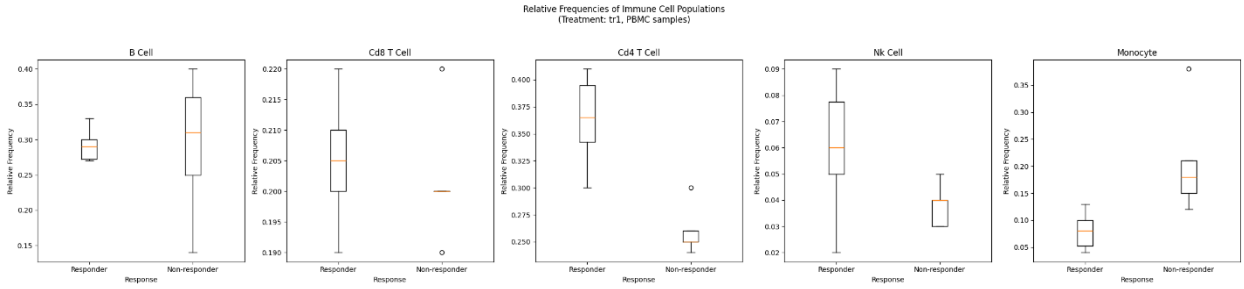
Python:

2a. Boxplots comparing relative blood cell percentage fro Responders vs Non-Responders



2b. To determine which immune cell populations exhibit significant differences in relative frequencies between responders and non-responders to treatment tr1, we performed a Mann-Whitney U test on PBMC (blood) samples. Considering a threshold for significance of p = 0.05, 0ur analysis identified two immune cell populations with statistically significant differences:

1. **NK cells** (Natural Killer cells):
    - Mann-Whitney U statistic: 35.0
    - p-value: 0.0328
2. **Monocytes**:
    - Mann-Whitney U statistic: 3.0
    - p-value: 0.0150

These findings indicate that NK cells and monocytes have significantly different relative frequencies between responders and non-responders, suggesting their potential utility as biomarkers for predicting patient response to treatment tr1.

Database:

1. The question here is to decide between having multiple databases, (e.g. a two-database setup where the first manages the “project” and “subject” fields, containing information such as project id, subject id, subject demographics etc while the second would manage samples) or using a single large database where size/complexity is managed through indexing and sharding. For this circumstance, I would lean towards using a single database with the project being the primary index and treatment being a secondary index. If the ‘project’ or ‘subject’ information was

more substantial, for example including large text sections, then I would be more inclined towards the first approach, however, in this case it seems like unnecessary overhead.

I would structure the database with something like the following tables:

```
CREATE TABLE Subjects (
  subject_id VARCHAR(50) PRIMARY KEY,
  project_id VARCHAR(50),
  condition VARCHAR(100),
  age INTEGER,
  sex VARCHAR (10),
  treatment VARCHAR(50),
  response VARCHAR (10),
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

CREATE TABLE Samples (
  sample_id VARCHAR(50) PRIMARY KEY,
  subject_id VARCHAR(50),
  sample_type VARCHAR(50),
  time_from_treatment_start DECIMAL(5,2),
  b_cell INTEGER,
  cd8_t_cell INTEGER,
  cd4_t_cell INTEGER,
  nk_cell INTEGER,
  monocyte INTEGER,
  FOREIGN KEY (subject_id) REFERENCES Subjects(subject_id)
);
```

2. The advantages of using a database, as opposed to leaving it in a single large csv for example, include:
    1. Version control
    2. Read/Write Access control
        a. Allows for easy concurrent usage
            i. Guaranteed atomic operations, consistency, isolation etc.
    3. Cloud options for scalability/redundance
    4. Efficient lookups and easy to use integration into different scripts
        a. Reusable and portable lookups
3. SELECT condition, COUNT(subject_id) AS subject_count FROM Subjects GROUP BY condition
4. SELECT s.* FROM Samples s JOIN Subjects p ON s.subject_id = p.subject_id WHERE p.condition = 'melanoma' AND s.sample_type = 'PBMC' AND s.time_from_treatment_start = 0 AND p.treatment = 'tr1'
5. SELECT sub.project_id, COUNT(sam.sample_id) AS sample_count FROM Samples AS sam JOIN Subjects AS sub ON sam.subject_id = sub.subject_id GROUP BY sub.project_id
6. SELECT sub.response, COUNT(sam.sample_id) AS sample_count FROM Samples AS sam JOIN Subjects AS sub ON sam.subject_id = sub.subject_id GROUP BY sub.response
7. SELECT sub.sex, COUNT(sam.sample_id) AS sample_count FROM Samples AS sam JOIN Subjects AS sub ON sam.subject_id = sub.subject_id GROUP BY sub. sex