

Testing checklist: machine learning model service

Source: <https://howtotestaproduct.com/how-to-test/ml-model-service/>

Generated: 2026-10-07. Licence: CC BY 4.0.

☐ 1. release smoke tests

Done when: After each model or code deploy, a fixed set of reference inputs returns the stored expected scores and labels from the live endpoint.

Risks: wrong-decisions. Tool class: API client. Usually run by: developer.

☐ 2. data integrity and migration testing

Done when: On a held-out set labelled by people, accuracy and false-positive results are recorded per group and per input quality, and the release record shows them.

Risks: wrong-decisions, user-harm, regulatory-penalty. Tool class: data validation. Usually run by: QA engineer.

☐ 3. exploratory testing

Done when: A session log covers low-quality, edge and adversarial inputs, records each false match found and links it to a filed issue.

Risks: user-harm, wrong-decisions. Tool class: test management. Usually run by: QA engineer.

☐ 4. analytics and event tracking tests

Done when: Production logs record each score, threshold and the action taken, and a weekly review compares a reviewed sample against the model output.

Risks: wrong-decisions, user-harm. Tool class: data validation. Usually run by: product manager.

☐ 5. unit tests

Done when: In the test report, every missing, out-of-range and low-quality sample input gets its documented value, and every sample score comes back with at least one reason.

Risks: wrong-decisions, regulatory-penalty. Tool class: unit framework. Usually run by: developer.

☐ 6. regression suite

Done when: Each new model version is scored on the same held-out set as the live one, and a worse result for any group blocks the release.

Risks: wrong-decisions, regulatory-penalty. Tool class: test management. Usually run by: QA engineer.

☐ 7. compliance checks

Done when: Each score returned for a credit or screening use carries the top reasons in plain words, the model record lists data sources and tested groups, and a sample of prediction logs holds no raw personal fields.

Risks: regulatory-penalty, data-breach. Tool class: test management. Usually run by: product manager.

☐ 8. security scanning

Done when: The scan of the endpoint, logs and training buckets shows no public access and no raw personal fields outside approved stores.

Risks: data-breach. Tool class: DAST scanner. Usually run by: security specialist.

☐ 9. penetration testing

Done when: A tester with one client key cannot read another client's predictions, logs or training data through the API.

Risks: data-breach. Tool class: proxy. Usually run by: security specialist.