**Task 1: Correlation between fields**

How to represent vectors in Spark

from pyspark.ml.linalg import Vectors

1. Dense vector

#create a vector of 4 features

Vectors.dense([1,2,3,4])

1. Sparse vector

#create a vector of 4 features

Vectors.sparse(4,[(0,1),(2,3)])

#Read data

df = spark.read.csv("/home/s\_kante/spark/data/developers\_survey\_training.csv", header='true')

#Replace IsDeveloper value with integer 1 or 0

#Approach1

df.createOrReplaceTempView("inputData")

df1 = spark.sql("SELECT CASE IsDeveloper WHEN 'Yes' THEN 1 ELSE 0 END AS IsDeveloper, CAST(YearsOfExp AS FLOAT) AS YearsOfExp, CAST(Salary AS FLOAT) AS Salary FROM inputData ");

#Approach2

from pyspark.sql import functions as F

df1 = df.select(F.when(df.IsDeveloper=="Yes",1).otherwise(0).alias("IsDeveloper"), df.YearsOfExp.cast("float"), df.Salary.cast("float"))

#Create feature vector

#Approach1

vector\_rdd = df1.rdd.map(lambda x: (Vectors.dense([x.IsDeveloper, x.YearsOfExp,x.Salary]),))

vector\_df = spark.createDataFrame(vector\_rdd, ["features"])

#Approach2

from pyspark.ml.feature import VectorAssembler

assembler = VectorAssembler(inputCols=["IsDeveloper","YearsOfExp", "Salary"], outputCol="features")

combined = assembler.transform(df1)

vector\_df = combined.select(combined.features)

#Find the correlation

from pyspark.ml.stat import Correlation

correlation1 = Correlation.corr(vector\_df, "features").head()

print("Pearson correlation matrix:\n" + str(correlation1[0]))

correlation2 = Correlation.corr(vector\_df, "features", "spearman").head()

print("Spearman correlation matrix:\n" + str(correlation2[0]))

**#Ref:** <https://stackoverflow.com/questions/39982135/apache-spark-dealing-with-case-statements>