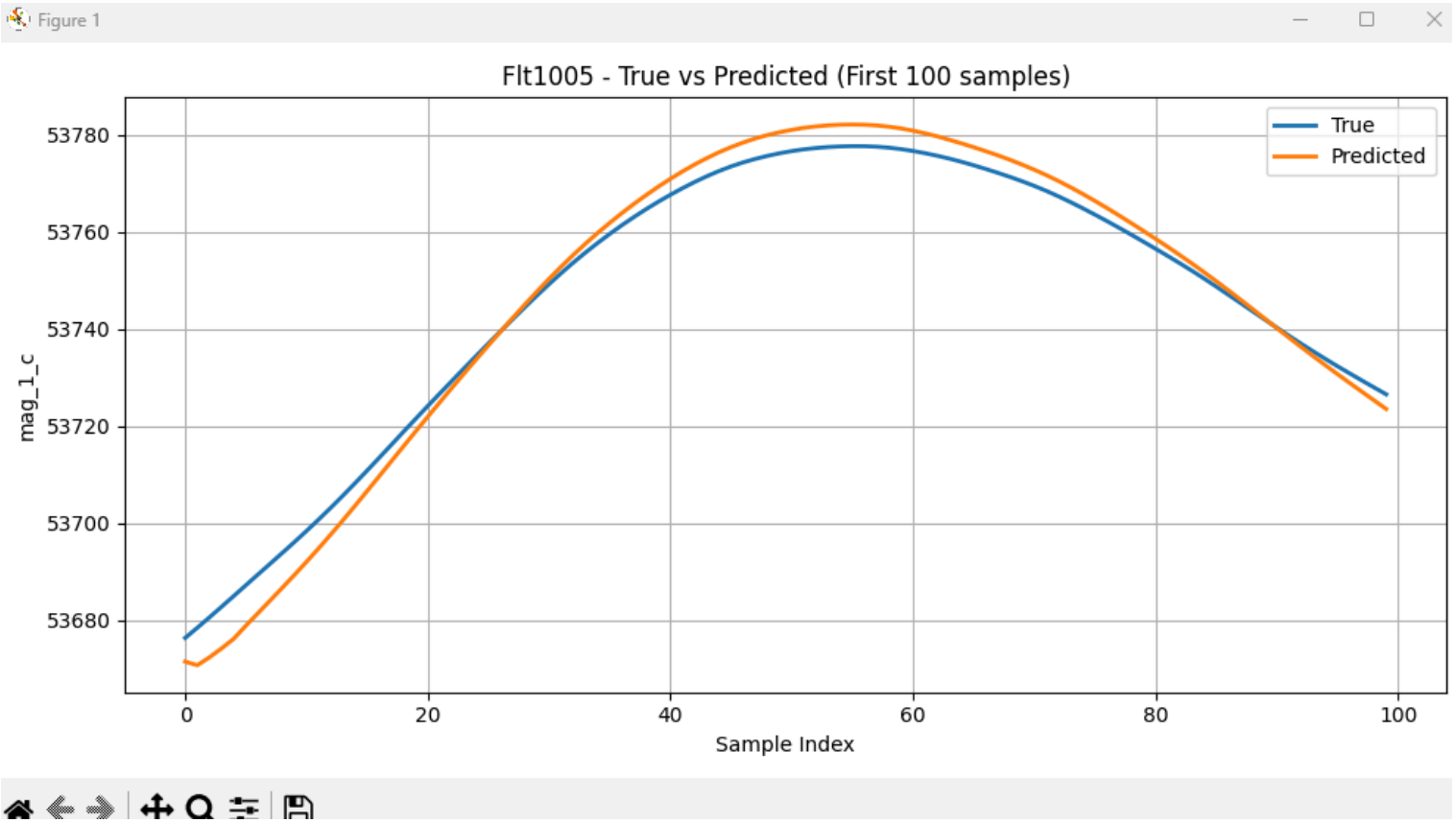
First 10 true target values:

[54065.011 53319.792 54276.271 53414.537 54065.047 53429.162 53830.532

53838.061 53652.887 54055.692]

Inverse Transformed Test MSE: 1.3289

WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model(model)`. This file format is considered legacy. We recommend using instead the native Keras format, e.g. `model.save('my_model.keras')` or `keras.saving.save_model(model, 'my_model.keras')`.



Interpretation of Results:

The model was trained to predict a target variable whose values range as follows:

- Minimum value: 52,948.136
- Maximum value: 55,282.793
- Mean value: 53,711.4107

The reported Inverse Transformed Test Mean Squared Error (MSE) is 1.3289, which translates to a Root Mean Squared Error (RMSE) of approximately 1.1532 (since RMSE = √MSE).

This implies that the model's average prediction deviates from the true target value by about 1.15 units on the original (inverse-transformed) scale.

Percentage Error:

To express the error relative to the scale of the data, we compute the RMSE as a percentage of the mean target value:

Percentage Error = 1.15/53711.41 times 100 = approx 0.0021 %

This extremely low percentage error indicates that the model's predictions are highly accurate in relation to the magnitude of the target variable.

Conclusion:

The very low RMSE and corresponding percentage error suggest that the model can reconstruct or predict the target variable with minimal deviation. This level of accuracy indicates that the model has likely captured the underlying patterns in the data effectively, with negligible overfitting or underfitting evident in the evaluation.

Everytime we test and train it we get different results

It’ll depend on the number of epochs we’re training it for - for the parameters of the training model is, if we have it dropout at any stage.

MLP Architecture (Multilayer Perceptron)

- Input Layer:
  Accepts as many neurons as there are input features. Each neuron corresponds to one feature of the magnetometer data.
- Hidden Layer 1:
  50 neurons using the tanh activation function.
  This allows modeling of non-linear relationships in the data.
- Hidden Layer 2:
  30 neurons with tanh activation again — further refining learned features.
- Hidden Layer 3:
  10 neurons with tanh activation — allowing the model to extract deeper patterns.
- Output Layer:
  Just 1 neuron, since we're performing a regression task (predicting a continuous value). No activation is used here (linear output).

Compilation

- Loss function: Mean Squared Error (MSE), standard for regression tasks.
- Optimizer: Adam — an adaptive gradient descent method

Training

The model is trained on the training data:

- For 100 epochs (iterations through the dataset).
- With a batch size of 32, meaning 32 samples are used at a time to update model weights.
- During training, it evaluates performance on the validation (test) set.

This helps monitor for overfitting or underfitting. -> we don't have either right now

In our case some weights were going in the negative - especially when we were checking what features would work best -> so we're going with the tanh activation.