Если рассмотреть вариант с изменяемым размером данных, то временная сложность операций будет зависеть от количества строк n*n* и количества столбцов m*m*. Давайте разберем каждую операцию и оценим её сложность.

**1. Создание данных**

* Генерация оценок: O(n×m)*O*(*n*×*m*).
* Создание DataFrame: O(n×m)*O*(*n*×*m*).

**Итого**: O(n×m)*O*(*n*×*m*).

**2. Вывод первых строк (df.head())**

* Операция head() возвращает первые k*k* строк (по умолчанию k=5*k*=5). Это занимает O(k)*O*(*k*), что можно считать O(1)*O*(1), если k*k* фиксировано.

**Итого**: O(1)*O*(1).

**3. Вычисление средней оценки по каждому предмету**

* Для каждого из m*m* столбцов вычисляется среднее значение. Для одного столбца это O(n)*O*(*n*).
* Общая сложность: O(m×n)*O*(*m*×*n*).

**Итого**: O(n×m)*O*(*n*×*m*).

**4. Вычисление медианной оценки по каждому предмету**

* Для каждого из m*m* столбцов:
  + Сортировка столбца: O(nlog⁡n)*O*(*n*log*n*).
  + Нахождение медианы: O(1)*O*(1).
* Общая сложность: O(m×nlog⁡n)*O*(*m*×*n*log*n*).

**Итого**: O(nlog⁡n×m)*O*(*n*log*n*×*m*).

**5. Вычисление Q1 и Q3 для оценок по математике**

* Для одного столбца:
  + Сортировка: O(nlog⁡n)*O*(*n*log*n*).
  + Нахождение Q1 и Q3: O(1)*O*(1).
* Общая сложность: O(nlog⁡n)*O*(*n*log*n*).

**Итого**: O(nlog⁡n)*O*(*n*log*n*).

**6. Вычисление стандартного отклонения**

* Для каждого из m*m* столбцов:
  + Вычисление среднего: O(n)*O*(*n*).
  + Вычисление суммы квадратов отклонений: O(n)*O*(*n*).
  + Вычисление стандартного отклонения: O(1)*O*(1).
* Общая сложность: O(m×n)*O*(*m*×*n*).

**Итого**: O(n×m)*O*(*n*×*m*).

**Итоговая сложность**

Если объединить все операции, то общая временная сложность алгоритма будет определяться самой "тяжелой" операцией. В данном случае это **сортировка для медианы и квартилей**, которая имеет сложность O(nlog⁡n×m)*O*(*n*log*n*×*m*).

**Общая временная сложность:**

O(nlog⁡n×m)*O*(*n*log*n*×*m*)

**Обоснование:**

* **Доминирующая операция**: Сортировка для медианы и квартилей (O(nlog⁡n×m)*O*(*n*log*n*×*m*)).
* Остальные операции (среднее, стандартное отклонение) имеют меньшую сложность (O(n×m)*O*(*n*×*m*)) и не влияют на итоговую оценку.

**Пример для n=1000*n*=1000 и m=5*m*=5:**

* Сортировка для медианы: 1000log⁡1000×5≈5000×10=500001000log1000×5≈5000×10=50000.
* Среднее и стандартное отклонение: 1000×5=50001000×5=5000.

Таким образом, сортировка будет самой затратной операцией.

**Улучшение:**

Если нужно оптимизировать алгоритм, можно:

1. Использовать более быстрые алгоритмы сортировки (например, Timsort, который используется в Python, уже достаточно оптимизирован).
2. Кэшировать результаты сортировки, если медиана и квартили вычисляются для одного и того же столбца несколько раз.

**Заключение:**

Для изменяемого размера данных сложность алгоритма составляет O(nlog⁡n×m)*O*(*n*log*n*×*m*), где n*n* — количество строк, а m*m* — количество столбцов.