DATA WRANGLING WITH PYSPARK FOR DATA SCIENTISTS WHO KNOW PANDAS

# LOAD CSV

Pandas

df= pd.read\_csv("mtcars.csv")

PySpark

df = spark.read \

.options(header=True, inferSchema=True) \

.csv("mtcars.csv " )

# VIEW DATAFRAME

Pandas

df

df.head(10)

PySpark

df . show( )

df.show(10)

# COLUMNS AND DATA TYPES

Pandas

df.columns

df.dtypes

PySpark

df.columns

df.dtypes

# RENAME COLUMNS

Pandas

df .columns = [‘a', ‘b',’c’]

df.rename(columns {'old': ‘new’ })

PySpark

Df.toDF (‘a', ‘b',’c’)

df.withColumnRenamed('old', ‘new’ )

# DROP COLUMN

Pandas

df.drop('mpg', axis=1)

PySpark

df.drop('mpg')

# FILTERING

Pandas

df[df.mpg < 20]

df[(df.mpg < 20) & (df.cyl == 6)]

PySpark

df[df.mpg < 20]

df[(df.mpg < 20) & (df.cyl == 6)]

# ADD COLUMN

Pandas

df[‘new' |] = 1 / df.mpg

PySpark

df.withColumn('new', 1 / df.mpg)

# FILL NULLS

Pandas

df.fillna(0) + Many more options

PySpark

df.fillna(0)

# AGGREGATION

Pandas

df.groupby([ ‘cyl’,’gear’])\

.agg({ ‘mpg’: ‘mean’, ‘disp’:‘min' })

PySpark

df.groupby([ ‘cyl’,’gear’])\

.agg({ ‘mpg’: ‘mean’, ‘disp’:‘min' })

# STANDARD TRANSFORMATIONS

Pandas

Import numpy as **np**

df['logVar1'] = **np.**log(df.var1)

PySpark

import pyspark.sgl.functions as **F** df.withColumn( 'logVar1', **F.**log(df.var1))

# ROW CONDITIONAL STATEMENTS

Pandas

df['cond'] =df.apply(lambda r:

1 if r.mpg > 20 else 2 if r.cyl == 6 else 3, axis=1)

PySpark

Import pyspark.sql.functions as F

df.withColumn( ‘cond’, \

F.when(df.mpg > 20, 1) \

.when(df.cyl == 6, 2)\

.otherwise(3))

# PYTHON WHEN REQUIRED

Pandas

df[‘disp1'] = df.disp.apply(lambda x: x+1)

PySpark

Import pyspark.sql.functions as F

from pyspark.sgl.types import DoubleType

**fn** = **F.udf**(lambda x: x+1, **DoubleType**()) #return type

df.withColumn('disp1', **fn**(df.disp) )

# MERGE/JOIN DATAFRAMES

# Pandas

left.merge(right, on='key' ) left.merge(right, left\_on='a', right\_on="b')

# PySpark

left.join(right, on='key' )

left. join(right, left.a == right.b)

# PIVOT TABLE

# Pandas

pd.pivot\_table(df, values='D', \ index=[‘A’, °B’'], columns=['C'], \ age func=np.sum)

# PySpark

df.groupBy("A ,“B”}.pivot("C" ).sum("D")

# SUMMARY STATISTICS

# Pandas

df.describe()

# PySpark

df.describe().show()

(only count, mean, stddev, min, max)

df.**selectExpr** (

"percentile approx(mpg, array(.25, .5, .75)) as mpg"

). show() #column and array of percentile

# HISTOGRAMS

# Pandas

df.hist()

# PySpark

Df.sample(False,0.1).toPandas().hist()

# SQL

# Pandas

# n/a

# PySpark

df.createOrReplaceTempView('foo')

df2 = spark.sql('select \* from foo')