Big Data e Computação em Nuvem

Aula 08

Spark ML

Prof. Michel Fornaciali, PhD.
Prof. Thanuci Silva, PhD.

Contatos:
MichelSF@insper.edu.br
thanucis@insper.edu.br

Insper

Panorama da disciplina

O que veremos?

- **Aula 7 [09/nov]:**
  - Abertura projeto final
  - DataFrames em análise descritiva de datasets reais
- **Aula 8 [11/nov]:**
  - Spark Machine Learning
  - Spark Pipelines
  - Projeto final
- **Aula 9 [18/nov]:**
  - ML em datasets reais
  - Projeto final
- **Aula 10 [23/nov]:**
  - Recursos avançados de DataFrame
  - Checkpoint
- **Aula 11 [25/nov]:**
  - Cloud
  - Projeto final
- **Aula 12 [04/dez]:**
  - Projeto final
- **Aula 13 [09/dez]:**
  - Apresentação do projeto final

Spark ML

Machine Learning em larga escala

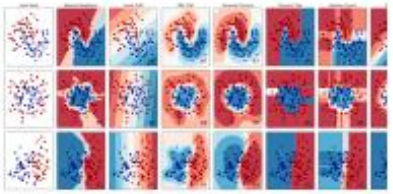
scikit-learn (sklearn)

Principal biblioteca Python para ML

Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...

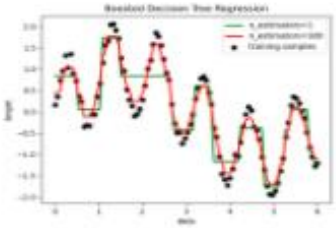


Examples

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...



Examples

Clustering

Automatic grouping of similar objects into sets.

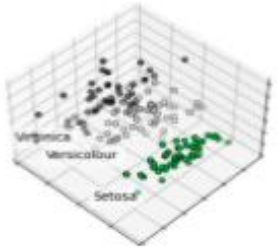
Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...

Examples

Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, Increased efficiency
Algorithms: k-Means, feature selection, non-negative matrix factorization, and more...

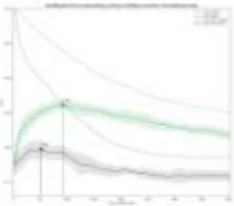


Examples

Model selection

Comparing, validating and choosing parameters and models.

Applications: Improved accuracy via parameter tuning
Algorithms: grid search, cross validation, metrics, and more...

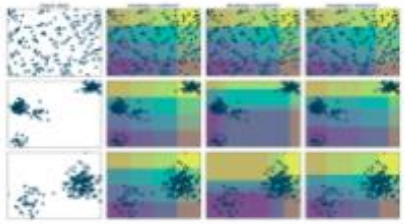


Examples

Preprocessing

Feature extraction and normalization.

Applications: Transforming input data such as text for use with machine learning algorithms.
Algorithms: preprocessing, feature extraction, and more...



Examples

News

On-going development: What's new (Changelog)

October 2021. scikit-learn 1.0.1 is available for download (Changelog).

September 2021. scikit-learn 1.0 is available for download (Changelog).

April 2021. scikit-learn 0.24.2 is available for download (Changelog).

January 2021. scikit-learn 0.24.1 is available for download (Changelog).

December 2020. scikit-learn 0.24.0 is available for download (Changelog).

August 2020. scikit-learn 0.23.2 is available for download (Changelog).

May 2020. scikit-learn 0.23.1 is available for download (Changelog).

May 2020. scikit-learn 0.23.0 is available for download (Changelog).

Scikit-learn from 0.23 requires Python 3.6 or newer.

March 2020. scikit-learn 0.22.2 is available for download (Changelog).

January 2020. scikit-learn 0.22.1 is available for download (Changelog).

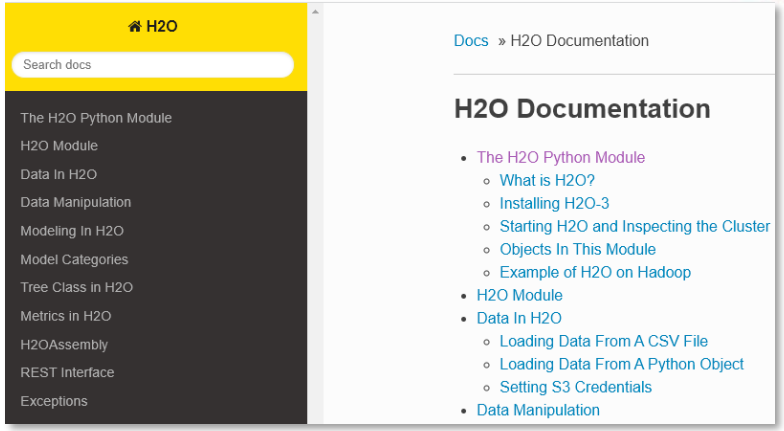
December 2019. scikit-learn 0.22 is available for download (Changelog and Release Highlights).

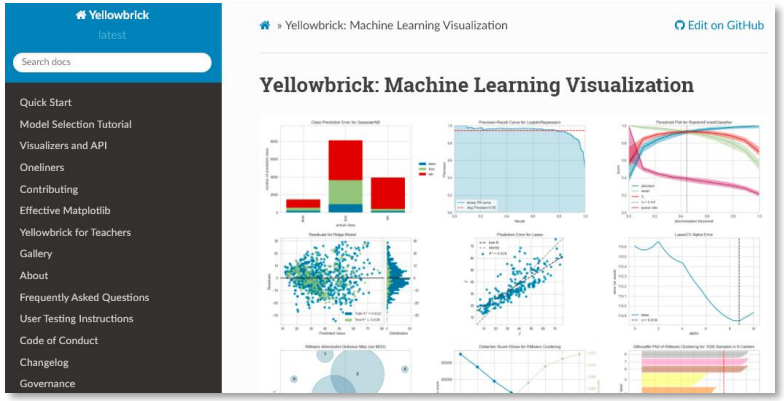
Insper

scikit-learn (sklearn)

Principal biblioteca Python para ML

```python
import matplotlib.pyplot as plt
import numpy as np
from sklearn import datasets, linear_model
from sklearn.metrics import mean_squared_error, r2_score

# Load the diabetes dataset and use only one feature
diabetes_X, diabetes_y = datasets.load_diabetes(return_X_y=True)
diabetes_X = diabetes_X[:, np.newaxis, 2]

# Split the DATA and TARGETS into training/testing sets
diabetes_X_train = diabetes_X[:-20]
diabetes_X_test = diabetes_X[-20:]
diabetes_y_train = diabetes_y[:-20]
diabetes_y_test = diabetes_y[-20:]

# Create linear regression object
regr = linear_model.LinearRegression()

# Train the model using the training sets
regr.fit(diabetes_X_train, diabetes_y_train)

# Make predictions using the testing set
diabetes_y_pred = regr.predict(diabetes_X_test)

# The coefficients
print("Coefficients: \n", regr.coef_)
# The mean squared error
print("Mean squared error: %.2f" % mean_squared_error(diabetes_y_test, diabetes_y_pred))
# The coefficient of determination: 1 is perfect prediction
print("Coefficient of determination: %.2f" % r2_score(diabetes_y_test, diabetes_y_pred))
```





K Keras

Insper

SparkML

Machine Learning em Spark

Spark 1.6.0 | Overview | Programming Guides | API Docs | Deploying | More

spark.ml package

- **Overview: estimators, transformers and pipelines**
- Extracting, transforming and selecting features
- Classification and Regression
- Clustering
- Advanced topics

spark.mllib package

- Data types
- Basic statistics
- Classification and regression
- Collaborative filtering
- Clustering
- Dimensionality reduction
- Feature extraction and transformation
- Frequent pattern mining
- Evaluation metrics

Overview: estimators, transformers and pipelines - spark.ml

The `spark.ml` package aims to provide a uniform set of high-level APIs built on top of DataFrames that help users create and tune practical machine learning pipelines. See the algorithm guides section below for guides on sub-packages of `spark.ml`, including feature transformers unique to the Pipelines API, ensembles, and more.

Table of contents

Insper

SparkML

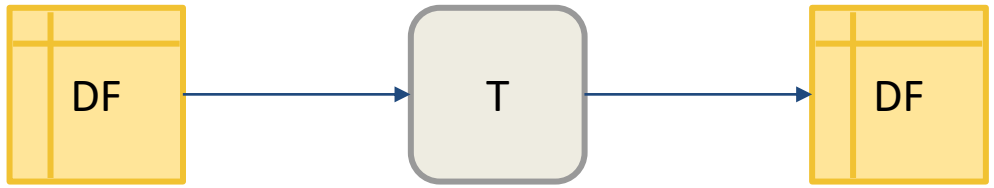
Principais componentes

- DataFrame
- Transformer
- Estimator
- Pipeline
- Parameter

SparkML

Principais componentes

- DataFrame
- **Transformer**
- Estimator
- Pipeline
- Parameter



Transformer

.transform()

This section covers algorithms for working with features, roughly divided into these groups:

- Extraction: Extracting features from "raw" data
- Transformation: Scaling, converting, or modifying features
- Selection: Selecting a subset from a larger set of features

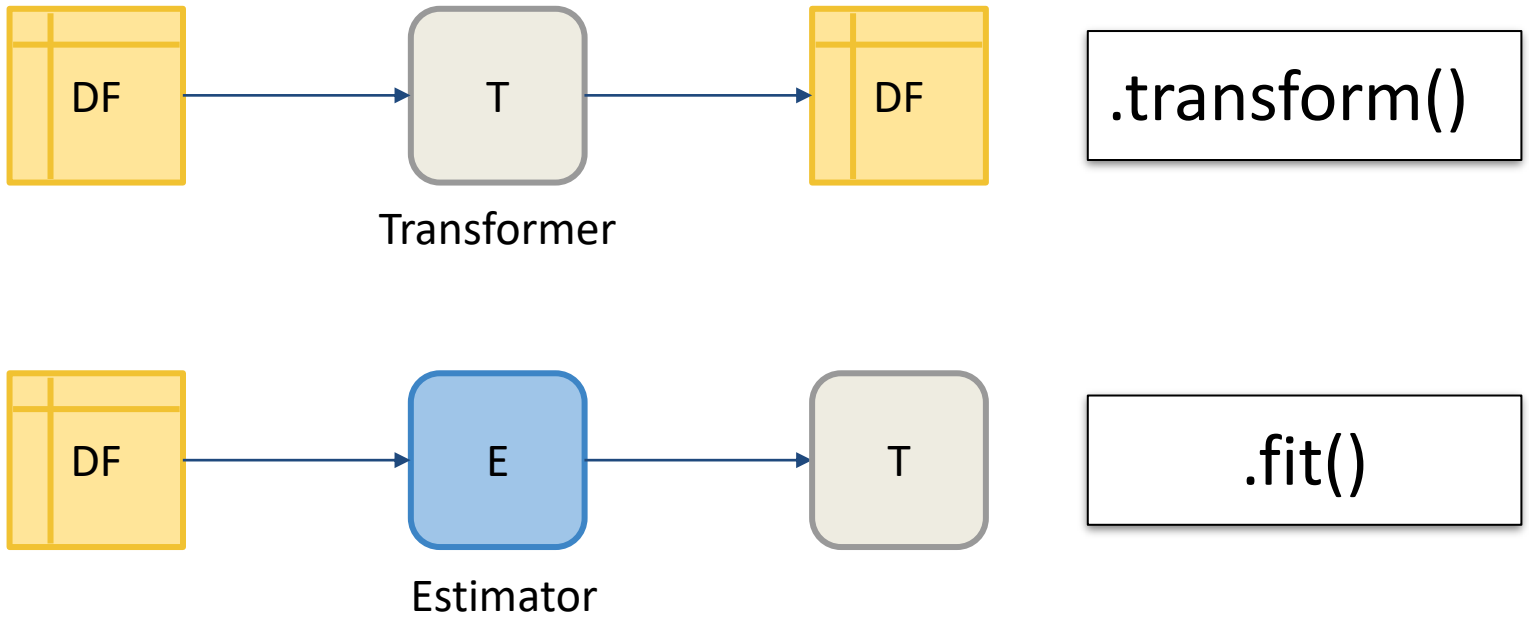
Table of Contents

Insper

SparkML

Principais componentes

- DataFrame
- Transformer
- **Estimator**
- Pipeline
- Parameter



SparkML

Principais componentes

- DataFrame
- Transformer
- **Estimator**
- Pipeline
- Parameter

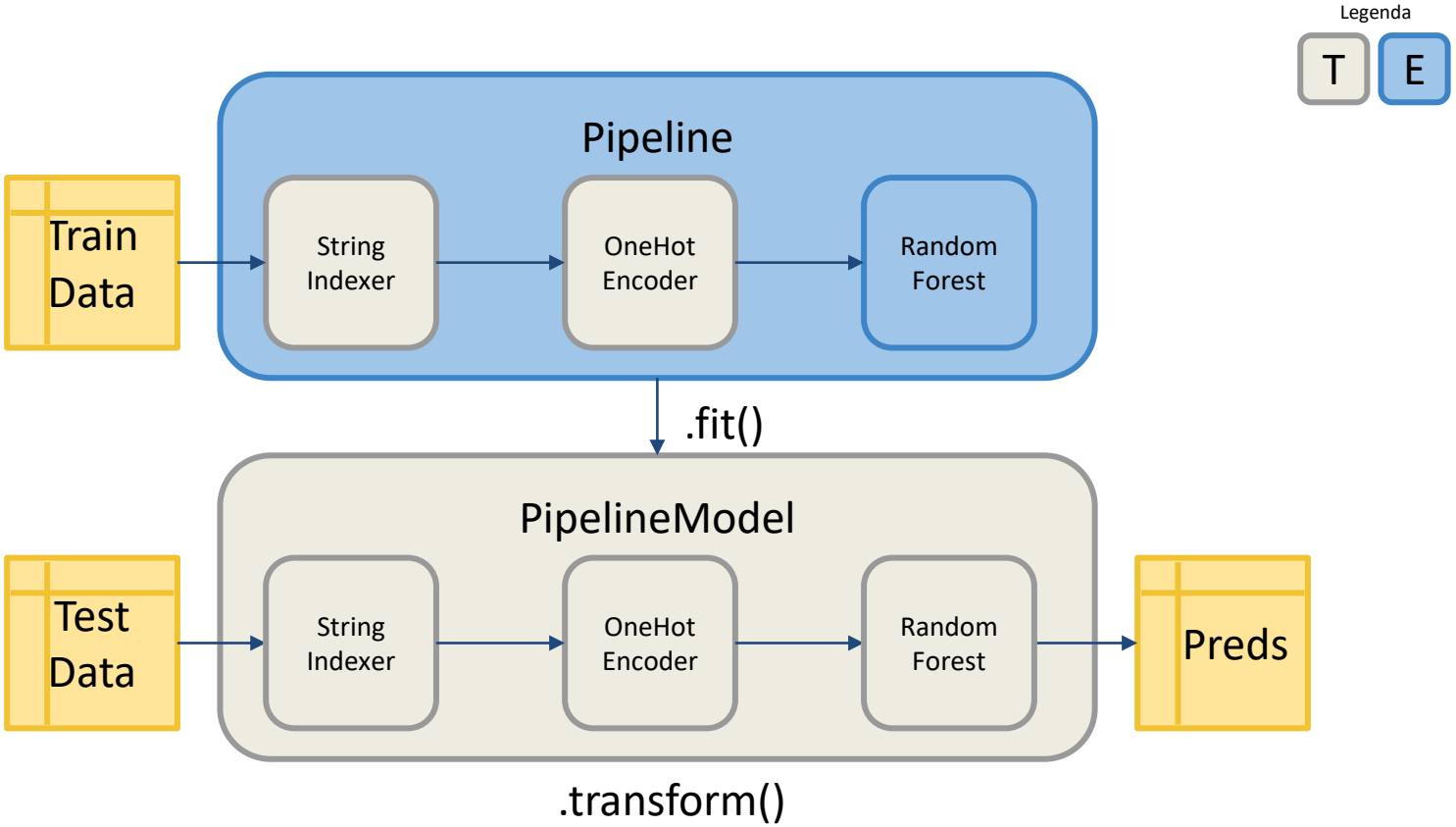
Table of Contents

Insper

SparkML

Principais componentes | Pipeline durante o desenvolvimento

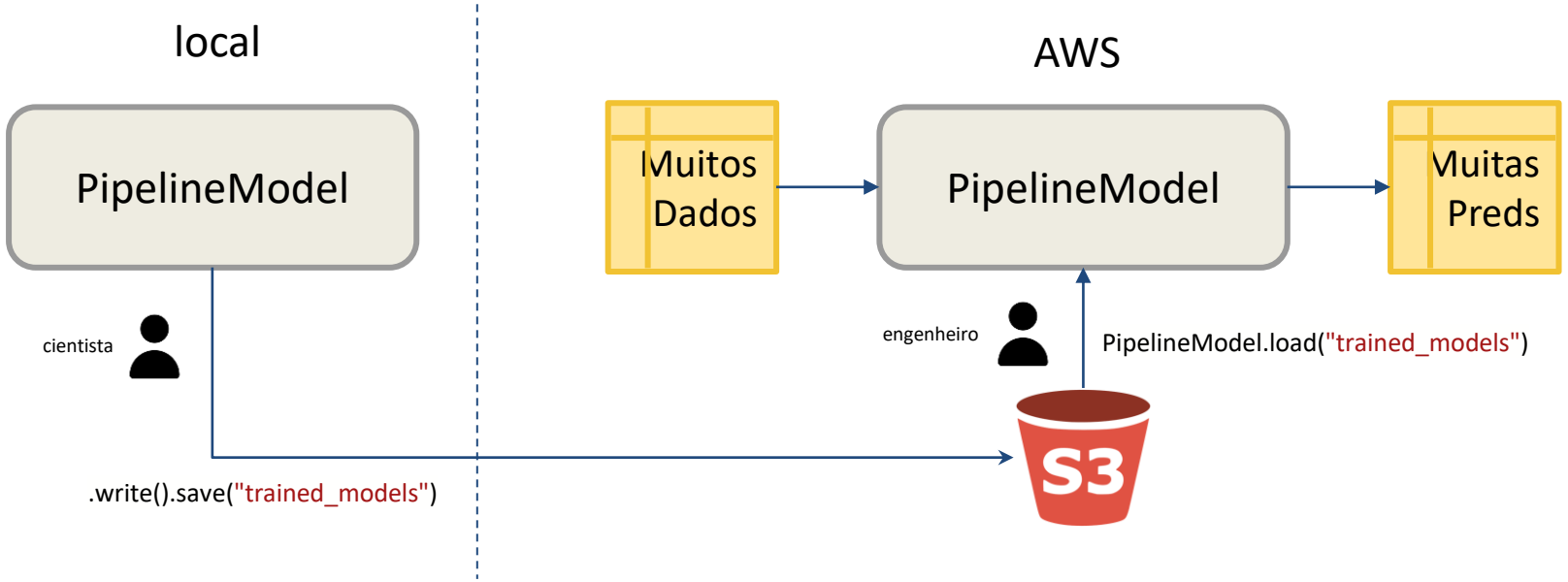
- DataFrame
- Transformer
- Estimator
- **Pipeline**
- Parameter



SparkML

Principais componentes | Pipeline em produção

- DataFrame
- Transformer
- Estimator
- **Pipeline**
- Parameter



SparkML

Links para documentações relevantes

- https://spark.apache.org/docs/3.5.1/ml-guide.html
- https://spark.apache.org/docs/latest/ml-features
- https://spark.apache.org/docs/latest/ml-classification-regression.html

Prática

Criando modelos preditivos no Spark

- Modelo de regressão linear
- Usando Pipeline
- Feature engineering
- Salvando/Carregando modelos
- Otimização de hiperparâmetros e Validação cruzada

Insper