Assignment 03

Bhargavi Manyala

September 24, 2025

> In my code, I am casting salary and experience columns to float and then using approxQuantile to compute medians for SALARY_FROM, SALARY_TO, and SALARY. Can you explain what this casting step and the median calculation are doing, and why they are necessary before filling missing values?

- Great question! Let me break down both the casting and median calculation steps:

1. Casting to Float

```
df = df.withColumn("SALARY_FROM", col("SALARY_FROM").cast("float"))
df = df.withColumn("SALARY_TO", col("SALARY_TO").cast("float"))
df = df.withColumn("SALARY", col("SALARY").cast("float"))

salary_from_median = df.approxQuantile("SALARY_FROM", [0.5], 0.01)[0]
salary_to_median = df.approxQuantile("SALARY_TO", [0.5], 0.01)[0]
salary_median = df.approxQuantile("SALARY", [0.5], 0.01)[0]
```

- Why casting is necessary:
- Original data type: These columns are likely stored as strings in the raw data
- Mathematical operations: You can't compute medians, averages, or other statistics on string data
- Null handling: String columns might have values like “N/* A”, “null”, or empty strings that need to become actual null values
- Performance: Numeric operations on float columns are much faster than string operations

2. Median Calculation with approxQuantile

- What approxQuantile does: