

Tutorial Proposal Review Guidelines

Tutorial proposals in the form of an abstract should include a clear and concise description of the proposed coding tutorial, including the ML methods and tools to be highlighted, the climate-relevant challenges the tutorial aims to address, and the users' expected learning outcomes.

Accepted tutorials will be presented in the form of executable notebooks (e.g. Jupyter, Colab). They should demonstrate a step-by-step code walkthrough of how ML methods or tools (such as libraries, packages, services, datasets, or frameworks) can be used to address climate change. Submissions will be reviewed based on their potential impact and overall usability by the climate change and AI research communities. Specifically, the tutorials will be reviewed based on the following rubric:

Climate Relevance. Is the problem tackled in the tutorial relevant to climate change (mitigation, adaptation, or climate science)? How important is it to solve this problem (in terms of reducing greenhouse gas emissions or helping society adapt to a changing climate)?

Technical Correctness. Are the methods, tools, or techniques described in the proposed tutorial well-matched to the problem? Is ML/AI necessary, or might simpler tools be sufficient?

Pedagogy/Usability. Do the authors describe the potential impact of the proposed methods or tools introduced in the tutorial? Do the authors clearly articulate the ways in which work that builds on the tutorial would fit into broader efforts to address climate change?

Clarity/ Accessibility. How clear and accessible is the submission overall? Would the proposed tutorial be accessible to broad technical audiences? Are the goals and methods described with an appropriate level of detail?

Dataset/Tools/Tutorial Format. Are the tools and datasets to be used in the tutorial free and openly accessible to the general public? Note that there is a strong preference for Python, specifically interactive Colab notebooks, but we also welcome tutorials written in other forms or languages, e.g. Binder, R, Matlab, etc.