Regularization: Takeaways

by Dataquest Labs, Inc. - All rights reserved © 2022

Syntax

- Creating a `RidgeCV` object for ridge regression with built-in cross-validation:

```
from sklearn.linear_model import RidgeCV
model = RidgeCV(alphas=linspace(0.1, 10, num=100))
model.fit(X, y)
```

- Creating a `LassoCV` object for LASSO with built-in cross-validation:

```
from sklearn.linear_model import LassoCV
lasso = LassoCV(alphas=linspace(0.1, 10, num=100))
lasso.fit(X, y)
```

- Creating a `StandardScaler` object which helps to standardize all of the features of a dataset `X`:

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
standardized_X = scaler.fit_transform(X)
```

Concepts

- Regularization is the process of simplifying a model so that it generalizes better to unseen data.
- One way we can regularize a model is to reduce the magnitude of the coefficients in a model. This is potentially useful in high-dimensional contexts because we need a way to narrow down the feature set.
- Ridge regression implements a penalty term in the loss function to reduce coefficient magnitude.
- LASSO also incorporates a penalty term, but its penalty also enables feature selection.
- Standardization is essential for regularized models because we need to make sure that each of the coefficients are penalized similarly.

Resources

- `scikit-learn` official documentation
- An explanation on why LASSO does feature selection
- RidgeCV()
- LassoCV()
- StandardScaler()

Takeaways by Dataquest Labs, Inc. - All rights reserved © 2022