LLM-CODEVAL: A Framework for Verifying Implementations of Mathematical Functions Using Language Models [Suplementary material]

Felipe Cunha[1], Mirko Perkusich[1], Danyllo Albuquerque[1], Kyller Gorgonio[1], Angelo Perkusich[1]
[1]Federal University of Campina Grande - Intelligent Software Engineering Group (ISE/VIRTUS) - Paraíba, Brazil

July 2025

About this Supplementary Material

This document provides examples of **LLM-generated implementations before and after manual review** within the LLM-CODEVAL framework. All examples are based on the `wmean` function to illustrate the type of corrections typically introduced during the human-in-the-loop stage.

Table 1: Examples of LLM Output Before and After Manual Review (`wmean` Function)

Before Review (LLM Output)	**After Review (Validated)**
Example 1 – Missing Weight Validation: The LLM-generated code computed the weighted mean correctly but did not enforce that weights $w_i \geq 0$. Negative weights would silently pass.	**Example 1 – Correction:** Introduced explicit check: `if w < 0:  raise ValueError("Weights must be non-negative.")` This ensures compliance with the functional contract and prevents semantic violations.
Example 2 – Lack of Type Checking: The initial code assumed all inputs were numeric. Passing a string or `None` caused runtime errors without clear messages.	**Example 2 – Correction:** Added type validation: `if not isinstance(w,(int,float)) or not isinstance(x,(int,float)):` `raise ValueError("Arguments must be numeric.")` This avoids silent failures and improves robustness.