What does `evaluate.py` do?

- Loads the trained LSTM Autoencoder
- Runs inference on all test files (F1L to F7M + healthy)
- Computes MSE for every sliding window (200 samples)
- Applies threshold (p95 from healthy validation)
- Flags windows with MSE ¿ threshold to ANOMALY

Outputs

File	Content
`metrics_*.csv`	Mean MSE, p95, % anomalies per fault
`per_sample_*.csv`	MSE + is_anomaly for each window
`results_summary.xlsx`	Full table

Good Results?

Metric	Good	Excellent
Healthy (F0)	<10%	<5%
Severe fault	>70%	>90%
Subtle fault	>20%	>50%

Results of Pipeline: `none_minmax_butterworth_none_Slidingwindow_`

Fault	Mean MSE	% Anomalies	Interpretation
F1L (LPPT)	7,335.7	**60.4%**	Strong
F1M (MPPT)	6,353.6	0.0%	Not detected
F2L (LPPT)	7,307.1	39.6%	Moderate
F3L (LPPT)	7,643.4	**73.5%**	Strong
F3M (MPPT)	7,352.5	4.3%	Weak
F4L	6,987.6	0.0%	Invisible
F5L	6,914.4	29.8%	Moderate
F6L	7,371.7	**92.3%**	**EXCELLENT**
F6M	7,328.6	44.3%	Good
F7L	7,349.9	72.0%	Strong
F7M	7,240.9	0.0%	Not detected
Healthy LPPT	7,306.7	8.1%	Acceptable
Healthy MPPT	7,250.8	0.0%	Perfect

Threshold = 7,335.63 (p95 on validation)

Discussion

- **Model works well**: F6L detected in 92% of windows to clear fault
- **Best on LPPT faults**: F6L, F3L, F7L >70%
- **MPPT faults (M) weak**: likely too mild or compensated
- **Per-sample files**: gradual MSE rise to early detection possible

Conclusion

Unsupervised detection is effective for severe faults.
F6L, F3L, F7L reliably detected.
MPPT faults need better modeling or multi-signal input.