Jamie Wessels | Workshop Overview

CHOP Data Instructional Specialist Interview

"You are asked to present a half-day workshop that teaches a computational or data-related skill of your choosing to an audience of adult learners who have requested to participate and are motivated to learn. This could be virtual or in-person, and you can propose an audience size that makes sense for the topic and environment. For example, perhaps you are teaching novice web developers how to make a simple React web app, or instructing a group of lab techs how to bring their Excel spreadsheets into Python and visualize their data to understand it better, or helping graduate students design a data collection instrument that reduces bias."

Workshop Overview

Objectives/ Goals

I want students to walk away from the workshop with:

1. A knowledge of *why* it's important to quantify test metrics during model selection, tuning, and reporting of results
2. Example code for how this might be used in practice
3. Resources for further exploration

Objectives*:*
SWBAT describe the bias variance trade-off
SWBAT explain the difference between training data and test data
SWBAT explain the importance and motivations behind cross-validation
SWBAT perform a train_test_split in order to create train and holdout sets
SWBAT estimate the test error for different values of k in a KNN algorithm and choose the "best model" using KFold cross-validation

Description of the Audience

A description of the audience and any techniques you might use to discover their strengths and weaknesses more fully

Prior Knowledge:

- *Classification vs. Regression*
- *Loss function / metrics:* students have experience choosing appropriate metrics to evaluate model performance (both classification and regression)
- *K Nearest Neighbors (KNN) Algorithm & Hyperparameter(s)*
- *Basics of using Jupyter Notebook/Lab, Numpy, Pandas, Matplotlib.pyplot*
- *Intro to sklearn (through KNN lecture)*

Requirements for Workshop:

- *Students should bring a dataset (can be from shared database)*

Accommodations:

- *Lecture will be given from a jupyter notebook (slides format, but students have the code to run alongside instructor)*
    - *This gives students extra processing time and provides a space for meaningful comments*
    - *This also allows instructor to adjust/run code in real time depending on the question*
- *For new concepts, start with high level visuals before diving into code*
- *Ideally, instructor would have office hours following workshop*
- *Providing troubleshooting tools*
    - *Important to <u>show</u> students how to look at documentation, how to search stack overflow, how to use dir(xxx), print statements, breakpoints, etc.*

Assessing Skills:

- *Upfront Planning: try to anticipate misconceptions ahead of time*
    - *Examples from this workshop: train test split unpacking, nomenclature of holdout/test/validation sets, fitting on training set only*
- *Concept Knowledge: Frequent checks for understanding during lectures (I prefer cold calling to get a more accurate pulse of the room)*
- *Programming: slightly more difficult in a virtual setting.*
    - *Checks for understanding: for example, ask volunteers to walk me through the lecture code in their own words (or in pairs)*
    - *Provide starter code for accessibility: gives students a starting point*
    - *Pop around breakout rooms during pair activity*

Agenda

An itinerary / timeline of what is taught and what methods are used

Objectives and Agenda

Intro: Which model would you choose? + discussion

Topic 1: Bias-variance tradeoff
    Lecture
    Simulation (jupyter nb)

Topic 2: Cross-validation
    Lecture
    Simulation cont (jupyter nb)
    Additional Resources

Partner Task (breakout rooms, jupyter nb)

Solutions Discussion

Individual Task: (students bring their own dataset and/or analysis)

- Provide scaffolding for students who are more advanced vs. those who are at beginning stages

Closeout
    Revisit objectives and final checks for understanding
    Up next: addressing overfitting for different methods (ex: regularized regression)

Classroom Norms

Expectations / community agreements / code of conduct you would offer to learners

- Safety Norms
    - Creating a safe and inclusive learning environment is a non-negotiable. Our words matter.
    - All questions and experiences are valid and important.
    - Step up/ step back (share the mic) during discussions.
    - Take risks and support others in their risk-taking. This is how we learn.

- Participation Norms
    - Cameras on, when possible
    - Mute your mic if you aren't speaking, but also feel free to unmute and ask a question at any time during the lecture
    - There is a lot of knowledge in this room! If you have something you'd like to share with the group regarding your experience on the topic, please do so!

- Above are my non-negotiables. Together, our class can add any additional norms that they would like to see.

Measuring Success:

Proposed methods to measure the success of the workshop

Frequent checks for understanding during lectures

- In the moment corrections and clarification
- "Take the pulse of the room"

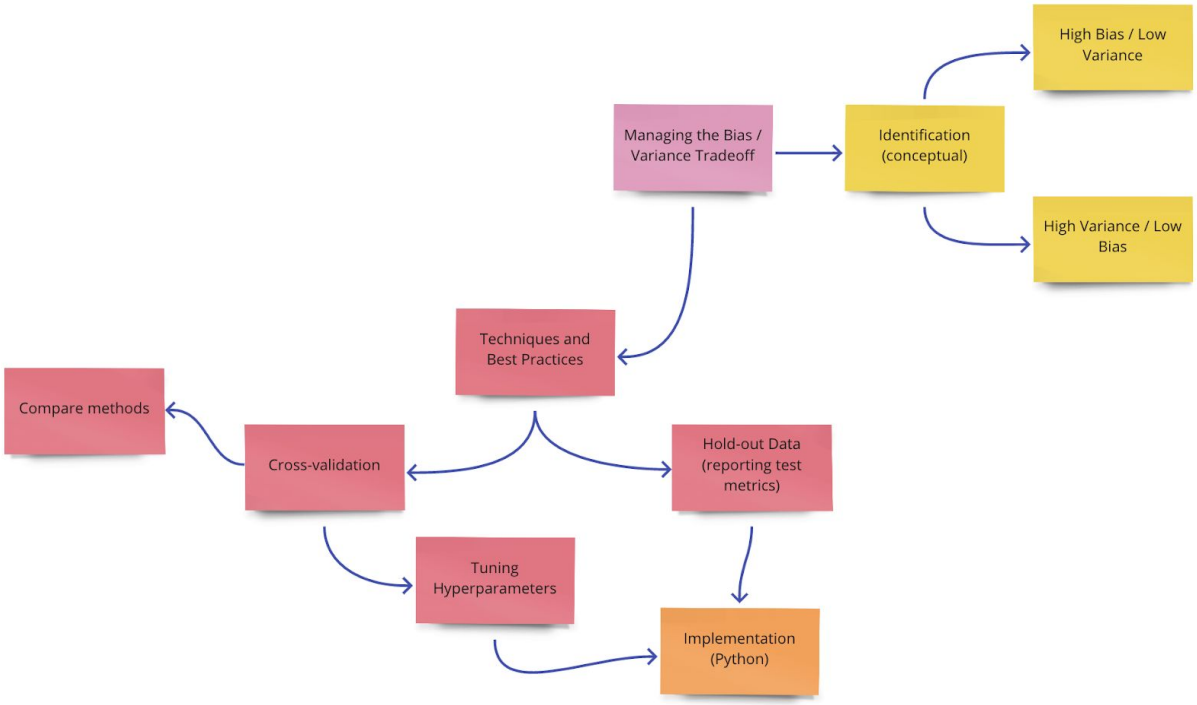
Partner task → initial assessment of class as a whole

- Provides one last opportunity to address misconceptions/themes observed

Individual task → students turn share their code to instructor at end for feedback

Concept Map

A mind map / concept map of the topics you're covering


High Bias / Low Variance
Managing the Bias / Variance Tradeoff
Identification (conceptual)
High Variance / Low Bias
Techniques and Best Practices
Compare methods
Cross-validation
Hold-out Data (reporting test metrics)
Tuning Hyperparameters
Implementation (Python)