project_profiling

```
val filePath = "bitcoin.csv"
```

FINISHED

```
filePath: String = bitcoin.csv
```

Took 31 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:14:27 PM.

```
val rawDF = spark.read
  .option("header", "true")
  .option("multiLine", "true")
  .option("inferSchema", "true")
  .option("escape", "\"")
  .csv(filePath)

z.show(rawDF)
```

SPARK JOB FINISHED

settings ▾

Index	Open	High	Low	Close	Volume_(BTC)
2011-12-31 07:52:00.0	4.39	4.39	4.39	4.39	0.45558087
2011-12-31 07:53:00.0	4.39	4.39	4.39	4.39	0.555046181987448
2011-12-31 07:54:00.0	4.39	4.39	4.39	4.39	0.654511493974895
2011-12-31 07:55:00.0	4.39	4.39	4.39	4.39	0.753976805962343
2011-12-31 07:56:00.0	4.39	4.39	4.39	4.39	0.853442117949791
2011-12-31 07:57:00.0	4.39	4.39	4.39	4.39	0.952907429937239
2011-12-31 07:58:00.0	4.39	4.39	4.39	4.39	1.05237274192469
2011-12-31 07:59:00.0	4.39	4.39	4.39	4.39	1.15183805391213

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult** ×

```
rawDF: org.apache.spark.sql.DataFrame = [Index: timestamp, Open: double ... 6 more fields]
```

Took 49 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:16 PM.

1. Column Description Infer

FINISHED

First I convert the column name to more readable names since the database lack column description.

Took 3 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:19 PM.

```
val df1 = rawDF.withColumnRenamed("timestamp", "Index")
z.show(df1)
```

SPARK JOB (http://nyu-dataproc-sw-z4gn.c.hpc-dataproc-19b8.internal:33239/jobs/job?id=3) FINISHED

settings ▾

Index	Open	High	Low	Close	Volume_(BTC)
2011-12-31 07:52:00.0	4.39	4.39	4.39	4.39	0.45558087
2011-12-31 07:53:00.0	4.39	4.39	4.39	4.39	0.555046181987448
2011-12-31 07:54:00.0	4.39	4.39	4.39	4.39	0.654511493974895
2011-12-31 07:55:00.0	4.39	4.39	4.39	4.39	0.753976805962343
2011-12-31 07:56:00.0	4.39	4.39	4.39	4.39	0.853442117949791
2011-12-31 07:57:00.0	4.39	4.39	4.39	4.39	0.952907429937239
2011-12-31 07:58:00.0	4.39	4.39	4.39	4.39	1.05237274192469
2011-12-31 07:59:00.0	4.39	4.39	4.39	4.39	1.15183805391213

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult** ×

```
df1: org.apache.spark.sql.DataFrame = [Index: timestamp, Open: double ... 6 more fields]
```

Took 1 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:20 PM.

```
import org.apache.spark.sql.functions.{min, max}
import org.apache.spark.sql.Row

val Row(minValue: Double, maxValue: Double) = df1.agg(min("Close"), max("Close")).head
```

SPARK JOB FINISHED

```
import org.apache.spark.sql.functions.{min, max}
import org.apache.spark.sql.Row
minValue: Double = 1.5
maxValue: Double = 19665.75
```

Took 9 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:29 PM.

From here we can infer the currency is US dollar

FINISHED

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:29 PM.

```
val df2 = df1.withColumnRenamed("Volume_(BTC)", "btc_volume").withColumnRenamed("Volume_(Currency)", "usd_volume")
```

FINISHED

```
df2: org.apache.spark.sql.DataFrame = [Index: timestamp, Open: double ... 6 more fields]
```

Took 1 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:30 PM.

```
z.show(df2)
```

SPARK JOB (http://nyu-dataproc-sw-z4gn.c.hpc-dataproc-19b8.internal:33239/jobs/job?id=6) FINISHED

settings ▾

Index	Open	High	Low	Close	btc_volume
2011-12-31 07:52:00.0	4.39	4.39	4.39	4.39	0.45558087
2011-12-31 07:53:00.0	4.39	4.39	4.39	4.39	0.555046181987448
2011-12-31 07:54:00.0	4.39	4.39	4.39	4.39	0.654511493974895
2011-12-31 07:55:00.0	4.39	4.39	4.39	4.39	0.753976805962343
2011-12-31 07:56:00.0	4.39	4.39	4.39	4.39	0.853442117949791
2011-12-31 07:57:00.0	4.39	4.39	4.39	4.39	0.952907429937239
2011-12-31 07:58:00.0	4.39	4.39	4.39	4.39	1.05237274192469
2011-12-31 07:59:00.0	4.39	4.39	4.39	4.39	1.15183805391213

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult** ×

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:30 PM.

2. Datatype check

FINISHED

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:30 PM.

```
df2.printSchema
```

FINISHED

```
root
 |-- Index: timestamp (nullable = true)
 |-- Open: double (nullable = true)
 |-- High: double (nullable = true)
 |-- Low: double (nullable = true)
 |-- Close: double (nullable = true)
 |-- btc_volume: double (nullable = true)
 |-- usd_volume: double (nullable = true)
 |-- Weighted_Price: double (nullable = true)
```

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:31 PM.

The data type is fine. The 'infer' option has already convert the String type timestamp to timestamp data type

FINISHED

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:31 PM.

3. statistical check

FINISHED

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:32 PM.

```
z.show(df2.describe())
```

SPARK JOB FINISHED

settings

summary	Open	High	Low	Close	btc_volume
count	4363457	4363457	4363457	4363457	4363457
mean	2751.729390252802	2753.702802844385	2749.6000101520044	2751.6859930984483	9.95276457039659
stddev	3686.9892053174726	3690.1546275862775	3683.5030550230326	3686.908384770707	31.020480762667948
min	3.8	3.8	1.5	1.5	0.0
max	19665.76	19666.0	19649.96	19665.75	5853.8521659

Took 9 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:41 PM.

```
val imputeCols = Array(
  "Open",
  "High",
  "Low",
  "Close",
  "btc_volume",
  "usd_volume",
  "Weighted_Price",
)
```

FINISHED

```
imputeCols: Array[String] = Array(Open, High, Low, Close, btc_volume, usd_volume, Weighted_Price)
```

Took 1 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:15:42 PM.

```
import org.apache.spark.ml.feature.Imputer

val imputer = new Imputer()
  .setStrategy("median")
  .setInputCols(imputeCols)
  .setOutputCols(imputeCols)

val imputedDF = imputer.fit(df2).transform(df2)

z.show(imputedDF.describe())
z.show(imputedDF)
```

SPARK JOB FINISHED

settings

summary	Open	High	Low	Close	btc_volume
count	4363457	4363457	4363457	4363457	4363457
mean	2751.729390252802	2753.702802844385	2749.6000101520044	2751.6859930984483	9.95276457039659
stddev	3686.9892053174726	3690.1546275862775	3683.5030550230326	3686.908384770707	31.020480762667948
min	3.8	3.8	1.5	1.5	0.0
max	19665.76	19666.0	19649.96	19665.75	5853.8521659

settings

Index	Open	High	Low	Close	btc_volume
2011-12-31 07:52:00.0	4.39	4.39	4.39	4.39	0.45558087
2011-12-31 07:53:00.0	4.39	4.39	4.39	4.39	0.555046181987448
2011-12-31 07:54:00.0	4.39	4.39	4.39	4.39	0.654511493974895
2011-12-31 07:55:00.0	4.39	4.39	4.39	4.39	0.753976805962343

2011-12-31 07:56:00.0	4.39	4.39	4.39	4.39	0.853442117949791
2011-12-31 07:57:00.0	4.39	4.39	4.39	4.39	0.952907429937239
2011-12-31 07:58:00.0	4.39	4.39	4.39	4.39	1.05237274192469
2011-12-31 07:59:00.0	4.39	4.39	4.39	4.39	1.15183805391213

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult** ×

```
import org.apache.spark.ml.feature.Imputer
imputer: org.apache.spark.ml.feature.Imputer = imputer_eb11547106f6
imputedDF: org.apache.spark.sql.DataFrame = [Index: timestamp, Open: double ... 6 more fields]
```

Took 27 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:16:09 PM.

4. Getting rid of extreme or abnormal values

FINISHED

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:16:09 PM.

```
imputedDF.filter($"Weighted_Price" <= 0).count
```

SPARK JOB FINISHED

```
res7: Long = 0
```

Took 8 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:16:17 PM.

```
val valueDF = imputedDF.withColumn("PctChange",($"Close"- $"Open")/$"Open")
```

FINISHED

```
valueDF: org.apache.spark.sql.DataFrame = [Index: timestamp, Open: double ... 7 more fields]
```

Took 1 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:16:18 PM.

```
z.show(valueDF)
z.show(valueDF.describe())
```

SPARK JOB FINISHED

settings ▾

Index	Open	High	Low	Close	btc_volume	usd_volume
2011-12-31 07:52:00.0	4.39	4.39	4.39	4.39	0.45558087	2.000000019
2011-12-31 07:53:00.0	4.39	4.39	4.39	4.39	0.555046181987448	2.436652738
2011-12-31 07:54:00.0	4.39	4.39	4.39	4.39	0.654511493974895	2.873305458
2011-12-31 07:55:00.0	4.39	4.39	4.39	4.39	0.753976805962343	3.309958178
2011-12-31 07:56:00.0	4.39	4.39	4.39	4.39	0.853442117949791	3.746610897
2011-12-31 07:57:00.0	4.39	4.39	4.39	4.39	0.952907429937239	4.183263617
2011-12-31 07:58:00.0	4.39	4.39	4.39	4.39	1.05237274192469	4.619916337
2011-12-31 07:59:00.0	4.39	4.39	4.39	4.39	1.15183805391213	5.056569056

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult** ×

settings ▾

summary	Open	High	Low	Close	btc_volume	usd_volume
count	4363457	4363457	4363457	4363457	4363457	4363457
mean	2751.729390252802	2753.702802844385	2749.6000101520044	2751.6859930984483	9.95276457039659	21047.59911
stddev	3686.9892053174726	3690.1546275862775	3683.5030550230326	3686.908384770707	31.020480762667948	86491.63548
min	3.8	3.8	1.5	1.5	0.0	0.0
max	19665.76	19666.0	19649.96	19665.75	5853.8521659	7569437.061

Took 12 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:16:30 PM.

The percentage of price change with one minute is reasonable. FINISHED

At most 41% increase and less than 100% decrease. Standard deviation is 0.2%.

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:16:30 PM.

```
import org.apache.spark.sql.expressions.Window

val w = Window.orderBy("Index")
```

FINISHED

```
import org.apache.spark.sql.expressions.Window
w: org.apache.spark.sql.expressions.WindowSpec = org.apache.spark.sql.expressions.WindowSpec@2fcb4b80
```

Took 1 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:16:31 PM.

```
val df3 = imputedDF.withColumn("return", (col("Close") - lag("Close", 1).over(w)) / lag("close", 1).over(w))
```

FINISHED

```
df3: org.apache.spark.sql.DataFrame = [Index: timestamp, Open: double ... 7 more fields]
```

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:16:31 PM.

```
z.show(df3)
```

SPARK JOB FINISHED

settings ▾

Index	Open	High	Low	Close	btc_volume	usd_volume
2011-12-31 07:52:00.0	4.39	4.39	4.39	4.39	0.45558087	2.00000001
2011-12-31 07:53:00.0	4.39	4.39	4.39	4.39	0.555046181987448	2.43665273
2011-12-31 07:54:00.0	4.39	4.39	4.39	4.39	0.654511493974895	2.87330545
2011-12-31 07:55:00.0	4.39	4.39	4.39	4.39	0.753976805962343	3.30995817
2011-12-31 07:56:00.0	4.39	4.39	4.39	4.39	0.853442117949791	3.74661089
2011-12-31 07:57:00.0	4.39	4.39	4.39	4.39	0.952907429937239	4.18326361
2011-12-31 07:58:00.0	4.39	4.39	4.39	4.39	1.05237274192469	4.61991633
2011-12-31 07:59:00.0	4.39	4.39	4.39	4.39	1.15183805391213	5.05656905

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult** ×

Took 29 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:17:00 PM.

```
z.show(df3.describe())
```

SPARK JOB FINISHED

settings ▾

summary	Open	High	Low	Close	btc_volume	usd_volume
count	4363457	4363457	4363457	4363457	4363457	4363457
mean	2751.729390252802	2753.702802844385	2749.6000101520044	2751.6859930984483	9.95276457039659	21047.59911
stddev	3686.9892053174726	3690.1546275862775	3683.5030550230326	3686.908384770707	31.020480762667948	86491.63548
min	3.8	3.8	1.5	1.5	0.0	0.0
max	19665.76	19666.0	19649.96	19665.75	5853.8521659	7569437.061

Took 24 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:17:24 PM.

```
val df4 = df3.na.drop("any")
```

FINISHED

```
df4: org.apache.spark.sql.DataFrame = [Index: timestamp, Open: double ... 7 more fields]
```

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:17:25 PM.

```
z.show(df4)
```

SPARK JOB FINISHED

settings ▾

Index	Open	High	Low	Close	btc_volume	usd_volume
2011-12-31 07:55:00.0	4.39	4.39	4.39	4.39	0.753976805962343	3.30995817
2011-12-31 07:56:00.0	4.39	4.39	4.39	4.39	0.853442117949791	3.74661089
2011-12-31 07:57:00.0	4.39	4.39	4.39	4.39	0.952907429937239	4.18326361
2011-12-31 07:58:00.0	4.39	4.39	4.39	4.39	1.05237274192469	4.61991633
2011-12-31 07:59:00.0	4.39	4.39	4.39	4.39	1.15183805391213	5.05656905
2011-12-31 08:00:00.0	4.39	4.39	4.39	4.39	1.25130336589958	5.49322177
2011-12-31 08:01:00.0	4.39	4.39	4.39	4.39	1.35076867788703	5.92987449
2011-12-31 08:02:00.0	4.39	4.39	4.39	4.39	1.45023398987448	6.36652721
2011-12-31 08:03:00.0	4.39	4.39	4.39	4.39	1.54969930186192	6.80317993

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult**

Took 23 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:17:48 PM.

```
val df5 = df4.withColumn("Volatility", $"High"-$"Low")
z.show(df5)
```

SPARK JOB FINISHED

settings ▾

Index	Open	High	Low	Close	btc_volume	usd_volume
11:32:00.0						
2011-12-31 11:33:00.0	4.39	4.39	4.39	4.39	22.4374148192259	98.5002510564019
2011-12-31 11:34:00.0	4.39	4.39	4.39	4.39	22.5368801312134	98.9369037760268
2011-12-31 11:35:00.0	4.39	4.39	4.39	4.39	22.6363454432008	99.3735564956517
2011-12-31 11:36:00.0	4.39	4.39	4.39	4.39	22.7358107551883	99.8102092152766
2011-12-31 11:37:00.0	4.39	4.39	4.39	4.39	22.8352760671757	100.246861934901

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult**

```
df5: org.apache.spark.sql.DataFrame = [Index: timestamp, Open: double ... 8 more fields]
```

Took 27 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:27:53 PM. (outdated)

```
z.show(df5.describe())
```

SPARK JOB FINISHED

settings ▾

ry	Open	High	Low	Close	btc_volume	usd_volume	Weight
	4363456	4363456	4363456	4363456	4363456	4363456	436345
	2751.7300198774374	2753.703432921279	2749.600639288636	2751.6866227131372	9.952766746924485	21047.603940597306	2751.66
	3686.989393221721	3690.154815717843	3683.5032426696835	3686.9085726680382	31.0204839840625	86491.64481331731	3686.88
	3.8	3.8	1.5	1.5	0.0	0.0	3.8
	19665.76	19666.0	19649.96	19665.75	5853.8521659	7569437.0613	19663.2

Took 24 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:29:09 PM. (outdated)

```
val outputPath = "bitcoin-clean.parquet"
```

FINISHED

```
outputPath: String = bitcoin-clean.parquet
```

Took 0 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:30:40 PM.

```
df5.write.mode("overwrite").parquet(outputPath)
```

SPARK JOB FINISHED

Took 32 sec. Last updated by ls6211_nyu_edu at December 04 2023, 11:31:16 PM.

FINISHED