Milestone 5: Automated Testing & Validation

# Objective

To implement automated tests for the PM2.5 prediction API to ensure reliable behavior under both valid and invalid inputs, and to validate the integration of model serving via FastAPI.

# Test Setup

- Framework: pytest

- Tools: FastAPI TestClient, Pydantic for input validation

- Environment: Python 3.13, Poetry, Windows

# Test Cases

1. 1. Valid Input Test

* - Sends a POST request with properly structured PM2.5 features.
* - Expects HTTP 200 and a float prediction in the response.

1. 2. Invalid Input Test

* - Sends a POST request with an invalid type (string instead of float).
* - Expects HTTP 422 Unprocessable Entity due to failed validation.

# Results

Both test cases passed successfully.

One non-critical warning was observed from scikit-learn regarding missing feature names in the test data, which does not affect correctness.

# Conclusion

Automated testing was successfully integrated to validate the core functionality of the PM2.5 prediction API. The test suite confirms that the endpoint correctly processes valid input and rejects malformed requests. This marks a key milestone toward production-readiness and deployability of the model-backed forecasting service.