AI-First Workflow Guidelines

In the AI era, the most successful people will not be the "smartest," but those who consistently ask the best questions. The purpose of this AI-First workflow is not to reduce your workload; it is to teach you how to use AI well, and real learning still requires effort. If this process feels effortless, that is a warning sign that you are outsourcing thinking rather than developing skill. AI will make many people dumber by encouraging passive acceptance of polished answers—your goal is to use AI to make yourself smarter by demanding evidence, testing assumptions, and refining your questions. Along the way, you must also guard against **p-hacking:** the practice of trying many analyses, models, feature sets, or data slices until something "looks significant," and then presenting that result as if it were the original plan.

Compared to "human-only" workflows, AI changes three things:

1. **Speed:** you can generate many options quickly (models, features, writeups).
2. **Risk:** AI can confidently produce incorrect code, incorrect claims, and fabricated citations.
3. **Complexity:** AI can expand the project faster than you can validate it.

Two non-negotiables for this course:

- **Baseline model required** (a simple benchmark you must beat and interpret).
- **Oral defense required** (you must be able to explain and justify what you submit).

Rules and Checklists (How to Succeed)

1) Quality control of AI output (accuracy + "runs on my machine" reproducibility)

What changes vs human-only

- In human-only work, most mistakes come from coding bugs or misunderstanding concepts.
- In AI-assisted work, mistakes also come from **plausible nonsense** (hallucinations) and **hidden pipeline errors** (especially leakage).

Your quality standard

You are responsible for: **correctness, interpretation, and integrity** of all results—regardless of whether AI suggested them.

Required quality habits

- **Treat AI output as a hypothesis.** Verify before you trust.
- **Make baselines first.** If a complex model doesn't clearly improve over baseline, pause and reassess.
- **One change at a time.** Implement one AI suggestion, rerun, compare, then proceed.

QC Checklist (turn this in as an appendix to your report.)

Data and claims

☐ I can explain what each column means (units, encoding, missing values).
☐ Any dataset statistics in the report are produced by code (not guessed).
☐ Any citations were opened and verified (no AI-generated fake references).

Splits and leakage

☐ Train/test split is appropriate (stratified if classification; time-split if temporal).
☐ Preprocessing is fit on training data only (e.g., scaling/imputation/encoding).
☐ I ran at least one leakage sanity check (e.g., shuffle labels → performance collapses).

Models and evaluation

☐ Baseline model included + metric reported.
☐ Primary metric matches the task (and I can define it).
☐ I inspected errors, not just a single score (examples of misclassifications / residuals / cluster exemplars).

Reproducibility ("runs on my machine")

☐ A teammate can run the project on the team's machine following the README.
☐ Dependencies documented (Python version + requirements).
☐ Random seeds set and noted (even if results vary slightly).

2) Academic integrity in AI-assisted workflows

Definition (tool-agnostic)

Academic integrity in this course means:

- **Transparency:** you disclose how AI was used.
- **Understanding:** you can explain what you submit.
- **Verification:** you check AI outputs for correctness.
- **Human ownership:** you make and justify key decisions.
- **Attribution:** you credit sources (datasets, code, and AI tools).

Disclosure requirement: You must include an **AI Use Appendix** in your final report describing how AI was used.

Integrity “Do / Don’t” (practical rules)

DO

- Disclose AI use in the report appendix (what you used it for, and what you verified).
- Make sure you completely understand every word, sentence and paragraph of AI generated text in your report.
- Validate AI-generated code by running tests, baselines, and sanity checks.
- Cite datasets, libraries, external code, and any nontrivial borrowed material.
- Be able to explain every plot, metric, preprocessing step, and conclusion.

DON’T

- Submit AI-generated work you cannot explain.
- Include citations you did not open and verify.
- Claim results you did not reproduce with code.
- Copy AI outputs/prompts from others without attribution.
- Use AI to fabricate experiments, data, or references.

For misconduct consequences, see the course syllabus.

3) Managing complexity (avoid AI-driven “project

explosion”)

What changes vs human-only

AI makes it easy to accumulate: extra features, extra models, extra plots, extra storylines—often without validation.

Complexity control habits that work

- **Minimum Viable Analysis (MVA) first:** one dataset, one target, one baseline, one metric.
- **Freeze key decisions early:** dataset, target, split strategy, primary metric.
- **Keep a short experiment log:** what changed, why, what happened (3 lines per experiment).
- **Prefer simple models and clear diagnostics** before adding complexity.

Common failure modes (and fixes)

Symptom	Likely cause (often AI-driven)	Fix
Great metric, but can’t explain why	Leakage or target proxy features	Audit features + pipeline; do label-shuffle test; time-split if needed
Code works once, then breaks	Hidden notebook state / ad-hoc edits	Convert to a clean run path; restart kernel; rerun from top
10+ models, no conclusion	AI generated options faster than validation	Pick 1 baseline + 1 “best” model; delete the rest; focus on error analysis
Fancy report, weak evidence	AI wrote prose beyond what results support	Replace claims with measured outputs; add diagnostics and baselines
Conflicting results run-to-run	Randomness / unstable pipeline	Set seeds; document versions; simplify; confirm split logic

Examples (What “Good” Looks Like)

A) Example: AI Use Appendix (copy/paste template)

AI Tools Used (tool-agnostic):

- Tool(s): ____________________________
- Dates used: _________________________

How AI was used (check all that apply + brief notes):

☐ Idea generation (how we narrowed to final question): ______________________
☐ Data cleaning suggestions (what we verified): ______________________________
☐ Feature engineering ideas (what we kept/dropped): ________________________
☐ Code generation (modules affected): ______________________________________
☐ Debugging (examples of issues): ____
☐ Writing/editing (sections assisted): ______________________________________
☐ Visualization suggestions: ____

What we verified (required):

- Baseline model built and compared: yes/no (evidence: _________)
- Leakage checks performed: yes/no (evidence: _________)
- Citations verified: yes/no (evidence: _________)
- Reproducibility: “runs on my machine”: yes/no (how tested: _________)

One example of an AI mistake we caught:

- What AI suggested: ____
- Why it was wrong / risky: _______________________________________
- What we did instead: ____

B) Example: Baseline model writeup:

> **Baseline:** Predict the majority class (or mean target).
> **Metric:** Accuracy (or RMSE).

> **Baseline performance:** 0.72 accuracy (or RMSE = 14.3).
> **Our model:** Logistic regression (or linear regression).
> **Performance:** 0.79 accuracy (or RMSE = 12.1).
> **Interpretation:** Improvement is meaningful because it exceeds baseline by X and persists on the held-out test set.

This communicates competence more than a complex model with no benchmark.

C) Example: “Sanity checks” paragraph (QC evidence)

> We validated our pipeline by (1) rerunning the full notebook from a fresh kernel, (2) confirming preprocessing is fit on training only, and (3) performing a label-shuffle test: after shuffling labels, accuracy dropped from 0.79 to 0.51 (near chance), suggesting performance is not due to leakage.

This is exactly the kind of evidence that makes AI-assisted work trustworthy.

D) Example: Complexity reset plan (when things sprawl)

If your project feels out of control, do this:

1. Write your **one-sentence main question**.
2. Keep only: dataset + target + split + baseline + one model + one metric.
3. Delete or archive everything else.
4. Add complexity back only if it answers: “What decision will this change?”

A good project is not the biggest one; it is the clearest one.

E) Oral defense prep (what you must be able to answer)

Be ready to explain, without notes:

1. What is your target and why does it matter?
2. How did you split the data and why?

3. What is your baseline and what does it represent?
4. What preprocessing did you do, and where could leakage occur?
5. Why is your metric appropriate?
6. What are the top 2 errors/failure cases and why do they happen?
7. What is one decision you *changed* after validating results?
8. What would you do next if you had one more week?

If you cannot answer these, reduce complexity until you can.

F) AI safety with sensitive data (required mindset)

- **Do not upload sensitive, proprietary, or personally identifiable data** to external AI tools.
- If unsure whether data is sensitive: treat it as sensitive and ask the instructor.
- When using AI, prefer sharing **schemas, small synthetic examples, or redacted samples** instead of raw private data.

AI Use Disclosure: OpenAI's ChatGPT 5.2 was extensively used to brainstorm ideas, create drafts and check for inconsistencies. (Yosi Shibberu)