

AIRC 201 / TOOLKIT

Build experiments you can trust.

10 classes / 2 hours per class

AIRC 201 turns Python, models, and data into a practical research workflow. Students learn to build a baseline, test a change, and understand what the results mean. Reproducible experiments, a Streamlit demonstration, and a guided mini-paper on a pre-planned multi-agent systems topic bring the work together. Students leave with a workflow they can reuse, adapt, and extend as their own research interests develop.

WHAT YOU WILL DEVELOP

- 01 Build a Python workflow from data preparation to model training and evaluation.
- 02 Use visualizations, validation, and error analysis to understand results.
- 03 Keep experiments reproducible through seeds, configurations, and recorded results.
- 04 Make a model and its results accessible through a Streamlit demonstration.
- 05 Write a guided multi-agent systems mini-paper, including methods, results, limitations, and ethics.

WHAT YOU WILL TAKE AWAY

A practical machine learning workflow, a Streamlit demonstration, and a guided mini-paper that connects an experiment to a research argument.

WHERE IT CAN LEAD

ISEF-affiliated science fairs / S.-T. Yau Science Award / Journal of Emerging Investigators

AIRC 201 / CURRICULUM

Build a reliable workflow.

01 Set up a working research environment

Build confidence with Python, notebooks, and a simple project structure. Set up the working environment, learn Git and GitHub basics, and develop debugging habits that help students understand and improve their own code.

02 Get to know the data

Use NumPy and Pandas to explore, organize, and preprocess a dataset. Connect each preparation choice to the question being investigated, building a foundation for useful analysis and modeling.

03 Make patterns visible

Use Matplotlib to explore data and communicate results. Examine good chart design and creative examples, then make visual choices that help a reader see a pattern and understand what it means.

04 Establish a baseline

Build a scikit-learn workflow with data splits, cross-validation, baseline models, and evaluation metrics. Learn why a reference model matters and how validation helps make comparisons more informative.

05 Look beyond the score

Analyze results and model errors to understand where an approach works and where it falls short. Gain experience running a language model and consider the ethical questions that arise when using AI.

HOW WE LEARN

Lectures, guided mini-projects, and interactive activities. Most work happens in class. Optional homework provides extra practice with the concepts covered in each session. AIRC 101 and 201 can be taken independently.

AIRC 201 / CURRICULUM

Make the work reproducible.

06 **Keep an experiment traceable**

Use seeds, configurations, saved results, and experiment logs to record what changed and what happened. Explore experiment tracking with Weights & Biases and learn how organized records support reproducible work.

07 **Build a research demonstration**

Create a Streamlit interface with inputs and outputs, then connect a model and its plots. Run the demonstration locally and make the experiment easier for someone else to inspect and understand.

08 **Bring AI into a working application**

Use language models effectively in a practical workshop. Build a data-driven Streamlit application that connects the course's tools, giving students experience moving between an idea, an implementation, and a usable demonstration.

09 **Turn the experiment into a paper**

Develop a guided mini-paper on a pre-planned multi-agent systems topic using a template and checkpoints. Compare strategies, describe methods and results, and discuss the work's limitations and ethical considerations.

10 **Present the work and look ahead**

Present the project and explain the choices behind its methods and results. Explore selected special topics and consider how the course's research toolkit can support further experimentation or a future original research project.

BEFORE THE FIRST CLASS

Bring a Mac or Windows computer. Install VS Code, Conda or Python, and Git; create a GitHub account.