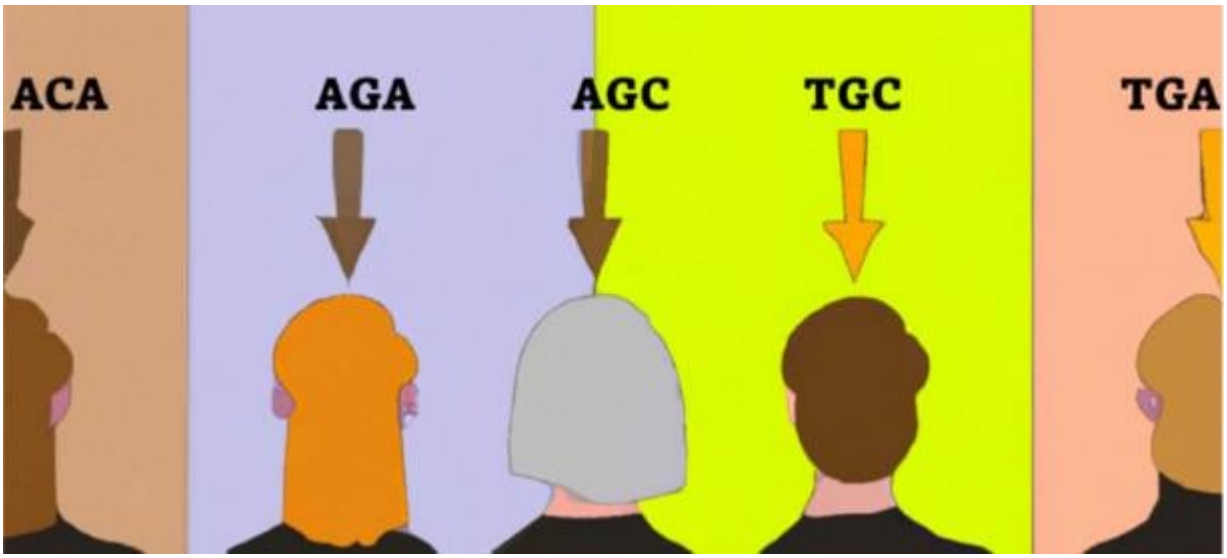
BINP29 *Population Genetic* Projects



Analyzing ancient DNA with home DNA tests, scientists can identify specific mutations and determine how your traits have been passed on to you over time. (Enkigen Genetics Limited)

Table of Contents

What is a bioinformatic project? 3
From an idea to a bioinformatic application 3
Proposed population genetic projects 3
Datasets 4
High complexity 4
Low complexity 6
Read more about mtDNA and Y tests 10
Product design 10
Oral presentation 10
Written report 10
Submission (format and deadline) 11
Project evaluations 11
Writing the paper 11
Other guidelines 12
Most common comments from past years: 13
FAQ 14

Instructions for BINP29 *Population Genetic* Projects

Welcome to the final part of BINP29 – the development of a Bioinformatics project in population genetics of your choosing!

The goals of BINP29 Projects are to teach you how to develop software by allowing you to sharpen your coding skills, get acquainted with population genetics, and expand your GitHub portfolio. You will also learn how to write an academic paper and present your work.

BINP29 projects include a **development project**, an **oral presentation**, and a **written paper**. Before we start bouncing off ideas for projects, let us first understand what constitutes a bioinformatics project.

What is a bioinformatic project?

You should develop software (e.g., web application, script, tool) and apply it to some population genetic dataset. You need to think of a problem in population genetics that your code solves and explain why it is an important problem. Consider the following examples:

Not a biological problem	**A biological problem**
Counting the number of GC% in the 8th position of the DNA sequence is not biologically informative because the 8th position of genomes has no biological function or meaning.	Comparing heterozygosity levels in different populations is informative to study inbreeding patterns.

If you decide to develop a new tool, you should include 1-3 examples (an example can contain multiple genomes) of how you apply your code to one or more datasets and explain why it is useful.
Your end goal is to build a useful tool for the science community.

With that in mind, let us now consider some ideas for bioinformatics projects.

From an idea to a bioinformatic application

You have different options to choose from in developing your bioinformatics application. Any programming language could be used. You can choose to develop one or more of the following:

- A software that connects to a database
- A web application in pure JavaScript
- A web application that allows querying files
- A web application that connects to a database

Below are suggestions for a Bioinformatics project. You may expand on these suggestions as you wish or, develop variants of these ideas, or combine them with other features, for example, a toolbox that is accessible through a web application (note, complex applications would receive a higher grade in the *Software* category).

All the projects must be pre-approved by the teacher. The work is independent. Two students CAN NOT select the same project.

Proposed population genetic projects

Note that the projects are ranked by complexity, one of the components of the software grade.

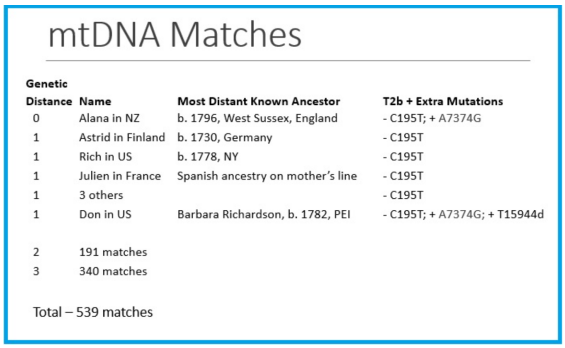
Datasets

- AADR_54.1 – The entire aDNA datasets in PLINK format
- AncientMtDNA – Ancient mtDNA data. mtdb_metadata.xlsx and mtdb_888-records_ekz82j1k.fasta, include the annotation and mtDNA sequences in FASTA format.
- AncientYDNA – Ancient Y data
- TestUsers – 5 files of test users. The the mtDNA appears as chromosome 0 or 25.
- VIPHaplogroups – mtDNA and Y haplogroups of VIPs
- IBD - IBD connections between ancient individuals.
- **ClinVAR** data are available in https://ftp.ncbi.nlm.nih.gov/pub/clinvar/tab_delimited/ (`variant_summary.txt.gz`)

High complexity

1. **MtDNA haplogroup matches app**. This app will report the genetic similarity between the user's mtDNA and ancient mtDNA data.
    a. Dataset. AncientMtDNA. Read the paper: *AmtDB: a database of ancient human mitochondrial genomes.*
    b. User data. Use TestUsers. Assume the user's haplogroup is known.
    c. Identify all positions relevant to the haplogroup using the file mt_phyloTree_b17_Mutation.txt. Analyze ONLY those positions from hereon.
    d. Compare the user's file with the FASTA file (you can assume that the position indexes are aligned).
    e. Rank all the reference individuals by their highest similarity to the user and report the genetic distance (0=perfect similarity).
    f. Develop an interface that allows selecting a reference individual from the FASTA file.
    g. If a certain reference individual is selected, output in a table shows which of the user's mutations are similar or different to the reference individual chosen. Report all other metadata for the reference individual.
    h. Print a table with all the mutations that are similar/different between the user and the reference individual. If both are missing a mutation (of the haplogroup) report it as - <mutation>; otherwise, report it as a + <mutation>

    For example:



mtDNA Matches

Genetic Distance	Name	Most Distant Known Ancestor	T2b + Extra Mutations
0	Alana in NZ	b. 1796, West Sussex, England	- C195T; + A7374G
1	Astrid in Finland	b. 1730, Germany	- C195T
1	Rich in US	b. 1778, NY	- C195T
1	Julien in France	Spanish ancestry on mother's line	- C195T
1	3 others		- C195T
1	Don in US	Barbara Richardson, b. 1782, PEI	- C195T; + A7374G; + T15944d
2	191 matches		
3	340 matches		

Total – 539 matches

https://leannecoopergenealogy.ca/2018/01/19/using-mtdna-for-genealogy-a-case-study-comparing-results/

2. **Y haplogroup matches app**. This app will report the genetic similarity between the user's mtDNA and ancient mtDNA data.
    a. Dataset. The AADR files include the Y chromosomes marked as chromosome 24 and are available in PLINK format. Annotation is available in the Excel file.
    b. User data. Assume that the user's file is available in plink format, where the Y data appears as chromosome 24. Assume the user's haplogroup is known.
    c. Identify all positions relevant to the haplogroup using the file snpFile_b38_isogg2019.txt. Analyze ONLY those positions from hereon.
    d. Compare the user's file with the AADR file (assume that both are in b38).
    e. Continue as per above.

3. **Plot haplogroup frequencies on a table and map (Y & mtDNA).**
    a. Dataset: AADR.
    b. Print a table that shows the basal haplogroup (single letter) frequencies for ancient populations.

Ancient pop name	Country	Age	Lat	Long	A	B	C	….	Total
Germans 1000 BP					10%				1
Germans 1500 BP					5%				
Germans 2000 BP					3%				
Germans 2500 BP									

c. Show the populations as points on the map. Clicking on a point would show a pie chart with the frequencies for this population.

4. **Visualize ancient IBD connections using circus plots**
    a. Dataset. IBD.
    b. Provided a dataset of identity by descent (IBD) distances among ancient individuals (individual-individual). Write an interface for this dataset.
    c. Draw circus plots for a selected individual/population.

5. **Visualize ancient IBD connections on a map**
    a. Dataset. IBD.
    b. Provided a dataset of identity by descent (IBD) distances among ancient individuals (individual-individual). Write an interface for this dataset.
    c. Develop a graphic interface that draws IBD patterns on a map for a selected individual/ population.

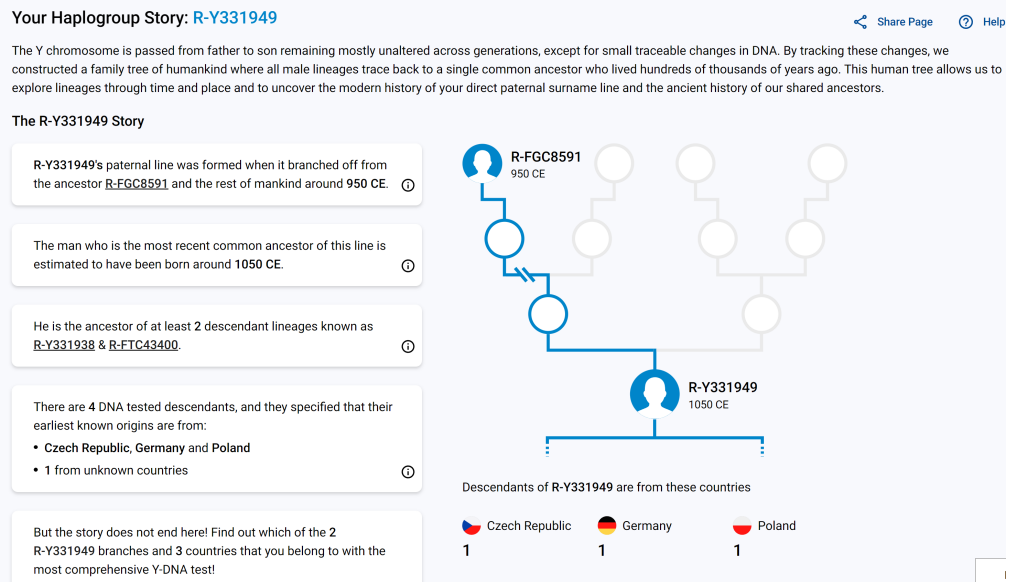
6. **Calculate ancient IBD between a modern individual and ancient ones**
    a. Dataset. IBD, TestUsers
    b. Implement the ancIBD pipeline mentioned in https://www.nature.com/articles/s41588-023-01582-w#Sec25 and available in https://github.com/hringbauer/ancIBD for a modern individual. Your application should take a Test individual file and calculate the IBD against all ancient populations in the IBD folder.
    c. Report the results in a table (sort the results by the highest total share IBD). Report it by genomic region (chr, start, stop)

7. **Violations of genetic-geographic distances**
    a. **Dataset.** AADR_54.1
    b. Calculate the IBS distances between all the samples using PLINK https://zzz.bwh.harvard.edu/plink/strat.shtml#matrix.
    c. Divide the world into hexagons (decide on a reasonable size).
    d. Divide the aDNA data into time bins.
    e. For every time bin, calculate the average IBS distance between all the neighboring hexagons and report the six distances. Population genetic theory suggests that geographic and genetic distances are correlated. Report any anomalies.
    f. Do the same for the next time bin.

Low complexity

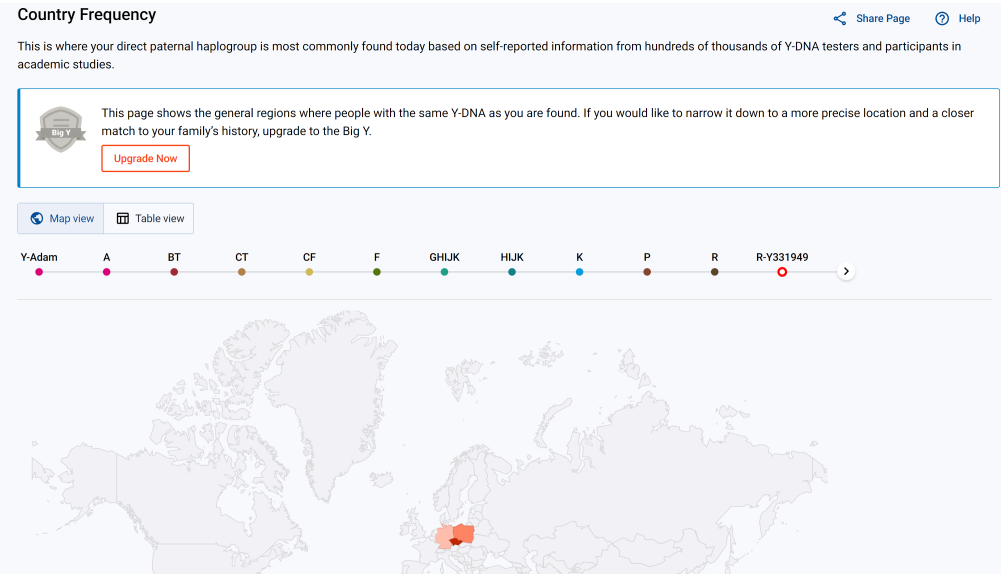
8. **The haplogroup storyteller (Y or mtDNA)**. Assume that the user's haplogroup is given. Write an app and interface that provides the following report.

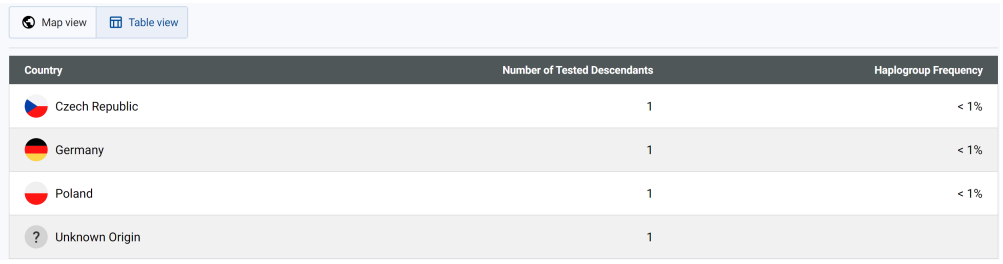
    a. Use AADR's data to infer the most ancient emerges of this haplogroup and its countries of origin.



9. **Haplogroup frequency per country (Y or mtDNA)**. Assume that the user's haplogroup is given. Write an app and interface that provides the following report and heatmap.

    a. Use AADR's data to infer the most ancient emerges of this haplogroup and its countries of origin.



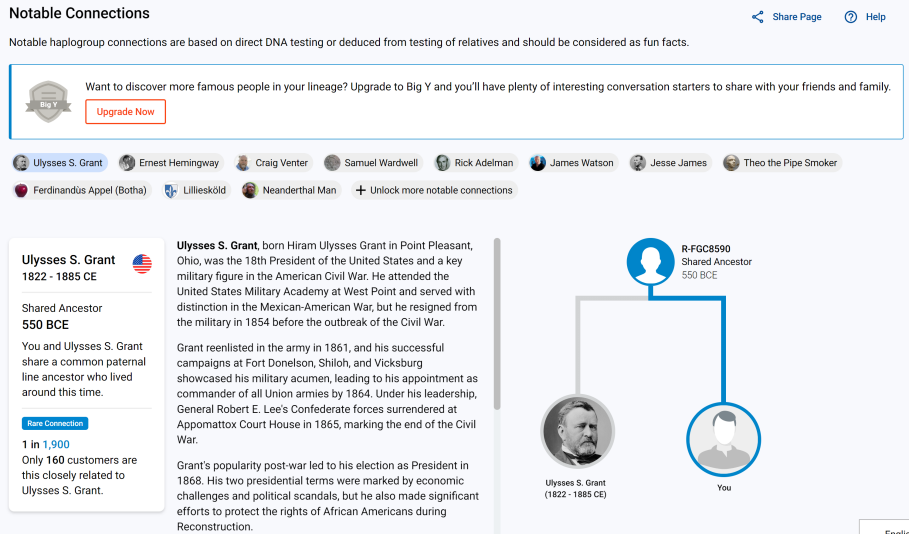


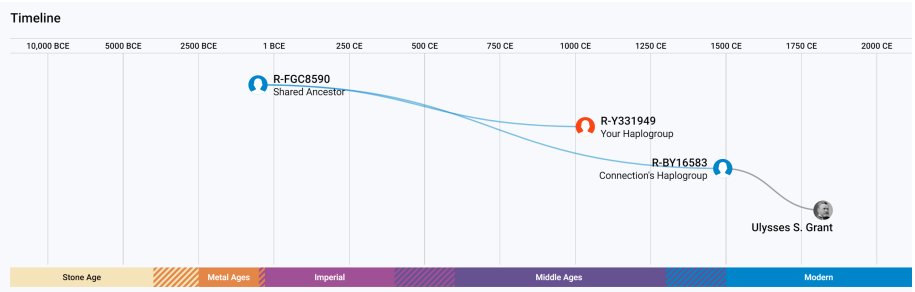
10. **Haplogroup of VIPs (Y or mtDNA)**. Assume that the user's haplogroup is given. Write an app and interface that provides the following report.

    a. Use AADR's data to infer the most ancient emerges of this haplogroup and its countries of origin.

    b. VIP haplogroups are in the designated folder.

    c. **Bonus**. Can you think of a way to generate text quickly?



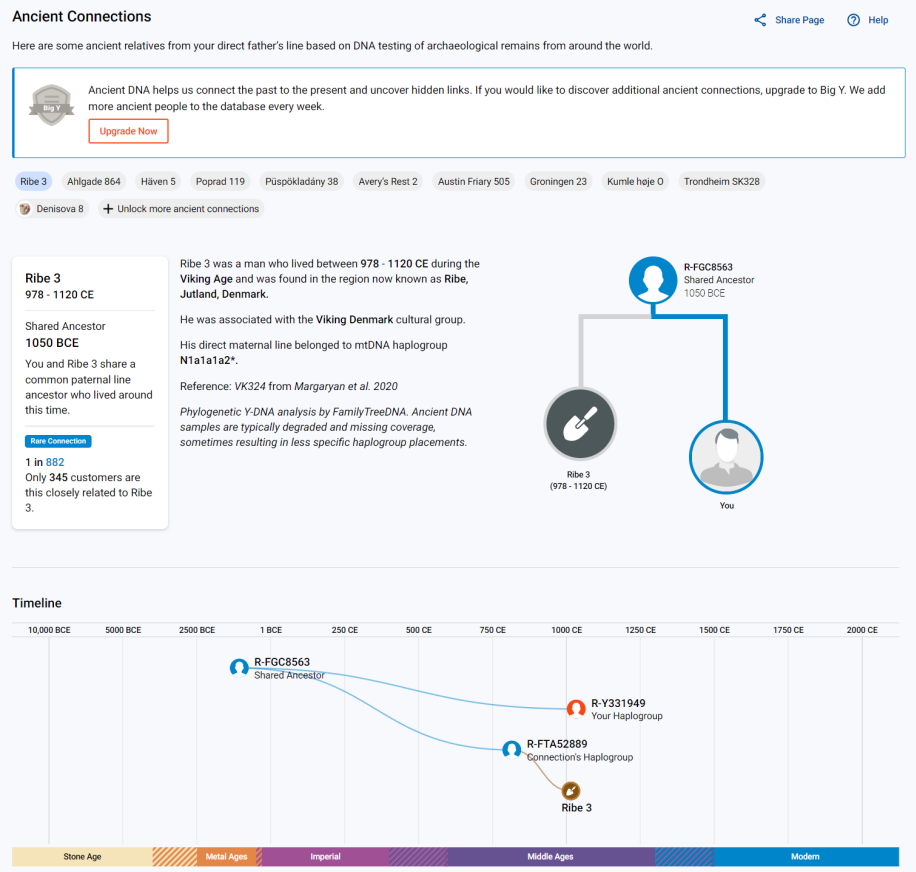


11. **Ancient Connections (Y or mtDNA)**. Write an app that allows selecting 2 people (e.g., 2 test users/2 AADR samples).

    a. Calculate their most common ancestral haplogroup (If its R1a1 and R1a2 then the common ancestor is R1a).

    b. Find their most ancient common ancestors in the AADR database and plot them on the timeline.

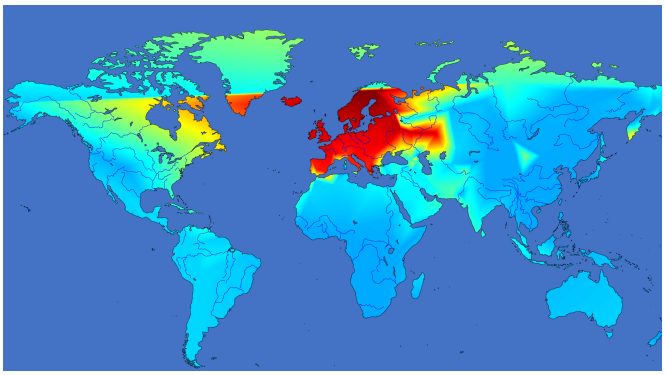
    c. Connect all the samples along the timeline.



12. **Ancient paths (Y or mtDNA)**. Check out the screen https://discover.familytreedna.com/y-dna/R-Y331938/path. Write an app that allows selecting a haplogroup and calculate the paths (e.g., if you chose R1a, then show R1a -> R1 -> R -> HIJK -> F -> FT -> C -> DE -> CF-> CT-> B -> A -> A00). Then, it pulls information from the AADR to calculate a table like this screen. Since you don't know the number of users, estimate the number of people in the world who carry each haplogroup based on the frequencies per country in the AADR and your knowledge of the population size of each country.

13. **Disease susceptibility.**
    a. Dataset. ClinVAR markers (markers associated with diseases) https://www.ncbi.nlm.nih.gov/clinvar/.
    b. Identify which ancient people have CliVar marker and output that to a table.
    c. Reads the user's file and outputs a table of the ClinVAR markers the user has.
    d. Output to a table which ancient people have the same ClinVAR mutations as the user.
14. **Animation of Heat Maps**
    a. **Dataset.** ModernSamples
    b. The Excel file GeneticDistances.xlsx contains the distances of modern populations (Year before the presence [BP]=0) to Iceland Vikings (code 1000). Create a geographic heat

map using this data, which should look like this:



c. Extend the table using Ancient DNA records to the table based on current knowledge of the Vikings (accuracy is not super important). Divide the periods into 100-year bins. Your code should now generate multiple heat maps, one per time bin.
d. Create an animation from all these images and save it as a video file.

You are welcome to come up with ideas outside of this list, as long as they are pre-approved by the teacher, to ensure that their scope fits the timeline and expected complexity of your project. You can use the databases of modern (PLINK files for Projects.zip) and ancient populations.

Read more about mtDNA and Y tests

https://help.familytreedna.com/hc/en-us/categories/1500001398742-Types-of-DNA-Tests

Product design

- Your code should work and be well-documented.
- Ensure a smooth and presentable design of your product.
- When you are done developing your software, upload it to GitHub.
- Implement your knowledge of version control, documentation, and proper instructions.
- Provide Readme files with instructions on how to execute your code on a dataset or example file that you will also include, allowing anyone to REPLICATE your work.

Oral presentation

You will be asked to present your work to the class. You will have 3-4 minutes to present + 2 minutes for questions. Your presentation and claims would be evaluated by a referee (the teacher).
You don't need to actually run your tool (unless you must, in the cases of web applications). Don't show your code. Instead, show screens of your tool working and convince us that you built a valuable and useful tool. The order of presentations will be by surname. If you do not present, you will get a 0 in that component.

Written report

Use a template. A 2-4 page report should be written using the Bioinformatics templates for writing papers (MS Word Template Bioinformatics.dotx). The template is found in the **Projects directory** on the course website.

How to write: After you review the template, read this https://www.scribbr.com/dissertation/discussion/ to understand how to write each part. Two additional documents with great writing tips available to you (Writing tips - How to write a great science paper.pdf, Writing tips - Ten simple rules for structuring papers.pdf).

Examples are available. A few application notes are provided (Application notes.zip).

Other examples. You are advised to find application notes or papers similar to what you will be writing to get an idea of the academic style that you will use (of course, not all the papers are perfect!).

Expectations. We expect you to submit excellent software products that work well and are usable. We expect you to deposit your code to GitHub. Finally, we expect you to submit well-written papers. Remember, this is a preparation for your Research Project, which prepares you for your thesis.

Submission (format and deadline)

- Submit your paper in Word format (firstname_lastname.docx). The paper should contain a link to GitHub.
- Submit your presentation in PowerPoint format (firstname_lastname.pptx)
- Submit your project files zipped in one file (firstname_lastname.zip)

- The deadline for the projects and papers is the 24th of March at noon.
- Late submissions would be penalized.

Project evaluations

The following will be evaluated:

Product design (10%)	Database/web design (if any) Clarity of instructions and GitHub presentation. Code documentation Organization of the GitHub folder. Existence of input/output files and folders
Software (40%)	Software replicability Accuracy Level of complexity
Oral presentation (20%)	Clarity and understanding of the project.
Written report (30%)	Publication quality

Writing the paper

Follow bioinformatics guidelines for authors on how to format your paper.

Title Section
Only include one author, you. You are also the corresponding author. Your affiliation is *Department of Biology, Box 118, 221 00, Lund University, Sweden*.

Abstract
Follow Bioinformatics guidelines for writing an abstract.

Introduction
Describe briefly what you have done. Motivate why this is good or interesting. Compare your work with other studies (How is tool work better? In what way does it improve the existing tools?).

Methods
Describe what programming languages you used including version number. What data did you use? Where was it obtained or how was it generated? Give references. Include version number in parentheses.

Features
Describe what the application does. What input is needed? What output does the user get? Pros and cons of having a web application instead of a standalone program (if you did a web application).

Discussion
Briefly describe what you have done and give your application's advantages. What do you foresee when it comes to the user base? What are the limitations and known bugs of the tool? What could be the future directions of developing this technology?

Acknowledgment
We are grateful to the editor and two anonymous reviewers for their valuable comments on the manuscript and for Lund university for their financial support.

References
Include 1-5 references.
Follow the guidelines of Bioinformatics for writing references.

Are you still confused as to what to put in each section? Check out this webpage.
It has links to more resources.

Other guidelines
There are various papers with useful writing tips in the Project folder. You are advised to read them.
Writing a paper always takes longer than you realize, so do not leave it to the last minute.

Most common comments from past years:

- There is no discussion of prior art (other published methods that do something similar to what your tool does) in the introduction and discussion (you need to have it in both).
- You need to provide example input/output files (in specific folders) and provide instructions on how to get from the input to the output in your readme file will refer to.
- Your English and writing style requires much improvement. Your writing style needs to be more academic.
- You didn't write figure/table legends.
- You were asked to write your article using the word template.
- You should put more effort into showing the value of your work to the reader and make a stronger case for why they should use your program. Right now, it doesn't look like that.
- The paper is out of focus. You focus on the application of the method rather than the method. You should have expanded on why it is useful, again, in light of prior art.
- Your references are not in the right format. I strongly recommend using an annotation tool like EndNotes (license required) or Mendeley (free) and then checking it manually.
- You are confusing the results and discussion sections in your paper. In the **Results** section, you report the results. In the **Discussion** section, you need to discuss them, not rephrase the results. Read more here: https://www.scribbr.com/dissertation/discussion/.

FAQ

- Which programming language\technology should I choose?
    - It is up to you! You should pick something that you are familiar with or interested in learning how to use. Please recall that not everyone is familiar with all the languages. Therefore, the sooner you will inform us of your choice, the better it is so that we can assign someone familiar with this language\technology to assist you.
- I need help using this function…
    - Did you google it? One of the goals of this course is to help you learn to find the answers on your own through online searching. Therefore, approach the teacher only after you have exhausted these options since our goal is to prepare you for real life.
- What are my outputs? What should I send you?
    - Read the **Submission** subject above.
- Where can I find relevant data?
    - We made several datasets available to you.
    - There are several ways that papers report data, which also depend on the kind of data.
    - Some deposit their data in NCBI (mainly sequence data and genomes).
    - A variety of biological data can be found in https://datadryad.org/stash.
    - Some post their data on GitHub or their personal websites.
- Do I have to publish my code on GitHub?
    - Yes!
- Is it OK if I write bad code because I don't care about the grade in this class, and I think there is a fair chance that you guys won't realize how bad my code is?
    - You are missing the point of this exercise. This exercise is for YOU. The code you deposit on GitHub will become a part of your portfolio (remember, the more tools you have on your GitHub, the better you look!). Your code is a reflection of your skills. It tells your future employer that you are a serious person with solid coding and organizational skills. If you do a poor job, you will just have a harder time finding one.
- Do I have to use the Bioinformatics template and follow all their other guidelines?
    - Yes!
- Should I use Python or R Shiny for the interface?
    - It's up to you. This document may help you decide: https://towardsdatascience.com/hello-covid-world-python-dash-and-r-shiny-comparison-cc97afef9d82
- What should I put in GitHub?
    - Your GitHub depository should include the following:
        - A readme file that explains about your project, methodology, files included, how to run your code, and unknown bugs. You should explain how to install dependencies and answer common questions.
        - Your code should be well documented.
        - If there are multiple versions, show it and explain the difference between the versions.
        - Provide several (typically, 1-3) input and output files that demonstrate that your code works in separate folders.
        - Don't put all the files in one folder, be organized.
        - Assume that the user who wants to use your code is ignorant.

- You can also have a folder with previous versions. This demonstrates that you did version control.

- Should I do “live demonstrations” during my presentation or just show slides?
  - It’s up to you how to allocate the time for your talk. Screens capture of your tool is acceptable.
- Can I use an existing toolbox/package for my project?
  - No. You have to build something of your own. You can use existing tools if you need their functionality, but you would be evaluated based on what you built. In other words, **your tool should be publishable**, and only new tools that promote the field are considered publishable. Learning how to use existing tools does not satisfy this requirement.
- Should I use a citation manager?
  - **It is strongly recommended**. You are far more likely to make mistakes if you do not use a citation manager. If you want to stay in Academia, you have to learn how to use one.
- Which citation manager should I use?
  - There are many citation managers like EndNotes (license required), Mendeley (free), or other ones. They are compared here: https://en.wikipedia.org/wiki/Comparison_of_reference_management_software. Most of the students like Mendeley or RefWorks.
- I don’t know how to write academic papers.
  - That’s OK. If you never wrote an academic paper, there is no reason that you should know how to write one. Writing is something that we learn by doing and through examples. You were provided with examples of papers. Many more examples are available online. In addition, you were provided documents with helpful writing tips; use them. Finally, when you are done, proofread your paper. Word can catch many typos. Grammarly is also useful, even if you use the free version. Finally, exchange your draft with your friend and proof each other papers. This is how we write papers.
  - Read "The elements of style" and "Eats shoots and leaves" to improve your grammar and writing.
- Can I cite Wikipedia? What types of references are appropriate?
  - No. Read this website and this one concerning citation format and rules.
- Can I copy code from other tools?
  - No. That is considered plagiarism. You can get ideas from other codes. If you reuse parts of other codes, you have to acknowledge that in your paper, and it cannot be the main function of your code (otherwise, what is the value of your code?).
- Should I include example data in my paper?
  - Yes! Show the output of your tool when applied to real biological data in a graph or table. Remember that it is your job to convince others to use your software. So you must provide evidence that your tool works.
- Should I provide figures and table legends?
  - Yes! This is where you explain what the figure/table shows and says. Just writing the title is not enough.
- How can I get a good grade on this assignment?

    - Read this document. Prepare a checklist and go over it before you submit it. Show your paper to your peers and proofread each other papers. If something is unclear, ask the teacher.

- What happens if I do not submit the paper/code on time?
    - Late submissions will be penalized.
    - You will get an incomplete grade in the course until you get a passing grade on this assignment. Such a thing happens yearly (due to illnesses, etc.) and should not be a reason to stress.
- Is it OK to use AI for coding, writing, or figure generation?
    - No. You are still learning how to do it yourself. You can use AI to understand concepts (e.g., what is a power law distribution), but we still recommend using reliable sites first and AI only as a follow-up.
    - This just happened https://www.vice.com/en/article/4a389b/ai-midjourney-rat-penis-study-retracted-frontiers

Good luck!