Introduction to Design of Experiments with Machine Learning
MECE 5397/6397
Design of Experiments/ Machine Learning

Integrating Design of Experiments (DoE) concepts with Machine Learning (ML) and Artificial Intelligence (AI) can significantly enhance the efficiency and effectiveness of experimental design, data collection, analysis, and the overall learning process. DoE is a systematic method to plan, conduct, and analyze experiments efficiently. Here are several ways to integrate DoE concepts with ML and AI:

Feature Selection and Optimization

- DoE for Feature Engineering: Use DoE principles to systematically explore combinations of features to determine their impact on model performance. This can help in identifying the most relevant features, interactions between features, and optimal feature transformations.
- Hyperparameter Optimization: Apply DoE techniques like factorial designs or response surface methodology to optimize hyperparameters of ML models. This systematic approach can be more efficient than random or grid search by exploring the parameter space more strategically.

Model Experimentation and Validation

- Comparative Studies of Models: Use DoE to design experiments for comparing different ML models or algorithms under various conditions. This can help in understanding the conditions under which certain models perform better and in validating model robustness.

Problem Setup:

Example Scenario: Code

Let's say we have a dataset with three features (X1, X2, X3) and we want to understand how different transformations (e.g., logarithmic, square root, and square) applied to these features affect the performance of a linear regression model. We'll use a 4-level full factorial design, considering the presence or absence of each transformation as the four levels.

The sample code provides an example for a full-factorial design at 4 levels and evaluates a Linear Regression model to minimize MSE based on the randomization of the different transformation functions.

Explore the following To Hand in: 7/18/2024

1) Replace the linear regression model with a nonlinear model in the context of a three-factor, three-level factorial design, use a decision tree regressor. Decision trees are capable of capturing nonlinear relationships between features and the target variable. We'll still be using the pyDOE2 package to create the factorial design but adjust the design to have three levels for each of the three factors. Modify using the following non-linear regressors:
    a. **DecisionTreeRegressor**
    b. **Random Forest Regressor**

c. **Support Vector Regression (SVR)**
d. **K-Nearest Neighbors Regressor (KNN)**

Key Changes:

1. Model Replacement: The LinearRegression model is replaced with DecisionTreeRegressor, Random Forest Regressor, etc from Scikit-learn, which can handle complex, nonlinear relationships better.
2. Three-Level Design: Each factor in the design matrix now has three levels, accommodating different transformations: no transformation, logarithmic, and square.
3. Safety in Transformations: A small constant (0.1) is added to X where logarithmic transformations are used to avoid taking the logarithm of zero.
4. This setup allows you to explore the impact of different nonlinear transformations on the dataset using a model that can capture such nonlinearities effectively.
5. Compare the results for MSE for the regressor models explored.