Training and Predicting with Linear Regression Models in scikit-learn

In this lecture, Professor Ashish explains how to train and make predictions using linear regression models in scikit-learn, covering both the normal equation method and iterative optimization using stochastic gradient descent (SGD).

Training Linear Regression Models

The lecture details two approaches for training linear regression models:

1. **Normal Equation:** This method uses the `LinearRegression` class from `sklearn.linear_model`. The training process involves instantiating the `LinearRegression` object and then calling the `fit` method with the training feature matrix (`X_train`) and label vector (`y_train`).

2. **Iterative Optimization (SGD):** This approach employs the `SGDRegressor` class, suitable for larger datasets. The `fit` method is used similarly, but `SGDRegressor` offers more control over the optimization process through various parameters.

`SGDRegressor` Parameters and Optimization

The lecture emphasizes the importance of several `SGDRegressor` parameters:

- **`learning_rate`:** Controls the step size during optimization. Inverse scaling is often used, calculated as (\eta_t = \frac{\eta_0}{t^{power_t}}), where (\eta_0) is the initial learning rate, (t) is the current iteration, and `power_t` is a hyperparameter.
- **`average`:** Enables averaging of the weights after a specified number of samples, improving performance with many features and a high (\eta_0). Setting `average` to an integer (n) starts averaging after seeing (n) samples.
- **`warm_start = True`:** This allows initializing SGD with the weight vector from a previous run, useful for monitoring loss iteration by iteration. Setting `max_iter = 1` and iteratively calling `fit` with `warm_start = True` enables this.

Tip: Always set a `random_state` in the `SGDRegressor` constructor for reproducible results.

Making Predictions

Prediction on new data involves two steps:

1. Arrange the data into a feature matrix (shape (#samples, #features)) or a sparse matrix.
2. Call the `predict` method on the trained linear regression object (either `LinearRegression` or `SGDRegressor`) with the feature matrix as an argument. The method returns the predicted labels.

Feature Scaling and DummyRegressor

The lecture also touches upon feature scaling, highlighting that SGD is sensitive to it. It recommends using `StandardScaler` for preprocessing. Finally, it briefly introduces `DummyRegressor`, a baseline model that predicts using strategies like mean, median, quantile, or a constant value.

Prof. Ashish concludes by summarizing the key steps in training and using linear regression models in scikit-learn, emphasizing the flexibility and control offered by `SGDRegressor` for large datasets and the importance of proper parameter tuning and feature scaling.