Pandas与Spark中DataFrame详细对比

被于中华添加，被于中华最后更新于一月 04, 2019

	Pandas	Spark
工作方式	单机single machine tool，没有并行机制parallelism 不支持Hadoop，处理大量数据有瓶颈	分布式并行计算框架，内建并行机制parallelism，所有的数据和操作自动并行分布在各个集群结点上。以处理in-memory数据的方式处理distributed数据。 支持Hadoop，能处理大量数据
延迟机制	not lazy-evaluated	lazy-evaluated
内存缓存	单机缓存	persist() or cache()将转换的RDDs保存在内存
DataFrame可变性	Pandas中DataFrame是可变的	Spark中RDDs是不可变的，因此DataFrame也是不可变的
创建	从spark_df转换：pandas_df = spark_df.toPandas()	从pandas_df转换：spark_df = SQLContext.createDataFrame(pandas_df) 另外，createDataFrame支持从list转换spark_df，其中list元素可以为tuple，dict，rdd
	list，dict，ndarray转换	已有的RDDs转换
	CSV数据集读取	结构化数据文件读取
	HDF5读取	JSON数据集读取
	EXCEL读取	Hive表读取
	读SQL/数据库	外部数据库读取
index索引	自动创建	没有index索引，若需要需要额外创建该列
行结构	Series结构，属于Pandas DataFrame结构	Row结构，属于Spark DataFrame结构
列结构	Series结构，属于Pandas DataFrame结构	Column结构，属于Spark DataFrame结构，如：DataFrame[name: string]
列名称	不允许重名	允许重名 修改列名采用alias方法
列添加	df[“xx”] = 0	df.withColumn(“xx”, 0).show() 会报错 from pyspark.sql import functions df.withColumn(“xx”, functions.lit(0)).show()
列修改	原来有df[“xx”]列，df[“xx”] = 1	原来有df[“xx”]列，df.withColumn(“xx”, 1).show()
显示		df 不输出具体内容，输出具体内容用show方法 输出形式：DataFrame[age: bigint, name: string]
	df 输出具体内容	df.show() 输出具体内容
	没有树结构输出形式	以树的形式打印概要：df.printSchema()
		df.collect()
排序	df.sort_index() 按轴进行排序，指定axis	

	df.sort() 在列中按值进行排序	df.sort() 在列中按值进行排序
选择或切片	df.name 输出具体内容	df[] 不输出具体内容，输出具体内容用show方法 df[“name”] 不输出具体内容，输出具体内容用show方法
	df[] 输出具体内容， df[“name”] 输出具体内容	df.select() 选择一列或多列 df.select(“name”) 切片 df.select(df[‘name’], df[‘age’]+1)
	df[0] df.ix[0]	df.first()
	df.head(2)	df.head(2)或者df.take(2)
	df.tail(2)	
	切片 df.ix[:3]或者df.ix[:”xx”]或者 df[:”xx”]	
	df.loc[] 通过标签进行选择	
	df.iloc[] 通过位置进行选择	
过滤	df[df[‘age’]>21]	df.filter(df[‘age’]>21) 或者 df.where(df[‘age’]>21)
整合	df.groupby(“age”) df.groupby(“A”).avg(“B”)	df.groupBy(“age”) df.groupBy(“A”).avg(“B”).show() 应用单个函数 from pyspark.sql import functions df.groupBy(“A”).agg(functions.avg(“B”), functions.min(“B”), functions.max(“B”)).show() 应用多个函数
统计	df.count() 输出每一列的非空行数	df.count() 输出总行数
	df.describe() 描述某些列的count, mean, std, min, 25%, 50%, 75%, max	df.describe() 描述某些列的count, mean, stddev, min, max
合并	Pandas下有concat方法，支持轴向合并	
	Pandas下有merge方法，支持多列合并 同名列自动添加后缀，对应键仅保留一份副本	Spark下有join方法即df.join() 同名列不自动添加后缀，只有键值完全匹配才保留一份副本
	df.join() 支持多列合并	
	df.append() 支持多行合并	
缺失数据处理	对缺失数据自动添加NaNs	不自动添加NaNs，且不抛出错误
	fillna函数：df.fillna()	fillna函数：df.na.fill()
	dropna函数：df.dropna()	dropna函数：df.na.drop()
SQL语句	import sqlite3 pd.read_sql(“SELECT name, age FROM people WHERE age >= 13 AND age <= 19″)	表格注册：把DataFrame结构注册成SQL语句使用类型 df.registerTempTable(“people”) 或者 sqlContext.registerDataFrameAsTable(df, “people”) sqlContext.sql(“SELECT name, age FROM people WHERE age >= 13 AND age <= 19″)
		功能注册：把函数注册成SQL语句使用类型 sqlContext.registerFunction(“stringLengthString”, lambda x: len(x)) sqlContext.sql(“SELECT stringLengthString(‘test’)”)

两者互相转换	pandas_df = spark_df.toPandas()	spark_df = sqlContext.createDataFrame(pandas_df)
函数应用	df.apply(f）将df的每一列应用函数f	df.foreach(f) 或者 df.rdd.foreach(f) 将df的每一列应用函数f df.foreachPartition(f) 或者 df.rdd.foreachPartition(f) 将df的每一块应用函数f
map-reduce操作	map(func, list)，reduce(func, list) 返回类型seq	df.map(func)，df.reduce(func) 返回类型seqRDDs
diff操作	有diff操作，处理时间序列数据（Pandas会对比当前行与上一行）	没有diff操作（Spark的上下行是相互独立，分布式存储的）

赞同 成为第一个赞同者