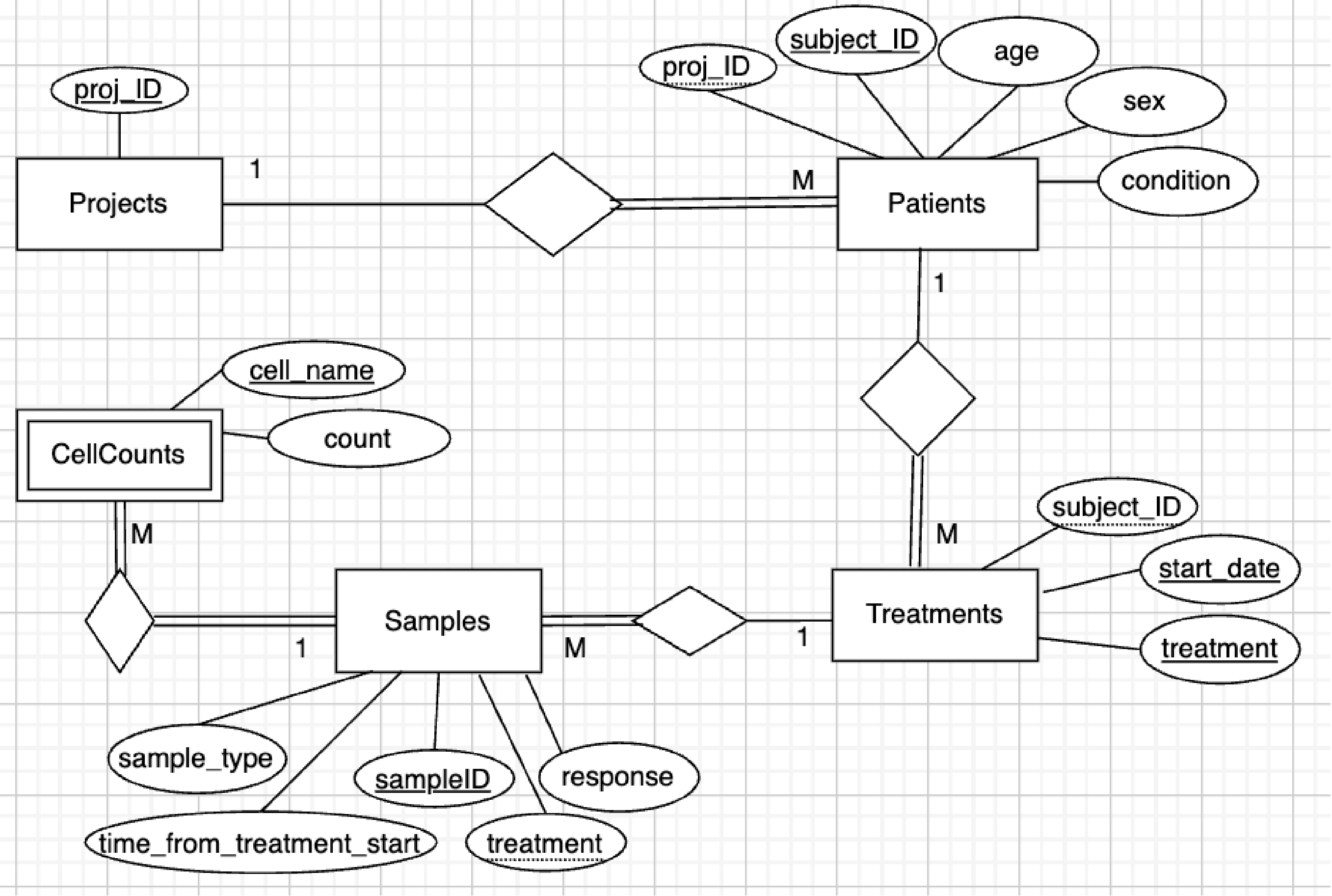
1: How would you design a database to capture the type of information and data in cell-count.csv?

To design this database, I must think about keeping the schema scalable, and easily queried for data analysis. To do so, I'll design an ER-diagram that shows the relationships between five entities: Projects, Patients, Treatments, Samples, and CellCounts.

Here is a prototype schema:



A high level overview:

- **Projects <-> Patients**: A project can hold data for many patients but a patient must work with exactly one project. This was inferred from the data in cell-count.csv, but could be easily tweaked to let patients be included in multiple projects.
- **Patients <-> Treatments**: A patient can have many treatments but a treatment must be attributed to exactly one patient. This design could expand future analysis by comparing samples of repeated treatments.
- **Treatments <-> Samples**: A treatment can have many samples taken, but a sample must belong to exactly one treatment. Again, supporting a new dimension of data analysis.

Because treatments store start_date, samples can easily compute 'time_from_treatment_start'.

- **Samples <-> CellCounts**: A sample must have at least 1 cell count (though will always have five with proper software implementation), and a cell count is weakly dependent on exactly one sample.

This setup effectively separates concerns into a neat hierarchy of data. It's scalable for hundreds of projects where each project could have thousands of samples. It's also easy to perform analysis on data points by querying samples from the patients table, and applying filters during the joins (like condition = 'melanoma', and treatment = 'tr1', etc.).

A few notes:

- When implementing this ER-diagram, some entities might benefit from more information. Projects for example could be expanded to include a project name and description.
- Keys were adjusted to be an int to save space and to enable auto-increment and therefore offload some referential-integrity to the DBMS.

2: What would be some advantages in capturing this information in a database?

- We still have all of the flexibility of the .csv file, but with more potential for scalability, because excel and other spreadsheets tend to slow down at large volumes of data. By capturing the information in some DBMS, querying data (especially complex data) is optimized considerably.
- The database also lets us easily represent more complex relationships. For example, we could include the relative frequency in the CellCounts entity.
- Our DBMS would automatically enforce referential integrity, which could keep data cleaner per the rules that we want to define in the schema.
- Better integration in software and pipelines. For example C#'s LINQ works incredibly well with databases, further extending scalability.

3: Based on the schema you provide in (1), please write a query to summarize the number of subjects available for each condition.

- All queries will be in MySQL:

```
SELECT condition, COUNT(DISTINCT subject_ID)
AS subject_count FROM Patients
GROUP BY condition;
```

4: Please write a query that returns all melanoma PBMC samples at baseline (time_from_treatment_start is 0) from patients who have treatment tr1.

```
SELECT s.*
FROM Samples s
JOIN Treatments t ON s.treatment = t.treatment
JOIN Patients p ON t.subject_ID = p.subject_ID
WHERE p.condition = 'melanoma'
AND t.treatment = 'tr1'
AND s.sampleType = 'PBMC'
AND s.time_from_treatment_start = 0;
```

5: Please write queries to provide these following further breakdowns for the samples in (4):

a. How many samples from each project

```
SELECT p.proj_ID, COUNT(*) AS sample_count
FROM Samples s
JOIN Treatments t ON s.treatment = t.treatment
JOIN Patients p ON t.subject_ID = p.subject_ID
WHERE p.condition = 'melanoma'
AND t.treatment = 'tr1'
AND s.sample_type = 'PBMC'
AND s.time_from_treatment_start = 0
GROUP BY p.proj_ID;
```

b. How many responders/non-responders

```
SELECT response, COUNT(*) AS response_count
FROM Samples s
JOIN Treatments t ON s.treatment = t.treatment
JOIN Patients p ON t.subject_ID = p.subject_ID
WHERE p.condition = 'melanoma'
AND t.treatment = 'tr1'
AND s.sample_type = 'PBMC'
AND s.time_from_treatment_start = 0
GROUP BY s.response;
```

c. How many males, females

```
SELECT sex, COUNT(*) AS sex_count
FROM Samples s
JOIN Treatments t ON s.treatment = t.treatment
JOIN Patients p ON t.subject_ID = p.subject_ID
WHERE p.condition = 'melanoma'
AND t.treatment = 'tr1'
AND s.sample_type = 'PBMC'
AND s.time_from_treatment_start = 0
GROUP BY p.sex;
```