Group Semester Projects

The goal of *Applied Bioinformatics* is to train students in practical analysis of complex biological data, using microbiome data as an entry point. Thus, the best way to assess these skills (computational competencies, quantitative analysis, and application of domain expertise) will be with a practical application. The majority of the grade (see course overview document) will be based on this group semester project.

Students will form small teams, ideally based on *shared interests* and *complementary skill sets* (e.g., from different programs/backgrounds), to solve a real problem using real biological data and a suite of bioinformatics skills learned in this course. Teams will analyze their data by applying data analysis and critical thinking skills learned during the course (see Specific Tasks at the end of this document). All students are expected to run the analysis workflow and contribute to all parts of the assessment, but balanced delegation of other tasks (e.g., different sections of the written report; different analytical sections) is allowed, provided that all students participate in all tasks, delegation is balanced and fair, and justification is provided in the work plan.

During the first few weeks of the course:

1. **Form a team of 3-5 students**. Students will be asked to self-organize into groups based on common interests, and final groups will be approved by the instructor(s). Students will self-identify as more interested in **human/health** or **food/agricultural** microbiomes.
2. **Project challenges** will be presented to each group of students. Each group will get a unique challenge, and a unique dataset! Start planning immediately to think over the problem as a group and brainstorm ways that the problem might be solved. *Do not worry if you do not have any clue right now*, just jot down your notes and revisit later in the semester. The goal is for groups to work on their project during each week of the semester (including class time) to apply each week's lessons to their project dataset. Solutions will become more apparent as the semester proceeds.

By the end of Module 2 (mid-term!):

1. Groups should have met on a weekly basis (minimum) to discuss the challenge and to start planning solutions.
2. **Midterm Report** (10% of overall project grade) is due (see timeline)
    a. This is a brief (1-2 page) report giving:
        i. A short summary of the problem (one paragraph)
        ii. A short description of what you have done so far (one paragraph; and provide a link to a GitHub repository containing all analysis code used so far).
        iii. A short description and Gantt chart describing your plans for addressing the problem by the end of the semester (one paragraph plus chart; hint: make sure this aligns with the course timeline, including final presentation day!)
    b. What should have been done by now:
        i. Data should be successfully imported and explored in a Jupyter notebook.
        ii. Sequence data should be demultiplexed, denoised or clustered, and quality filtered, as appropriate (i.e., apply procedures learned in Module 2 of this course).

iii. Code should be submitted to GitHub in a new project repository, which should be organized, concise, and well-annotated. Any notebooks should be fully executable (including data download and importing).

c. The grade will be assigned to the entire group, based on:
   i. A demonstrated understanding of the problem (20%).
   ii. A clear and systematic plan for how to address the problem (30%).
   iii. A clear and feasible timeline (10%).
   iv. The quality of GitHub code to date (40%; see section 2.b. above).

3. Groups are encouraged to schedule a meeting with the instructor(s) to discuss their progress, raise any concerns, and brainstorm ideas for analysis.

At the end of the semester (week 14):

1. **Final written reports are due on the last day of class**! (see below)
2. Groups will briefly present their projects in class (ungraded), as an oral presentation with powerpoint slides.
   a. Presentations will last 10-15 minutes maximum (strict limit!).
   b. Groups should present:
      i. An overview of their challenge (1-2 min).
      ii. An overview of their methodology (2-4 min; high-level detail).
      iii. Their main results/solution (5-7 min)
      iv. Shortly describe their team structure and strategy for addressing their problem (e.g., how tasks were delegated in their team) (1-2 min).
   c. Each member of the group should take turns presenting.
   d. ***These presentations are not assigned a grade*** (phew!) because this is not a course in formal presentation skills. The goal is instead to have a fun, informal, but concise discussion of what everyone has been doing for the past 14 weeks!

Final Report (due week 14):

The final report is the culmination of the semester work, and will comprise two components:

1. The GitHub repository for the project, worth 50% of the total project grade.
2. A final report, in the form of a research article (format below), worth 40% of the total project grade.

Part 1: Analysis code (GitHub repository)

Each project will have a GitHub repository (created by the instructors). Groups should perform all project steps (data analysis and documentation) in these repositories. All source code and notebooks should be committed to that repository.

1. To build on this repository (and demonstrate ability to use GitHub by the final) each student should:
   a. Fork the repository to create local branches with their own GitHub accounts.
   b. Push their individual contributions to their personal fork
   c. Submit pull requests to merge their changes into the main project repository.

2. Groups should create a README on the main project page, giving their project name and a short description (e.g., abstract from the written report).
3. Upload all analysis code as Jupyter notebooks, separating distinct steps into separate notebooks (e.g., upstream quality control vs. different downstream analyses). Different team members should take responsibility for different notebooks, and others should submit revisions as pull requests to make necessary revisions (e.g., correct spelling, add comments, etc). All notebooks should be fully executable; raw data can be retrieved via URL requests, and small data files can be stored in your repository to make downstream analyses executable (note: GitHub has a ~1 GB limit per repository).
4. Establish a clear internal structure (e.g., separate data from analysis notebooks and any code modules in separate directories).
5. Give clear filenames to all files, and use READMEs as appropriate to clarify what these files are and if there is any logical progression to them. Remove any unnecessary files, e.g., intermediate outputs or preliminary notebooks that are not relevant for the final report.

The grade will be assigned to each individual based on (1) clarity of code (e.g., code comments, style), (2) appropriate use of methodology, (3) executability of code, and (4) equal contributions of all members (note: GitHub tracks individual contributions so we will use this partially as the basis of judging group equality! And peer grading as a second factor).

Part 2: Written report

This should be written in the form of a research article and **submitted in PDF format**. Hence, it should be professional, concise, and clear. The report itself will have a specific structure:

1. Abstract (≤ 250 words) describing the problem, methodology, results, and conclusion in 1-2 sentences each. Should clearly present problem and findings
2. Introduction (~500-1000 words). Briefly describe the problem, and provide background on the biological context (e.g., describe why this problem is important and cite primary literature that has examined similar problems).
3. Methods (~750-1500 words). Concisely describe the methodology used to address the problem, and the rationale for this approach. This should cite the primary literature appropriately. Consider using a figure (e.g., flowchart) to give an overview of your methodology.
4. Results (~1500-2000 words). Concisely describe your findings, interpretation, and solution. Use figures and tables to show your findings, and avoid overly verbose descriptions of these findings — instead, distill these findings to demonstrate a clear understanding of the results and their interpretation.
5. Discussion (~1000-2000 words). Do not repeat the results. Instead, connect your findings and interpretation with the larger problem. Discuss limitations of your approach. Discuss future steps that could be taken to better address the problem. Describe your overall conclusion. Consider using another figure, if appropriate, to convey your conclusion.
6. References. Cite all primary literature, methods, etc, appropriately!

The grade will based on:

1. Abstract + Introduction: Clear understanding of the problem and discussion of background (10%)
2. Methods + Results: Appropriate application of methodology and presentation of methods (20%)
3. Results + Discussion: Clear presentation of results and interpretation, demonstrating understanding of methods, approaches, *and their limitations* (50%)
4. Data visualization: Quality figures (10%)

5. Overall quality: This class is not a scientific writing class, but this is a skill that you should have cultivated by now in your studies, and hence writing quality and style will be partially graded (10%).

All students are expected to contribute equally to the writing. Anonymous peer grading will be collected and used to award bonus points (i.e., bonus points will be given to students who contribute equally; those perceived by peers to have under-contributed will receive no bonus points).

Specific Tasks:

Each group will receive an individual project assignment, with some specific questions that are unique to that group's dataset. In addition, all groups are expected to utilize all or most of the methods learned in this course (and maybe some others as well!), except when it is not possible or appropriate for their data (ask an instructor if you think this is the case). Below are some specific tasks that should be performed or attempted by all groups. Each of these tasks should be accompanied by one or more visualizations or tables (as appropriate):

1. **Sequence quality control.** Appropriate quality control and filtering procedures should be applied. Appropriate denoising and/or clustering techniques should be applied, except if a group is given explicitly pre-processed data.
2. **Taxonomy classification**. Appropriate taxonomy classification techniques should be applied and explained. Try different methods and databases — you do not need to include this in your report, but you could if the differences are meaningful (e.g., one database clearly gives better results).
3. **Alpha diversity** should be estimated for all samples, and compared across the primary sample categories* (e.g., sample types or other main groups) and/or gradients* (e.g., age, space, time, pH, or other continuous sample metadata). Test out a few different metrics (including a phylogenetic metric) to see what they reveal. Apply appropriate statistical tests.
4. **Beta diversity** should be measured and compared between the primary sample categories or gradients*. Test out a few different metrics (including a phylogenetic metric) to see what they reveal. Apply appropriate statistical tests.
5. **Which features are more/less abundant in different groups?** Apply appropriate statistical tests and/or supervised learning methods to answer this question. Consider carefully what types of features you want to compare between groups — maybe use different feature definitions (e.g., collapsed on taxonomy or functional information).
6. **[Optional] Functional prediction**. We would probably use complementary methods to assess microbial functions in a real research/clinical/industrial context (e.g., targeted methods, culture-based methods, metagenomics, etc). That is not an option here, but you can pretend that you did! Use q2-picrust2 to predict the metagenome composition of your samples and use this to compare your primary categories or gradients*. Repeat beta diversity analysis based on predicted metagenome composition. **The functional prediction with q2-picrust activity was optional, so this step is not required. But if you do not choose to use q2-picrust, you should "do something new" twice (see below).**
7. **Do something new**. Part of this course is learning how to read and use bioinformatics documentation! Pick a method that we did not learn about in class, and which will add value to your group project (especially if it relates to one of the specific objectives of your group project). Install it, use it, and present your results! This could be a QIIME 2 plugin or action that we did not use, or different statistical or visualization packages for Python or even R.

*The "primary categories or gradients" of interest in your project will be described in your assignment sheet, but don't stop there! Other exciting things could be found in the sample metadata provided.