

best_model_builder

Python Library to automate and score combinations of encoder, scaler and K value for K-fold validation and model parameters for classification in machine learning.

Attributes:

split_size_list_default: *list, default=[0.7, 0.8, 0.9]*

Default list with values of split size (of training data) to be tried

split_size_list_default: *list, default=[StandardScaler(), MinMaxScaler(), MaxAbsScaler(), RobustScaler()]*

Default list of scalers to be tried

encoder_default: *list, default=[LabelEncoder(), OneHotEncoder(), OrdinalEncoder()]*

Default list of scalers to be tried

model_params_default: *dict*

Dictionary of default models parameters for each model. Each model parameters (dict) is mapped with name of the model (key)

encoder_default: *list, default=[LabelEncoder(), OneHotEncoder(), OrdinalEncoder()]*

Default dictionary of models to be tried. Each string name of model and the function is mapped

Methods:

best_combination(N, X, y, **kwargs)

Major Function. Return N best combinations with X, y and arguments input

build_model(K, model_key, X_train, y_train, X_test, y_test, model_params, split_size, method, scaler, encoder)

Get parameters for this estimator.

scaler_encode_split(X, y, split_size, encoder, scaler)

Predict using the linear model.

best_model_builder.best_combination

Major Function. Return N best combinations of classification model with K-Fold validation with X, y and arguments input

Parameters:

N: int

Number of combinations of best score to get returned

X{array-like, sparse matrix} of shape (n_samples, n_features)

Training data.

y{array-like of shape (n_samples,) or (n_samples, n_targets)}

Target values.

****kwargs**

Keyword arguments for setting **attributes** to make various combinations

Attributes:

get_distinct_values: bool, default=False

Whether user gets distinct results. When True, only one combination will be decided for one score

K_range: int, default: 5

K value for k-fold validation. Grid is created for K value in range (2, K_range)

split_size_list: list, default: split_size_list_default

List of split size (of train data) to be tried

scaler_list: list, default: scaler_list_default

List of scalers to be tried

encoder_list: list, default: encoder_list_default

List of encoders to be tried

model_params: dict, default: model_params_default

Dictionary of models parameters for each model.

Each model parameters (dict) is mapped with name of the model (key)

method: string, default: "accuracy"

Method of model scoring method. Input should be one of "accuracy, recall, precision, f1"

Returns:**df_result[:N]**

Return a N rows dataframe with combinations

columns=["model", "K", "model_params", "split_size", method, "scaler",
"encoder"]
