```
from pyspark.sql import SparkSession
from pyspark.sql.functions import col, mean
from pyspark.mllib.stat import Statistics

spark = SparkSession.builder.appName("BigDataCleaningEDA").getOrCreate()

# Load dataset as RDD
rdd = spark.sparkContext.textFile("/content/drive/MyDrive/bigdata.csv")

header = rdd.first()
rdd = rdd.filter(lambda row: row != header)

def parse_row(row):
    parts = row.split(",")
    try:
        return (int(parts[0]),  # ID
                parts[1],        # Name
                int(parts[2]) if parts[2] else None,  # Age
                float(parts[3]) if parts[3] else None,  # Salary
                int(parts[4]),  # Experience
                parts[5])       # Department
    except:
        return None

rdd = rdd.map(parse_row).filter(lambda x: x is not None)

# Convert RDD to DataFrame
columns = ["ID", "Name", "Age", "Salary", "Experience", "Department"]
df = spark.createDataFrame(rdd, columns)

age_mean = df.select(mean(col("Age"))).collect()[0][0]
salary_mean = df.select(mean(col("Salary"))).collect()[0][0]
df = df.fillna({"Age": age_mean, "Salary": salary_mean})

df.show()

numeric_rdd = df.select("Age", "Salary", "Experience").rdd.map(lambda row: [row.Age, row.Salary, row.Experience])

summary = Statistics.colStats(numeric_rdd)

print(f"Mean: {summary.mean()}")
print(f"Variance: {summary.variance()}")
print(f"Min: {summary.min()}")
print(f"Max: {summary.max()}")

spark.stop()
```

```
+---+---------+---+-----------------+----------+----------+
| ID|     Name|Age|           Salary|Experience|Department|
+---+---------+---+-----------------+----------+----------+
|  1|   Elijah| 37|          41624.0|        27|     Sales|
|  2|   Olivia| 40|          32971.0|         1|        HR|
|  3|   Sophia| 32|          73881.0|        27|     Sales|
|  4|     Noah| 25|         110157.0|        13|        IT|
|  5|   Elijah| 44|         100639.0|         3|        IT|
|  6|Charlotte| 52|          83540.0|        18|        IT|
|  7|     Liam| 40|         141664.0|        13|     Sales|
|  8|   Elijah| 26|90143.08625555555|        23|     Sales|
|  9|   Sophia| 48|          96345.0|        22|     Sales|
| 10|     Noah| 50|          83213.0|        11| Marketing|
| 11|   Olivia| 34|          56221.0|        28|        HR|
| 12|   Sophia| 56|         105675.0|        23|        HR|
| 13|   Sophia| 30|90143.08625555555|        10| Marketing|
| 14|     Liam| 39|90143.08625555555|         1| Marketing|
| 15|      Ava| 36|          69673.0|        25|     Sales|
| 16|     Noah| 45|90143.08625555555|        18| Marketing|
| 17|     Emma| 52|          80548.0|        27| Marketing|
| 18|   Sophia| 28|90143.08625555555|        21|        IT|
| 19|      Ava| 30|          60715.0|        17|        HR|
| 20|     Emma| 37|90143.08625555555|         5|     Sales|
+---+---------+---+-----------------+----------+----------+
only showing top 20 rows

Mean: [4.04397200e+01 9.01430863e+04 1.74804600e+01]
Variance: [1.08096907e+02 1.07923748e+09 9.62958211e+01]
Min: [2.2e+01 3.0e+04 1.0e+00]
Max: [5.90000e+01 1.49998e+05 3.40000e+01]
```