

DATA SCIENCE

Leandro Ferrado, Javier Lezama, Valentina Rubiolo

ACÁMICA

16 de Mayo de 2019

Ajustes del Modelo

- GridSearch.

Ajustes del Modelo

- GridSearch.
- `from sklearn.model_selection import GridSearchCV.`

Ajustes del Modelo

- GridSearch.
- `from sklearn.model_selection import GridSearchCV.`
- `param_grid = ['max_depth' : [2, 3, 4], 'max_features' : [2, 4, 6, 8],]`.

Propiedades de GridSearch

- Mejores parametros encontrados

Propiedades de GridSearch

- Mejores parametros encontrados
- `grid_search.best_params_`.

Propiedades de GridSearch

- Mejores parametros encontrados
- `grid_search.best_params_`.
- Mejor estimador.

Propiedades de GridSearch

- Mejores parametros encontrados
- `grid_search.best_params_`
- Mejor estimador.
- `grid_search.best_estimator_`

Ajustes del Modelo

- GridSearch.

Ajustes del Modelo

- GridSearch.
- `from sklearn.model_selection import GridSearchCV.`

Ajustes del Modelo

- GridSearch.
- `from sklearn.model_selection import GridSearchCV.`
- `param_grid = [{'max_depth' : [2, 3, 4], 'max_features' : [2, 4, 6, 8]},]`.

Ajustes del Modelo

- GridSearch.
- `from sklearn.model_selection import GridSearchCV.`
- `param_grid = [{'max_depth' : [2, 3, 4], 'max_features' : [2, 4, 6, 8]},]`.

Ajustes del Modelo

- GridSearch.
- `from sklearn.model_selection import GridSearchCV.`
- `param_grid = [{'max_depth' : [2, 3, 4], 'max_features' : [2, 4, 6, 8]},].`

```
tree_reg = DecisionTreeRegressor()  
grid_search = GridSearchCV(tree_reg, param_grid, cv=5,  
                           scoring='neg_mean_squared_error',  
                           return_train_score=True)  
grid_search.fit(X_train, y_train)
```

Propiedades de GridSearch

- Mejores parametros encontrados

Propiedades de GridSearch

- Mejores parametros encontrados
- `grid_search.best_params_`.

Propiedades de GridSearch

- Mejores parametros encontrados
- `grid_search.best_params_`.
- Mejor estimador.

Propiedades de GridSearch

- Mejores parametros encontrados
- `grid_search.best_params_`
- Mejor estimador.
- `grid_search.best_estimator`

Ajustes del Modelo

- RandomizedSearch.

Ajustes del Modelo

- RandomizedSearch.
- `from sklearn.model_selection import RandomizedSearchCV.`

Ajustes del Modelo

- RandomizedSearch.
- `from sklearn.model_selection import RandomizedSearchCV.`
- `from scipy.stats import randint as sp_randint.`

Propiedades de RandomizedSearch

- Mejores parametros encontrados

Propiedades de RandomizedSearch

- Mejores parametros encontrados
- `rand_search.best_params_`

Propiedades de RandomizedSearch

- Mejores parametros encontrados
- `rand_search.best_params_`.
- Mejor estimador.

Propiedades de RandomizedSearch

- Mejores parametros encontrados
- `rand_search.best_params_`
- Mejor estimador.
- `rand_search.best_estimator_`

Ajustes del Modelo

- RandomizedSearch.

Ajustes del Modelo

- RandomizedSearch.
- `from sklearn.model_selection import RandomizedSearchCV.`

Ajustes del Modelo

- RandomizedSearch.
- `from sklearn.model_selection import RandomizedSearchCV.`
- `from scipy.stats import randint as sp_randint.`

Ajustes del Modelo

- RandomizedSearch.
- `from sklearn.model_selection import RandomizedSearchCV.`
- `from scipy.stats import randint as sp_randint.`
- `param_grid`
= `{ 'max_depth' : sp_randint(2, 4), 'max_features' : sp_randint(2, 8) }`

Ajustes del Modelo

- RandomizedSearch.
- `from sklearn.model_selection import RandomizedSearchCV.`
- `from scipy.stats import randint as sp_randint.`
- `param_grid`
= `{ 'max_depth' : sp_randint(2, 4), 'max_features' : sp_randint(2, 8) }`

Ajustes del Modelo

- RandomizedSearch.
- `from sklearn.model_selection import RandomizedSearchCV.`
- `from scipy.stats import randint as sp_randint.`
- `param_grid`
`= {'max_depth' : sp_randint(2, 4), 'max_features' : sp_randint(2, 8)}`

```
tree_reg = DecisionTreeRegressor()  
rand_search = RandomizedSearchCV(tree_reg,  
param_distributions=param_grid, cv=5,  
    scoring='neg_mean_squared_error',  
    n_iter=10,  
    return_train_score=True)  
rand_search.fit(X_train, y_train)
```

Propiedades de RandomizedSearch

- Mejores parametros encontrados

Propiedades de RandomizedSearch

- Mejores parametros encontrados
- `rand_search.best_params_`

Propiedades de RandomizedSearch

- Mejores parametros encontrados
- `rand_search.best_params_`.
- Mejor estimador.

Propiedades de RandomizedSearch

- Mejores parametros encontrados
- `rand_search.best_params_`
- Mejor estimador.
- `rand_search.best_estimator_`

Evaluación en el test set

- `y_pred = grid_search.predict(X_test).`

Evaluación en el test set

- `y_pred = grid_search.predict(X_test).`
- `y_pred = rand_search.predict(X_test).`

Bibliografía

<https://towardsdatascience.com/grid-search-for-model-tuning-3319b259367e>

<https://towardsdatascience.com/algorithms-for-hyperparameter-optimisation-in-python-edda4bdb167>

<https://medium.com/datadriveninvestor/an-introduction-to-grid-search-ff57adcc0998>

https://scikit-learn.org/stable/modules/grid_search.html

[https://scikit-](https://scikit-learn.org/stable/auto_examples/model_selection/plot_grid_search_digits.html)

[learn.org/stable/auto_examples/model_selection/plot_grid_search_digits.html](https://scikit-learn.org/stable/auto_examples/model_selection/plot_grid_search_digits.html)

[https://scikit-](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html)

[learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html)

<https://stackabuse.com/cross-validation-and-grid-search-for-model-selection-in-python/>