Gators4SCI

ML4SCI
NMR Spin Challenge

Avanti Bhandarkar
Keerthiraj Nagaraj
Enes Grahovac

Overview

1. Problem Formulation
2. Motivation
3. ML Models
4. Data preprocessing
5. Model training
6. Gators4SCI Team

Problem formulation

- Design and train a model that predicts the strength and shape of interactions between the nuclear spins from simulated time-dependent magnetization curves
- From ML point of view - Predict four real numbers from a large input vector of real numbers
- Multi-Target Regression - One Vs different models for different targets.
- We decided to test multiple models for different targets.
    - This allows for more flexibility while choosing right model and its hyperparameters for a given target variable.

Motivation for our approach

- The data provided had 10,000 samples for model training with each sample having a dimensionality of (942x1), so overall 942 features in the data.
- We discussed about pros and cons of using a “simple Vs complex” ML model.
- We expected slightly complex models such as MLP neural networks to overfit with this data and our experiments validated it.
- Trusting Occam’s Razor rule, we decided to stick with simpler ML models.
- We ended up using Multivariate Linear Regression, Bayesian Ridge Regression, and Support Vector Regression.

Regression Models

- Linear Regression
  - Attempts to find the mapping from input regressors to target using ordinary least squares fit method.
  - Assumes a direct correlation between the independent (input) and dependent (target) variables.
- Support Vector Regression (SVR)
  - Aims to find decision boundaries around a hyperplane that fits good amount of training data and SVR accounts for non-linearity in the data and provides proficient prediction model.
- Bayesian Regression
  - This method works well even with insufficient data or poorly distributed data by formulating linear regression using probability distributors rather than point estimators

Data preprocessing

- We experimented with various data preprocessing techniques such as feature standardization, dimensionality reduction and feature selection.
- Selecting certain features gave us better performance as some features were not contributing anything to the target prediction and were only acting as noise.
- We used f_regression score, which is calculated based on correlation between each regressor input and target, F-score and a p-value.
- For 2 of the 4 targets, top 100 features were enough for prediction with low errors.

Model training

- We performed 5-fold cross validation to choose the hyperparameters for each target.
- This improved the model generalizability and helped us choose the optimal hyperparameters.
- We experimented with more than 10 regression models and >50 hyperparameter combination to get highest generalizability for our models.
- Our solution consisted of the following models implemented using Scikit-learn.
  - Correlation strength (Alpha) - Linear Regression (with intercept + top 300 features)
  - Correlation length (xi) - Bayesian Ridge Regression (with top 100 features)
  - Correlation power (p) - Support Vector Regression (C=10.0, epsilon=1e-4, kernel= polynomial)
  - Dissipation power (d) - Bayesian Ridge Regression (with top 100 features)

Gators4SCI

Team

Enes Grahovac
Physics, Undergraduate
FICS AI Researcher

Keerthiraj Nagaraj
Comp.Eng., PhD Student
WAM System Lab, UF

Avanti Bhandarkar
Comp.Eng., PhD Student
FICS AI Researcher

Resources

https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/

https://towardsdatascience.com/introduction-to-bayesian-linear-regression-e66e60791ea7

https://towardsdatascience.com/understanding-the-fundamentals-of-linear-regression-7e64afd614e1

https://scikit-learn.org/stable/user_guide.html