

Electric Motor PM Temperature Prediction (Flask + ML)

Small Flask app and training scripts for predicting PM (Permanent Magnet) temperature from motor sensor inputs using Machine Learning.

Repository Layout:

- app.py - Flask web application
- train_model.py - Simple training script
- train_model_eval.py - Training with evaluation
- requirements.txt - Dependencies
- data/motor_data.csv - Dataset
- model/decision_tree_model.pkl - Saved model
- model/scaler.pkl - Saved scaler
- templates/Manual_predict.html - Manual input page
- templates/Sensor_predict.html - Sensor page

Setup:

```
pip install -r requirements.txt
```

Training (Quick):

```
python train_model.py
```

Training (Detailed):

```
python train_model_eval.py
```

Features Used (Order Matters):

1. ambient
2. coolant
3. u_d
4. u_q
5. motor_speed
6. i_d
7. i_q
8. stator_yoke
9. stator_winding

Target:

pm (Permanent Magnet Temperature)

Running the Web App:

```
python app.py
```

Open <http://127.0.0.1:5000>

Flask Routes:

- GET / - Manual input page
- POST /predict - Prediction endpoint
- GET /sensor - Sensor page

Prediction Notes:

- Inputs are scaled before prediction
- Model and scaler loaded at startup
- Ensure feature order matches training

Recommended Improvements:

- Unify training scripts
- Add input validation
- Improve UI with units and limits
- Add logging and tests
- Use debug=False for production

Troubleshooting:

- Verify model/scaler path
- Ensure 9 input features
- Check dataset location