Hanze
University of Applied Sciences
Groningen

Institute for Life Science & Technology

Data Science for Life Sciences
Student Manual
2023-2024

Authors:

Dr. M.E.F. Apol
J. Bosman, PhD
Drs B. Barnard
F. Feenstra, MSc.
Dr. J. Hageman
M. Herber, PhD candidate
M. Kempenaar, BAS
Dr. ir. D.R.M. Langers
Drs. M.F. Lurvink
Dr. T.A. Wassenaar
R. Wedema, BAS

Table of Contents

1. Program outcomes

The master Data Science for Life Sciences has 6 program outcomes which you should be able to do after you have finished your graduation. In all modules you will be working on these program outcomes, in the next chapters it is explained exactly how you will be working on them.

Conduct critical and creative research (CR)
The graduate demonstrates advanced expertise in formulating testable hypotheses relevant to clients' questions. The graduate possesses a comprehensive understanding of existing methods, theories and solutions to similar problems and can critically evaluate their applicability in diverse contexts. The graduate adeptly selects appropriate data research methods, providing sound justifications for their choices, or creatively adapts existing methods to develop original solutions for complex problems. Upon implementation, the graduate demonstrates a rigorous evaluation of the obtained solution, adhering to the available technical and engineering best practices in the field. The graduate iteratively refines and optimize the solution to achieve the most optimal outcome. Moreover, the graduate can transfer and generalize these methods, effectively applying them to neighbouring fields and related problems in new and unfamiliar environments.

Model meaningful information (MM)
The graduate demonstrates a high level of competence and expertise, applying a wide range of mathematical, statistical, and machine learning techniques to effectively identify complex patterns, causal relationships, and actionable insights, as well as making accurate predictions. The graduate demonstrates the ability to integrate diverse knowledge domains, to effectively handle complexity, and to extract meaningful information from data, even in the presence of incomplete or challenging datasets.

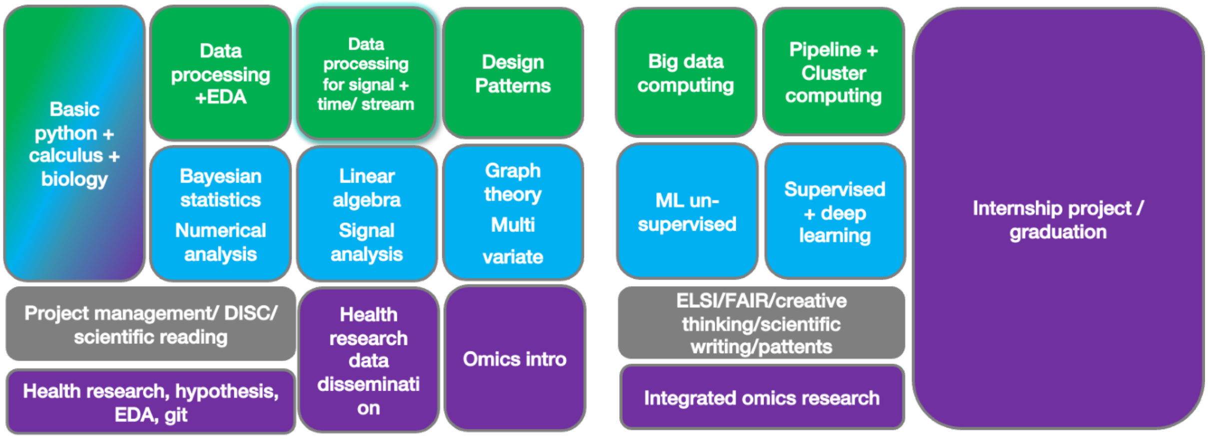
Deliver organized solutions (DO)
The graduate retrieves multilevel data from multiple sources and can organize, combine, clean, process and store those reliably, adhering to the FAIR principles (Findable, Accessible, Interoperable and Re-usable). Developed code is organized, well written, well documented, traceable via version control management systems, and suitably licensed.

Communicate effectively
The graduate communicates actively and effectively about his/her work, employing diverse modes of communication, including written, oral, and visual forms, with experts, peers, and individuals from non-specialist backgrounds. Specifically, the graduate adeptly formulates the research question or problem, provides comprehensive explanations and justifications for the chosen methods or approaches, and presents the results clearly, accompanied by a critical and reflective interpretation.

Being responsible (BR)
The graduate demonstrates awareness of ethical and legal considerations relevant to their work, taking responsibility for adhering to applicable laws and best practices. This includes a keen understanding of privacy issues, integrity, and security. Moreover, the graduate recognizes their professional responsibility within society and, whenever feasible, upholds the principles of FAIR (Findable, Accessible, Interoperable, and Reusable) for scientific data management and stewardship.

Being Entrepreneurial (BE)
The graduate demonstrates awareness of the broader and/or commercial applications of research outcomes, emphasizing a focus on practical implementation. The graduate demonstrates the ability to formulate viable business ideas and effectively engage stakeholders. As a self-directed and autonomous professional, the graduate assumes responsibility and takes proactive action when confronted with challenges.

2. Program overview

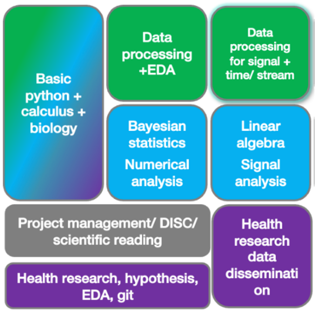


The master program is divided in four semesters. The first three semesters' courses are provided to lecture data science and programming subjects. Modules to enhance research and professional skills are provided as well. Research project needs to be conducted to apply theory and skills. The first three semesters research has to be carried out in small groups, the fourth semester is the graduation semester in which you have to carry out the research independently. You will graduate if you pass all the course modules exams, successful deliver the research projects and the proof of competence in which you proof the program outcomes and finally pass the graduation thesis, the graduation practical work and the graduation presentation (defense).

3. Program content

This chapter describes the theme of research project, the module overview shows the relation between the modules and the program outcomes and the assessment plans. Details per module such as descriptions about content, week overview, learning outcomes, literature, hours to spent and assessment method can be found in the module manual (chapter 4).

Research project 1 Data Science for personal health



The objective of this research project is to enhance students' comprehension of the Data Science domain, particularly in relation to personal health. To achieve this, health-related data will be visualized to answer health-related research questions. The students will apply fundamental data transformation and munging techniques to pre-process the data for visualization. Additionally, the project aims to reinforce the programming concepts taught in the preparatory course while teaching students the principles of good UI design. The project follows a cumulative design approach, wherein a basic prototype with a simple data structure is developed first, followed by more advanced iterations on the same theme. This allows students to build upon their existing knowledge and progressively enhance their skills.

Modules overview

Code	Name	ECTs	T*	Short description
BFVM23PREPPROGR1	Preparatory course programming	5	A	Basic Python
BFVM23PREPDATSC1	Preparatory course data science	5	W	Basics calculus
BFVM23PREPOMICS	Preparatory course omics	5	C	Biology
BFVM23PROG2	Programming II	5	A	Data processing + EDA
BFVM23DATASC2	Data science II	5	C	Bayesian statistics + numerical analysis
BFVM23PROG3	Programming III	5	A	Data processing for time series + signals
BFVM23DATASC3	Data science III	5	C	Linear Algebra + Signal Analysis
BFVM23RSRPFS	Research and Professional skills	5	D	Scientific method and collaboration
BFVM23DSPH	Research Project	10	P	

*: T is assessment **t**ype. D = digital portfolio; W = written Exam; C = computer exam; P = professional product; A = Assignment;

Relation between modules

The "Data science for personal health Project" will form the core module that allows the various skills that are learned in the other modules to be integrated and applied in a practical context. It will consist of a comparatively large project that runs throughout the semester. You will be asked to define your own health related research question and combine these questions into one design per group. Initially, there are three modules of the preparatory course which are intended for students without a sufficiently sound or broad background in life sciences. These serve to provide you with the required basics related to data analysis, programming, and biology that are necessary to successfully commence with the next modules and are considered optional if the student proves to already have the required entry level by either an exemption or passing the entry test. Subsequently, in the "Data Science II " module, you will learn how to analyze and assess data streams or data sets, either in exploratory or confirmatory fashion. This skill can be directly applied in the project, for instance to analyze time series data that are acquired by body-worn sensors that measure individual biological signals to obtain relevant summary statistics and to subsequently visualize the results to the user. This will require you to design and implement a specific analysis pipeline (e.g., using Python) and make the outcomes accessible by means of a (web) visualization (e.g., using the Python Bokeh library or the JavaScript programming language). These data analysis and visualization techniques will be covered in the module "Programming II and III". Finally, the module on "Research and Professional skills" will largely develop more general skills for which the data science for personal health projects serve as applications: for example, you may write a data management plan that pertains to your project or reflect on ethical issues surrounding the collection and processing of personal data.
For module description see chapter 4: module manual.

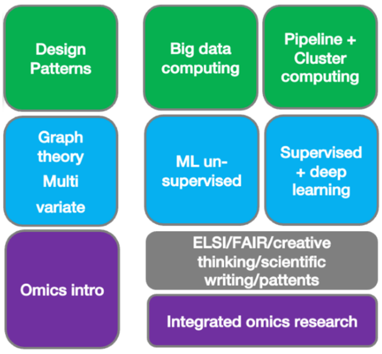
Relation to program outcomes

Program outcome	Assessment method
Conduct critical and creative research	Part of the assessment for **programming** assignments involves evaluating the use of an argumentative approach in justifying design decisions. In the **data science** II and III exams, students are evaluated on their ability to employ an argumentative approach when defending method selection and engaging in critical reflection of results within the context of the study case. The **research project** is assessed for critical and creative research competence through contributions in discussions, a paper, a code base, and a presentation. All should demonstrate the application of the scientific method. As part of self-assessment in **research and professional skills**, students will evaluate their competence using a competence card.
Model meaningful information	The evaluation of the ability to integrate diverse knowledge domains, navigate complexity effectively, and extract meaningful information from challenging datasets is conducted in the **Data Science** Exams and **Programming** assignments through the analysis of study cases. This competency is further assessed in the **research project**, where students are required to develop a website featuring comprehensive analysis and visualizations.
Deliver organised solutions	The code repository of the **programming** assignments as well as the **research project** is evaluated based on criteria such as the organization of code, documentation completeness, the comprehensiveness of the README file, and appropriate licensing.
Communicate effectively	Competence in this area is evaluated through the student's presentation of the research question or problem, the provided explanations and justifications for the chosen methods or approaches, and the clarity of presented results. The assessment includes code documentation in **programming** assignments, provided documentation in **data science** exams, sprint presentations within the **research project**, and the research paper associated with the research project. Furthermore, students are expected to conduct a self-assessment of their competence using the **research and professional skills** competence card.
Being responsible	Students are assessed for handling ethical issues with data during the **research project** and the **programming** assignments. The delivered solutions should be according to the FAIR principles. Furthermore, the student will conduct a self-assessment of the competence in the competence card (**research and professional skills**)
Being entrepreneurial	The assessment includes evaluating the student's ability to collaborate with others and their behavior as a self-directed and autonomous professional who proactively takes action when faced with challenges in the **research project**. This evaluation is done based on the taskboard participation and collaboration contract.

Assessment plan

Module	Quarter.Week	F/S	Method	Grading
Professional and research skills	Week 1.1	Formative	Disc Assessment	
Professional and research skills	Week 1.2	Formative	Competence card self-assessment	
Preparatory course programming	1.1..1.5	Formative	Week Assignments	
Research project	1.4	Formative	End of sprint	
Preparatory course data science (calculus)	1.9	Summative	Exam	100%
Preparatory course omics	1.9	Summative	Exam	100%
Preparatory course programming	1.10	Summative	Final Assignment	100%
	1.11	Summative	Interview	
Programming II	2.1..2.5	Formative	Week Assignments	
Research project	2.1	Formative	End of sprint	
Research project	2.4	Formative	End of sprint	
Research project	2.7	Formative	End of sprint	
Data science II Bayesian statistics	2.8 or 2.9	Summative	Exam	50%
Data science II Numerical analysis	2.8 or 2.9	Summative	Exam	50%
Programming II	2.10	Summative	Final Assignment	100%
Professional and research skills	2.10	Formative	Competence card self-assessment	
Programming III	3.1..3.5	Formative	Week Assignments	
Research project	3.1	Formative	End of sprint	
Research project	3.6	Formative	End of sprint	
Research project	3.8	Summative	Code repository	30%
Research project	3.8	Summative	Presentation	25%
Research project	3.8	Summative	Article	25%
Research project	3.8	Summative	Conduct	20%
Data science III Linear Algebra	3.9	Summative	Exam	50%
Data science III Signal Analysis	3.9	Summative	Exam	50%
Programming III	3.10	Summative	Final Assignment	100%

Research project 2 Integrated omics



"Omics" techniques are both quantitative as well as high throughput, leading to large datasets of information amenable to analysis by advanced statistics and machine learning. First, you will be introduced to state-of-the-art lab techniques in the areas of (meta)genomics, transcript omics, metabolomics, proteomics, epigenomics, food omics, imaging, epidemiology etc. depending on project choice. You will choose a research project for which datasets of multiple types are available and formulate and test a hypothesis using appropriate quantitative methods (statistics/Machine Learning). A crucial aspect is communicating the methods and the findings to peers and clients (if applicable). Where appropriate, visualisation and web techniques from semester I are used to report and clarify the findings.

Modules overview

Code	Name	ECTs	T*	Short description
BFVM23PROG4	Programming IV	5	A	Elective OO / R
BFVM23DATASC4	Data Science IV	5	C A	Graph Theory + Multi variate analysis
BFVM23PROG5	Programming V	5	A	Big data computing
BFVM23DATASC5	Data Science V	5	A	Unsupervised machine learning
BFVM23PROG6	Programming VI	5	A	Cluster computing
BFVM23DATASC6	Data Science VI	5	A	Supervised + deep learning
BFVM23RSRPFS	Research and professional skills	5	D	Competences workshops
	Research Project	10	P	Integrated Omics

*: T is assessment **t**ype. D = digital portfolio; W = written Exam; C = computer exam; P = professional product; A = Assignment;

Relation between modules

The omics project will be centered on a research question for which several datasets from different 'omics' origins need to be integrated. Within the project in omics, the relevant biological background will be presented and discussed. Students can, depending on the need of the project deepen themselves in either R or OO in python. This The integration requires balancing within dataset and between dataset analysis of relations, for which Data Science IV provides a theoretical and practical framework (graph theory, multivariate analysis). is later complemented with machine learning and image analysis theory and methods (Data Science V+ VI). To handle and analyze the complex datasets involved, students will learn strategies for distributed computing and the implementation thereof in Programming V and VI. Against the background of large (public/private/medical) datasets, the student is made aware of FAIR data management and possible legal, societal and privacy implications (FAIR/ELSI workshops). The project will be shaped according to state-of-the-art research. To this end, the student is to present a project plan with overview of the scientific background and context in the first weeks for peers and at the end report the results as a model scientific publication with ample supplementary information.

For module description see chapter 4: module manual

Relation to program outcomes

Program outcome	Assessment method
Conduct critical and creative research	Part of the assessment for **programming** assignments involves evaluating the use of an argumentative approach in justifying design decisions. In the **data science** IV, V and VI assessments, students are evaluated on their ability to employ an argumentative approach when defending method selection and engaging in critical reflection of results within the context of the study case. The student needs to demonstrate understanding of existing methods, theories and solutions to similar problems and should critically evaluate their applicability in presented contexts. The **research project** is assessed for critical and creative research competence through contributions the paper, the code base, and the final presentation. All should demonstrate the application of the scientific method. The student will be assessed on the competence of creatively adapting existing methods to develop original solutions for complex problems. As part of self-assessment in **research and professional skills**, students will evaluate their competence using a competence card.
Model meaningful information	The evaluation of the ability to integrate diverse knowledge domains, navigate complexity effectively, and extract meaningful information from challenging datasets is conducted in the **data science** Exams and **programming** assignments through the analysis of study cases. This competency is further assessed in the **research project**, where students will be assessed on the result and discussion part in the article.
Deliver organised solutions	The code repository of the **programming** and **data science** assignments as well as the **research project** is evaluated based on criteria such as the organization of code, documentation completeness, the comprehensiveness of the README file, and appropriate licensing.
Communicate effectively	Competence in this area is evaluated through the student’s presentation of the research question or problem, the provided explanations and justifications for the chosen methods or approaches, and the clarity of presented results. The assessment includes code documentation in **programming** assignments, provided documentation in **data science** exams, sprint presentations within the **research project**, and the poster presentation and research paper associated with the research project. Furthermore, students are expected to conduct a self-assessment of their competence using the **research and professional skills** competence card.
Being responsible	Students are assessed for handling ethical issues with data during the **research project** and the **programming** assignments. The delivered solutions should be according to the FAIR principles. Furthermore, the student will conduct a self-assessment of the competence in the competence card (**research and professional skills**)
Being entrepreneurial	The assessment includes evaluating the student's ability to collaborate with others and their behavior as a self-directed and autonomous professional who proactively takes action when faced with challenges in the **research project**. The student will furthermore be assessed in the research paper on awareness of the broader and/or commercial applications of research outcomes, emphasizing a focus on practical implementation.

Assessment plan

Module	Quarter.Week	F/S	Method	Grading
Research project	4.7	Formative	End of sprint	
Professional and research skills	4.8	Formative	Competence card self-assessment	
Data Science IV Multi variate analysis	4.8	Summative	Assignment	50%
Data Science IV Graph Theory	4.9	Summative	Exam	50%
Programming IV	4.10	Summative	Portfolio	100%
Data Science V Unsupervised ML	1.1..1.4	Formative	Week Assignments	
Data Science V Unsupervised ML	1.8	Summative	Final Assignment	100%
Programming V	1.9	Summative	Portfolio	100%
Research project	1.10	Formative	Midterm presentation	
Programming VI	2.7	Summative	Final Assignment	100%
Data Science VI Supervised and Deep Learning	2.8	Summative	Final Assignment + interview	100%
Research project	2.10	Summative	Research plan	15%
Research project	2.10	Summative	Code repository	25%
Research project	2.10	Summative	Poster Presentation + defence	15%
Research project	2.10	Summative	Article	35%
Research project	2.10	Summative	Conduct	10%
Professional and research skills	2.10	Formative or Summative	Competence card self-assessment	100%*

*When students request for summative assessment

Graduation Semester

To graduate from the master Data Science for Life Science, you must also write a master thesis based on research that you have carried out in graduation project in the final semester. For all the regulations concerning the graduation project a graduation manual will be handed to the students. Furthermore, you need to finish the research and professional skills subject by mean of a Proof of competence if not fulfilled yet.

Modules overview

Code	Name	ECTs	T*	Short description
BFVM23GRAD	Graduation project and thesis	30	W/ P/T /F	final research project in which the student conducts independently a research project at master level
BFVM23RSRPFS**	Proof of competence	10	D/F	Final digital portfolio in which the student delivers documented evidence of competences described in the program outcomes

*: T is assessment **t**ype. D = digital portfolio; W= practical work; T = Thesis; F = Defense
** when not already fulfilled

BVFM23GRAD Relation to program outcomes

The student is required to demonstrate the program outcomes through the completion of the thesis, practical work, and the defense. See also the graduation assessment form in the graduation blackboard course for more details.

BVFM23RSRPFS Relation to program outcomes

The student is required to provide proof of competences of the programme outcomes by means of a digital portfolio and its defense. See also the competence card manual provided on blackboard for more details.

Assessment plan

Module	Time of assessment	formative / summative	Method	Grading
Graduation	Week 20	summative	Thesis	60%
	Week 20	summative	Defence	15%
	Week 20	summative	Practical work	25%
Research and professional skills*	Week 19	summative	Proof of competence	100%

*When not already fulfilled

4. Module manuals

In this chapter an overview is given off all the course modules and professional and research skills modules. At first a glossary is presented.

Glossary

Term	Description
Tutor groups	Small group of students that come together to discuss and explain study material to each other that is assigned weekly, and to complete accompanying exercises. The lecturer is available at the onset of meetings for coordination of the students' study progress and furthermore on request to provide additional explanation
Courses	A course is a module. There are theoretical / skills modules like the data science modules, programming modules. There are skills related modules like the professional skills and research skills modules and there are research project modules.
Preparatory course	Based on the decision of the admission committee a student has to conduct the preparatory course. The preparatory course is a module to prepare the students up to the required level for following modules.
Lectures	A lecture usually involves a member of the senior academic teaching staff presenting themes and concepts related to a course of study to students enrolled in that course. The lecturer presents information to a large class, and while questions are encouraged, there is minimal group discussion.
Computer labs	During the computer labs you work either individually or in a small group to learn and experiment with the course material in a hands-on environment.
Tutorial session	A combination of theoretical background presentation in the format of a lecture, application of those concepts by means of tutorials with encouragement of experiments to be conducted in the computer labs
Project meetings and sprint meetings.	Meetings in which the progress of the research project (DS for personal health project and the Integrated omics project) is discussed with the team and the tutor. During project meetings feedback is provided on personal development, project performance, research (research question, hypothesis, validity and results) and the validity / future use in the field. During sprint meetings external stakeholders might attend.
Workshop / Tutorial	Workshops or Tutorials usually involve a member of teaching staff presenting themes and concepts, or the development of a skill, related to the course of study. Workshops may involve more hands-on learning however also allow discussion, interaction, presentation and debate on a given topic.
Masterclass	Masterclasses are lectures about cutting-edge knowlegde direct from researchers active in the field. These classes can be organized by the teaching staff as well as the students themselves
Graduation project and thesis	The graduation project is the final research project in which the student conducts independently a research project at master level to be reported in a master thesis.
Project in omics	The English-language neologism **omics** informally refers to a field of study in biology ending in *-omics*, such as genomics, proteomics or metabolomics. The related suffix **-ome** is used to address the objects of study of such fields, such as the genome, proteome or metabolome respectively. Omics aims at the collective characterization and quantification of pools of biological molecules that translate into the structure, function, and dynamics of an organism or organisms. There are two research projects related to the omics field (excluded the graduation project). The q DS for personal health project and the integrated omics project.
Integrated omics	Integration of current "omics" techniques and data in order to answer research questions that cannot be answered using only one type of analysis

Data Science subjects

Module code	BFVM23PREPDATASC1
Module name	Preparatory course data science
Module designers	Bryan Williams, Dave Langers, Fenna Feenstra, Marion Dam
Contact	Fenna Feenstra
Grading Teachers	Fenna Feenstra
ECTS and grading	5
Learning outcomes	- You interpret mathematical notation. - You apply basic equations analytically, including linear, rational, quadratic, trigonometric, exp/log equations in one variable. - You execute differentiation and integration of standard functions in simple forms. - You understand basic matrices operations
Description	In this course, the student will revise basic mathematical skills and knowledge in the fields of calculus. This is one of the three optional modules of the Preparatory course. It's intended for students without a sound background in mathematics. The basic mathematical skills are a required level for the data sciences subjects
Teaching method	Twice a week, students meet in small tutor groups (2 hrs) to discuss and explain study material to each other that is assigned weekly, and to complete accompanying exercises. The lecturer is available at the onset of meetings for coordination of the student's study progress and furthermore on request to provide additional explanation. Students submit their completed assignments
Scheduled	Quarter1, Year 1
Assessment	Written Exam
Competences	Model meaningful information
Entrance requirements	A basic level of mathematics is required
Planning	Week 1: Assessment and Notations - Assessment (first lesson) - Introduction to mathematical notation and conventions. Week 2: Solving Equations and Functions - basic algebraic operations and rules, linear and quadratic equations, and systems of linear equations. - piecewise-defined functions, composite functions - polynomial functions Week 3: Special functions and derivatives - Exponential, logarithmic, trigonometric, and their properties. - Introduction to derivatives Week 4: More derivatives and anti-derivatives - Applications of derivatives such as optimization and local extrema. - Introduction integration & applications for integration Week 6: Matrices Arithmetic - Matrices and vectors as arithmetic or geometric objects. - Matrix/vector addition/subtraction and multiplication. Week 7: Exam training
Contact time	28 hr (7 weeks x 2 x 2-hr blocks/week) – theoretical lectures/demonstrations
Literature	Suggested: S. K. Chung. Understanding basic calculus Calculus Syllabus on Blackboard

External links/sources	https://www.khanacademy.org/math/algebra-basics https://www.khanacademy.org/math/differential-calculus/ https://www.khanacademy.org/math/integral-calculus https://www.khanacademy.org/math/algebra-home/alg-matrices
Language	English

Module code	BFVM23DATASCNC2
Module name	Data science 2
Module designers	Dave Langers, Emile Apol
Contact	Fenna Feenstra
Grading Teachers	Dave Langers, Emile Apol
ECTS and grading	5
Learning outcomes	- You assess the quality of life science data and perform data clean-up - You apply frequentist and Bayesian methods to estimate parameters with standard errors and confidence intervals - You implement and apply numerical methods for analysis of data, including differentiation, integration and finding roots and extrema - You explain how discretization, round-off and error propagation may affect the results of outcomes
Description	This course is designed to provide students with the practical implementation of Bayesian Statistics and Numerical Analysis using Python. The first subject, Bayesian Statistics, covers topics such as constructing histograms, calculating sample moments, estimating parameters using the Method of Moment and Maximum Likelihood methods, evaluating various distributions, using null hypothesis significance tests, assessing normality, calculating standard errors and confidence intervals, and interpreting effect sizes. Students will learn how to use Bayes' theorem/law to calculate conditional probabilities and evaluate Bayesian estimators for parameters of various distributions using conjugated priors. The second subject, Numerical Analysis, is focused on the application of numerical methods to solve mathematical problems. The course will provide an overview of numerical differentiation, numerical integration, root finding, optimization, and differential equations, and characterize their propagated errors. By the end of the course, students will have an understanding of Bayesian Statistics and Numerical Analysis, and the ability to apply these concepts in their research
Teaching method	The course consists of lectures, tutorials, and computer labs. In the tutorial sessions held four times a week, which last for 1.5 hours, students can benefit from a combination of lectures and computer labs.
Scheduled	Quarter2, Year1
Assessment	The final grade will be composed based on two computer exams: Bayesian Statistics — 50% Numerical Analysis — 50%
Competences	Conduct critical and creative research Model meaningful information
Entrance requirements	This module presupposes a basis in programming skills and calculus. Prior to enrolling in this module, students are expected to demonstrate their

	competency by fulfilling the requirements of BFVM23PREPROGR1 BFVM23PREPDATASC1 or equivalent courses.
Planning	Bayesian statistics Week 1: Overview estimation techniques, normal distribution Week 2: Normal distribution, linear regression Week 3: Other continuous distributions Week 4: Discrete distributions, Bayes law Week 5: Bayesian estimators Week 6: Bootstrapping, confidence interval Week 7: exam training Numerical analysis Week 1: Discrete number representations Week 2: Numerical differentiation Week 3: Numerical Integration Week 4: Root finding Week 5: Optimization Week 6: Differential equations Week 7: Practical applications in Life Sciences
Contact time	56 hr (7 weeks x 2 subtopics x 2 2-hr blocks/week; i.e. 1 scheduled full day/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Required: "Numerical Methods in Engineering with Python 3", Jaan Kiusalaas, 2013, Cambridge University Press, ISBN: 9781107033856 (hardcover) / 9781139611282 (e-book), https://doi.org/10.1017/CBO9781139523899 Optional: Downey, A.B., 2012. Green Tea Press Think Bayes Green Tea Press
External links/sources	See Blackboard
Language	English

Module code	BFVM23DATASCNC3
Module name	Data science 3
Module designers	Dave Langers
Contact	Fenna Feenstra
Grading Teachers	Dave Langers
ECTS and grading	5
Learning outcomes	- You manipulate mathematical expressions involving real and complex numbers, scalars, vectors and matrices. - You invert and decompose matrices, assisted by computer, and diagnose and solve rank-deficient or ill-conditioned problems. - You process time-series and image data, including visualization, resampling and interpolation. - You apply linear filters and other data transformations in both the time- and frequency domains.
Description	This course introduces the fundamental concepts and techniques of linear algebra and their applications in solving problems in different areas. The course begins with an introduction to complex numbers, vectors and matrices and how to operate on these. It includes topics such as matrix determinants and trace, matrix inversion and decomposition, and characterization of matrix rank. In parallel, the course covers signal analysis. Topics covered in this section include interpolation and curve fitting, windowing, filtering and convolution, Fourier transformation, and discrete filter design. Overall

	students learn to analyze and manipulate a wide range of time series and image data to identify patterns, remove noise, and enhance signals.
Teaching method	The course consists of lectures, tutorials, and computer labs. In the tutorial sessions held four times a week, which last for 1.5 hours, students can benefit from a combination of lectures and computer labs.
Scheduled	Quarter 3, Year 1
Assessment	The final grade will be composed based on two partial exams. One written (W) and one computer exam (C) Linear Algebra — 50% — W Signal Analysis — 50% — C
Competences	Conduct critical and creative research Model meaningful information
Entrance requirements	To enroll in and be evaluated for this subject, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Linear Algebra Week 1: Gaussian elimination Week 2: Matrix algebra & matrix inversion Week 3: Complex number arithmetic Week 4: Introduction complex analysis Week 5: Matrix transformations Week 6: Operations on matrices Week 7: Eigenvalues & eigenvectors Signal Analysis Week 1: Interpolation Week 2: Curve fitting Week 3: Global and local signal approximation Week 4: Fourier transforms Week 5: Frequency-domain analysis Week 6: Signal filtering Week 7: Applications, time-series & images
Contact time	56 hr (7 weeks x 2 subtopics x 2 2-hr blocks/week; i.e. 1 scheduled full day/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Literature: Math 1410 – Elementary Linear Algebra, spring 2020 edition, university of Lethbridge, Sean Fitzpatrick, https://www.cs.uleth.ca/~fitzpat/Textbooks/Math1410_ebook.pdf “Numerical Methods in Engineering with Python 3”, Jaan Kiusalaas, 2013, Cambridge University Press, ISBN: 9781107033856 (hardcover) / 9781139611282 (e-book), https://doi.org/10.1017/CBO9781139523899
External links/sources	Essence of linear algebra videos: "NumPy Crash Course - Complete Tutorial" "Complete Python NumPy Tutorial"
Language	English

Module code	BFVM23DATASCNC4
Module name	Data Science 4
Module designers	Martijn Herber, Peter Kroon, Tsjerk Wassenaar
Contact	Fenna Feenstra
Grading Teachers	Peter Kroon, Tsjerk Wassenaar
ECTS and grading	5
Learning outcomes	- You can explain whether and how a life science data set corresponds to a graph

	- You can implement available graph-based algorithms to process data - You can explain whether and how a life science data set can be described by a multiset multilinear model - You can implement a specific multiset multilinear model for integrative modelling of data
Description	This course introduces to relational models of data, with a focus on graphs and multilinear models. The course begins with an overview of graph theory, including the concepts of graphs, trees, adjacency matrix, directed acyclic graphs, paths and cycles, tree search, shortest path, random walks, Markov chains, sorting, and algorithmic complexity. The course then delves into the analysis of complex datasets using multivariate linear models, including multiple linear regression, partial least squares, canonical correlations, singular value decomposition, and principal component analysis. Throughout the course, students will learn various methods for investigating and assessing relational features and complex datasets using graphs and multilinear models, with applications to the life sciences. Students will also gain practical experience through programming assignments and data analysis projects.
Teaching method	Each week would consist of lectures, readings, programming assignments, and problem sets to reinforce the concepts learned. The first four weeks would focus on graph theory, while the remaining three weeks would cover multivariate analysis
Scheduled	Quarter 4, Year 1
Assessment	The final grade will be composed based on two partial exams. One computer exam (C) and one assignment. The assignment is an 8 hours individual assignment. graph theory — 50% — C multivariate analysis — 50% — A
Competences	Conduct critical and creative research Model meaningful information
Entrance requirements	To enroll in and be evaluated for this subject, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Graph Theory Week 1: - Bridges of Koningsbruggen and Euler tours, What is a graph, node, edge, where to find them, Types of graphs, Programming with graphs, Dijkstra’s algorithm Week 2: - Bipartite graphs, planar graphs, embeddings, Map coloring, DiGraphs, in/out degree, strongly/weakly connected, trees, forests, DAGs, BFS, reachability. Kahn’s algorithm Week 3: - Cliques, communities, clustering coefficients, centrality, Adjacency matrix, graph Laplacian, Spectral graph theory, Spectral embedding, Markov modeling and random walks. Normalized spectral graph partitioning Week 4: Overflow - Multivariate Analysis Week 5: - Introduction overview and components - Principal Component Analysis (PCA) and Factor Analysis (FA) Week 6: - Regression (MLR, PCR, PLS-R) - Relations within and between two sets (SVD, PA, CCA)

	Week 7: - Distances and graphs (MDS, SGP) - Discriminant analysis
Contact time	56 hr (7 weeks x 2 subtopics x 2 2-hr blocks/week; i.e. 1 scheduled full day/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Matrix-Based Introduction to Multivariate Data Analysis by Kohei Adachi Graph Theory and Complex Networks ("GTCN") (c)2010 Maarten van Steen ISBN 978-90-815406-1-2 Chapters: 1, 2, 3, 6, 9. For reference only: Mathematics for Computer Science ("MCS") (c)2015 Lehman, Leighton, Meyer Available here under Creative Commons license Chapters: 1, 3, 4, 5.
External links/sources	https://networkx.org/documentation/stable/ Dijkstra Algorithm videos: https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm https://www.youtube.com/watch?v=GazC3A4OQTE https://www.youtube.com/watch?v=ySN5Wnu88nE
Language	English

Module code	BFVM23DATASCNC5
Module name	Data science 5
Module designers	Tjerk Wassenaar, Fenna Feenstra, Bart Barnard
Contact	Fenna Feenstra
Grading Teachers	Fenna Feenstra
ECTS and grading	5
Learning outcomes	- You utilize an argumentative approach to select and apply appropriate unsupervised machine learning techniques and algorithms for a given problem in life science, while also being able to evaluate their effectiveness. - You can execute the general steps of the machine learning lifecycle, including data engineering, model selection, hyperparameter tuning, and model deployment, and apply them effectively to life science problems. - You demonstrate knowledge and understanding of the challenges and limitations of unsupervised machine learning in the context of the life science problem at hand
Description	This course is an introduction to machine learning, with a particular focus on unsupervised machine learning techniques in the domain of life science. Throughout this course, students will learn about the basic concepts and techniques of unsupervised machine learning, and how to implement these including data reduction, multidimensional scaling, manifold learning, clustering, and outlier detection. They will also be introduced to some of the most widely used algorithms in these areas and how these techniques can be applied to problems in life science. Furthermore, this course will cover general steps in the machine learning lifecycle, including data engineering, model selection, hyperparameter tuning, and model deployment. By the end of this course, students will have a solid understanding of the principles and applications of machine learning in the context of life science and be well-prepared to take on more advanced topics in machine learning.
Teaching method	Method selection and model development strategies will be discussed during lectures and tutorials. Throughout the course, students will work on short weekly assignments to reinforce their understanding of the concepts

	taught. Peer feedback will be given to enable students to improve their elaboration and design
Scheduled	Quarter 1, Year 2
Assessment	At the end of the term, students will submit a portfolio of their assignments, which will be graded. In case the portfolio is considered insufficient, specific repair assignments will be given.
Competences	Conduct critical and creative research Model meaningful information Deliver organized solutions Communicate Effectively
Entrance requirements	To enroll in and be evaluated for this subject, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Week 1: Introduction to Machine Learning - Overview of Machine Learning and its applications in life science - Basics steps in Python for machine learning Week 2: Unsupervised Machine Learning - Introduction unsupervised learning - Dimensionality reduction techniques - Clustering techniques - Portfolio assignment dataset cluster and visualization Week 3: Manifold Learning - Introduction to manifold learning and its applications - Non-linear dimensionality reduction techniques - Portfolio assignment Week 4: Outlier Detection - Types of outlier detection techniques - Portfolio assignment Week 5: Machine Learning Lifecycle - Data engineering: data preprocessing, feature selection and feature engineering Model selection and hyperparameter tuning Model deployment: model evaluation and deployment strategies Week 6 – 10: Work on portfolio
Contact time	42 hr (7 weeks x 3 x 2-hr blocks/week; i.e. 1 scheduled 6 hr/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Géron, Aurélien. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media
External links/sources	https://arxiv.org/pdf/2202.02958.pdf
Language	English

Module name	BFVM23DATASCNC6
Module designers	Dave Langers, Fenna Feenstra, Bart Barnard
Contact	FEFE
Grading Teachers	Dave Langers, Bart Barnard
ECTS and grading	5
Learning outcomes	- You explain for several frequently used machine learning strategies and algorithms how they work and when they are applicable. - You implement machine learning algorithms in Python for prediction and classification - You check the validity of outcomes from the methods and algorithms used

	- You design a (pre)processing pipeline to extract features from image data - You implement a convolutional neural network to perform image classification and image recognition
Description	This course provides an overview of the key concepts and techniques used in predictive modeling, focusing on machine learning algorithms. Students will learn a wide range of machine learning algorithms, including k-nearest neighbor, logistic regression, decision trees, support vector machines, and neural networks. The course covers optimization and evaluation techniques such as ensemble techniques, feature selection, cross-validation, over-/underfitting, regularization, learning curves, confusion matrices, and ROC curves. In addition, the course includes image analysis techniques using deep learning by means of convolutional neural networks. Students will gain practical experience implementing these techniques in Python. Finally, the course concludes with a comprehensive overview of the real-world applications of artificial intelligence in the field of life sciences.
Teaching method	Method selection and model development strategies will be discussed during lectures and tutorials. Throughout the course, students will work on short weekly assignments to reinforce their understanding of the concepts taught.
Scheduled	Quarter 2, Year 2
Assessment	The student's performance in the course will be evaluated through a combination of a final individual assignment and a verbal exam, in which they will be asked to present and defend their work and choices
Competences	Conduct critical and creative research Model meaningful information Deliver organized solutions Communicate Effectively
Entrance requirements	To enroll in and be evaluated for this subject, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Week 1 - Recap ML landscape, terminology, main challenges, applications. - Recap Cycle: problem definition, data load, inspect, preprocess, split, train, validate, evaluate, finetune - Model evaluation metrics classification and regression/feature selection, keep it simple, baseline Week 2 - Linear regression / gradient descent / polynomial regression - Overfitting / Underfitting / regularization / Learning curve - Logistic regression / Support Vector Machine focus linear and Gaussian kernel Week 3 - Trees / Naïve Bayes - Optimizing: Bagging, boosting, ensemble learning - ML ops Week 4: - Multi-layer perceptron (forward-propagation, activation functions, classification vs. regression) - Training deep neural networks (loss-function, back-propagation, stochastic gradient descent, initialization - Over/underfitting (cross-validation, loss-curve, early stopping, L1/L2-regularization, dropout) Week 5:

	- Convolutional neural networks (convolutional layers, maxpooling, data augmentation) - Using pre-trained models (freezing, fine-tuning) Week 6: - Overview of AI applications - Work on final assignment Week 7: work on final assignment
Contact time	42 hr (7 weeks x 3 x 2-hr blocks/week; i.e. 1 scheduled 6hr/week in total per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Géron, Aurélien. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media. “Deep Learning with Python”, Francois Chollet, 2017/2018, Manning Publications, ISBN: 9781617294433 (paperback) / 9781638352044 (ebook),
External links/sources	https://www.manning.com/books/deep-learning-with-python
Language	English

Programming subjects

Module code	BFVM23PREPPROGR1
Module name	Preparatory course programming
Module designers	Ronald Wedema
Contact	Fenna Feenstra
Grading Teachers	Ronald Wedema / Fenna Feenstra / Arne Poortinga
ECTS and grading	5
Learning outcomes	- You apply various data types, implement functions, create and use modules, and effectively manipulate text files in Python to solve several bioinformatics problems. - You transform a given problem into a robust and flexible object-oriented software design. - You Incorporate exception-handling mechanisms into Python software solutions to ensure the robustness of the software solution. - You showcase professionalism by delivering appropriate documented and tested software solutions - You can navigate the Linux shell proficiently and perform basic text-processing tasks.
Description	Overview The course will start with introducing the basic programming concepts, code organization, data types, structures, and functions/standard libraries. Followed by more advanced technologies like the concepts of object-oriented programming. Context learning line In this course, the student will revise the basics of programming in preparation for the Programming 1 course needed for the DS for personal health project assignment. This is one of the three optional modules of the Preparatory course. It's intended for students without a sound background in programming.
Teaching method	Each week, students will have the opportunity to deepen their knowledge of Python through assignments that will receive formative feedback. The assignments will be introduced by the lecturer with relevant context and theoretical background, and students will be encouraged to actively participate in group discussions and peer review to enhance their learning experience and improve their work.
Scheduled	Quarter1, Year1
Assessment	The student can prove the learning outcomes by an individual final assignment solution. The solution should translate the problem into a modular Python software design that is executable, which deals with possible errors, and is properly documented.
Competences	Model meaningful information. Deliver organized solutions. Communicate Effectively.
Entrance requirements	A basic level of programming is required
Planning	Week 1 – 5: working on the week assignments. Week 6 – 8: working on final assignment.
Contact time	56 hr (7 weeks 2 x 4-hr blocks/week; i.e. 1 scheduled full day/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Head First Python: A Learner's Guide to the Fundamentals of Python Programming, A Brain-Friendly Guide
External links/sources	Blackboard

Language	English

Module code	BFVM23PROGRAM2
Module name	Data processing and data exploration
Module designers	Fenna Feenstra
Contact	Fenna Feenstra
Grading Teachers	Fenna Feenstra, Peter Kroon
ECTS and grading	5
Learning outcomes	- You can apply Python, Numpy and pandas to effectively clean, transform, and structure raw data into a format suitable for analysis. - You evaluate and choose appropriate data exploration, processing, analysing and visualization methods. You critically assess trade-offs between different strategies and justify the chosen approach. - You design and develop functional and visually appealing data visualizations that effectively communicate insights and finding of categorical and non-time related data. You create interactive and informative visualizations. - You demonstrate the ability to create a well-organized and well-documented codebase, with clear separation of concerns and modular components, all managed through a Git repository
Description	This course is designed to provide practical skills for processing and analyzing data, preparing it for modelling and visualization. You will learn to work with data science tools such as Jupyter Notebooks, NumPy, Pandas, Matplotlib, and Bokeh. Each week, you will practice programming techniques for loading, cleaning, analyzing, and visualizing different types of data, mostly focused on numerical and categorical data. You will explore best practices for organizing and documenting to ensure that it remains readable, understandable, and reusable over time. By the end of this course, you will have a solid understanding of the data processing pipeline and be able to use these tools and techniques to handle diverse datasets, conduct exploratory data analysis, and create effective visualizations.
Teaching method	Each week, students will have the opportunity to deepen their knowledge of Python data processing through assignments that will receive formative feedback. The assignments will be introduced by the lecturer with relevant context and theoretical background, and students will be encouraged to actively participate in group discussions and peer review to enhance their learning experience and improve their work. Upon request, additional lectures will be provided on theoretical topics.
Scheduled	Quarter 2, Year 1
Assessment	The final assignment requires students to define a research question based on at least two data sources that can be merged into a tidy dataset. The research question should be life science related. The question should be answered using an interactive visual, and if possible, tested for significance. The code should be well-documented, well-organized, and efficient, with all relevant information included in a readme file. The assessment criteria include data quality and quantity inspection, explicit assumptions and presuppositions, appropriate transformations, efficient coding, and functional, informative visualizations. Students can choose to

	use a dataset combination provided on Blackboard, two datasets from different sources, or data from their own project
Competences	Conduct critical and creative research Model meaningful information Deliver organized solutions Communicate Effectively Being responsible
Entrance requirements	BFVM23PREPPROGR1
Planning	Each week consists of two blocks of 4 hours each. Week 01-04: Weekly Assignments and Formative Feedback - At the beginning of each week, there will be a code review and an introduction to the weekly assignment. - During the first block, students will work on the weekly assignment. - During the second block, there will be lectures on theoretical concepts upon request, and students will continue working on the weekly assignment. - Formative feedback will be provided on the weekly assignments to help students improve their understanding and skills. Week 05-07: Final Assignment and Summative Feedback - During weeks 05-07, students will work on their final assignment, which will be a life science-related research question answered using interactive visualization and tested for significance. The lecturer and teaching assistance will be available for consultation and assistance. Lectures on theoretical concepts can be provided upon request. Once the assignment is submitted summative feedback will be provided on the final assignment to assess student learning and mastery of the course objectives.
Contact time	56 hr (7 weeks 2 x 4-hr blocks/week; i.e. 1 scheduled full day/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Python for Data Analysis, 3rd Edition by Wes McKinney Publisher(s): O'Reilly Media, Inc. ISBN: 9781098104030
External links/sources	External links to course material gitbook and github repository will be provided on blackboard
Language	English

Module code	BFVM23PROGRAM3
Module name	Data processing for signal and streaming data
Module designers	Fenna Feenstra, Bart Barnard
Contact	Fenna Feenstra
Grading Teachers	Fenna Feenstra, Peter Kroon, Bart Barnard
ECTS and grading	5
Learning outcomes	- You demonstrate a high level of competence in applying python and relevant libraries as well as appropriate mathematical, and statistical methods to effectively identify patterns, causal relationships, and actionable insights. - You adeptly select appropriate data analysis methods, provide sound justifications for your choice, or creatively adapt existing methods to develop solutions for the problems. - You can integrate diverse knowledge domains, effectively handle complexity, and extract meaningful information from data, even in the presence of incomplete or challenging datasets.

	- You develop a maintainable and effective (pre-)processing and evaluation pipeline for time series and or signal data and streaming data. You adhere to the fair principles. Your code is organized, well written, well documented, traceable via version control management systems, and suitably licensed.
Description	This course teaches practical skills for processing and analyzing time-series, streaming, and signal data using popular data science tools such as Jupyter Notebooks, NumPy, Pandas, and Bokeh. Throughout the course, you will practice programming techniques for loading, cleaning, analyzing, and visualizing streaming data, with an emphasis on creating maintainable solutions. By the end of the course, you will have a solid understanding of the data processing pipeline and be able to handle diverse streaming data, conduct exploratory data analysis, and create effective visualizations. The course is designed to equip you with the skills needed to work with complex data and provide new insights as a foundation for future research and exploration.
Teaching method	Each week, students will have the opportunity to deepen their knowledge of Python data processing through assignments that will receive formative feedback. The assignments will be introduced by the lecturer with relevant context and theoretical background, and students will be encouraged to actively participate in group discussions and peer review to enhance their learning experience and improve their work. Upon request, additional lectures will be provided on theoretical topics.
Scheduled	Quarter 3, Year 1
Assessment	The final assignment requires students to define a research question based on timeseries data. The research question should be life science related. The question should be answered using an interactive visual and should include signal analysis. The code should be well-documented, well-organized, and efficient, with all relevant information included in a readme file. The assessment criteria include data quality and quantity inspection, explicit assumptions and presuppositions, appropriate transformations, efficient coding, and functional, informative visualizations. Students can choose to use a data provided on Blackboard, or data from their own project
Competences	Conduct critical and creative research Model meaningful information Deliver organized solutions Communicate Effectively Being responsible
Entrance requirements	To enroll in and be evaluated for this subject, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Each week consists of two blocks of 4 hours each. Week 01-04: Weekly Assignments and Formative Feedback - At the beginning of each week, there will be a code review and an introduction to the weekly assignment. - During the first block, students will work on the weekly assignment. - During the second block, there will be lectures on theoretical concepts upon request, and students will continue working on the weekly assignment. - Formative feedback will be provided on the weekly assignments to help students improve their understanding and skills. Week 05-07: Final Assignment and Summative Feedback - During weeks 05-07, students will work on their final assignment, which will be a life science-related research question answered using interactive visualization. Students. The lecturer and teaching

	assistance will be available for consultation and assistance. Lectures on theoretical concepts can be provided upon request. Once the assignment is submitted summative feedback will be provided on the final assignment to assess student learning and mastery of the course objectives.
Contact time	56 hr (7 weeks 2 x 4-hr blocks/week; i.e. 1 scheduled full day/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Python for Data Analysis, 3rd Edition by Wes McKinney Publisher(s): O'Reilly Media, Inc. ISBN: 9781098104030
External links/sources	External links to course material gitbook and github repository will be provided on blackboard
Language	English

Module code	BFVM23PROGRAM4
Module name	OO for big data (elective)
Module designers	Bart Barnard, Martijn Herber
Contact	Fenna Feenstra
Grading Teachers	Bart Barnard
ECTS and grading	5
Learning outcomes	- You design and develop efficient (parallel) solutions for computational problems, considering scalability, efficiency, and optimization techniques like list comprehensions, generators, and map-reduce algorithms. You demonstrate a Divide and Conquer approach, in both mindset and algorithmic strategies. - You integrate SOLID principles into the design and architecture of software systems, ensuring robustness, extensibility, and maintainability. - You effectively manage the life cycle of objects and interactions between multiple classes, employing advanced strategies such as dependency injection and design patterns. - You implement testing strategies to ensure the reliability and correctness of your software solutions. - You demonstrate professionalism and deliver organized and responsible solutions to computational problems, adhering to FAIR (Findable, Accessible, Interoperable, and Reusable) principles and ethical considerations. Show awareness of broader and/or commercial applications of research outcomes, emphasizing a practical implementation focus
Description	This course teaches students how to design parallel solutions for computational problems that can't be solved by a single computer. It will cover topics such as design patterns for distributed systems, architecture and modeling, and design considerations for large-scale distributed systems. The course emphasizes the use of SOLID principles, object life cycle, and multiple class interaction to enable effective parallel solutions. Students will also learn to use list comprehensions, generators, and map-reduce techniques to design efficient parallel solutions. Finally, the course covers the use of Divide and Conquer algorithms, which help to divide a unit of work into smaller sub-problems that can be easily distributed over several machines. Students will learn to apply this mindset to their designs and develop effective parallel solutions
Teaching method	Design and test strategies will be discussed during lectures and tutorials. Throughout the course, students will work on short weekly assignments to

	reinforce their understanding of the concepts taught. Peer feedback will be given to enable students to improve their elaboration and design skills
Scheduled	Quarter 4, Year 1
Assessment	At the end of the term, students will submit a portfolio of their assignments, which will be graded. In case the portfolio is considered insufficient, specific repair assignments will be given.
Competences	Deliver organized solutions
Entrance requirements	To enroll in and be evaluated for this subject, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Week 1: Refresh UML, SOLID, and Design Patterns - Introduction to UML and its use in object-oriented design - Overview of SOLID principles and their importance in designing scalable software - Common design patterns, such as the Factory pattern and the Observer pattern Week 2: Classes and Objects, Constructors and Destructors, Object Lifecycle, Dunders - Introduction to classes and objects in Python - Constructors and destructors: what they are and how to use them - Object lifecycle management in Python - Dunder methods and their importance in object-oriented programming Week 3: Multiple Class Interaction and Modules - Interaction between multiple classes in Python - Introduction to Python modules and their use in organizing code - Designing modular and maintainable code using modules Week 4: List Comprehensions, Generators, and Map-Reduce - List comprehensions and generators: what they are and how to use them - Map-reduce techniques for parallel processing Week 5: Unit of Work and Divide and Conquer Algorithms - Divide and conquer algorithms: definition, use cases, and examples - Applying divide and conquer to parallel solution design Week 6 and 7: working on portfolio
Contact time	42 hr (7 weeks 3 x 2-hr blocks/week; i.e. 1 scheduled 6 hours/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	N/A
External links/sources	External links to e-books and websites are provided on blackboard
Language	English

Module code	BFVM23DATASCNC5
Module name	Data science 5
Module designers	Tsjerk Wassenaar, Fenna Feenstra, Bart Barnard
Contact	Fenna Feenstra
Grading Teachers	Fenna Feenstra
ECTS and grading	5
Learning outcomes	- You utilize an argumentative approach to select and apply appropriate unsupervised machine learning techniques and algorithms for a given problem in life science, while also being able to evaluate their effectiveness. - You can execute the general steps of the machine learning lifecycle, including data engineering, model selection,

	hyperparameter tuning, and model deployment, and apply them effectively to life science problems. - You demonstrate knowledge and understanding of the challenges and limitations of unsupervised machine learning in the context of the life science problem at hand
Description	This course is an introduction to machine learning, with a particular focus on unsupervised machine learning techniques in the domain of life science. Throughout this course, students will learn about the basic concepts and techniques of unsupervised machine learning, and how to implement these including data reduction, multidimensional scaling, manifold learning, clustering, and outlier detection. They will also be introduced to some of the most widely used algorithms in these areas and how these techniques can be applied to problems in life science. Furthermore, this course will cover general steps in the machine learning lifecycle, including data engineering, model selection, hyperparameter tuning, and model deployment. By the end of this course, students will have a solid understanding of the principles and applications of machine learning in the context of life science and be well-prepared to take on more advanced topics in machine learning.
Teaching method	Method selection and model development strategies will be discussed during lectures and tutorials. Throughout the course, students will work on short weekly assignments to reinforce their understanding of the concepts taught. Peer feedback will be given to enable students to improve their elaboration and design
Scheduled	Quarter 1, Year 2
Assessment	At the end of the term, students will submit a portfolio of their assignments, which will be graded. In case the portfolio is considered insufficient, specific repair assignments will be given.
Competences	Conduct critical and creative research Model meaningful information Deliver organized solutions Communicate Effectively
Entrance requirements	To enroll in and be evaluated for this subject, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Week 1: Introduction to Machine Learning - Overview of Machine Learning and its applications in life science - Basics steps in Python for machine learning Week 2: Unsupervised Machine Learning - Introduction unsupervised learning - Dimensionality reduction techniques - Clustering techniques - Portfolio assignment dataset cluster and visualization Week 3: Manifold Learning - Introduction to manifold learning and its applications - Non-linear dimensionality reduction techniques - Portfolio assignment Week 4: Outlier Detection - Types of outlier detection techniques - Portfolio assignment Week 5: Machine Learning Lifecycle - Data engineering: data preprocessing, feature selection and feature engineering - Model selection and hyperparameter tuning - Model deployment: model evaluation and deployment strategies

	Week 6 – 10: Work on portfolio
Contact time	42 hr (7 weeks x 3 x 2-hr blocks/week; i.e. 1 scheduled 6 hr/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Géron, Aurélien. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media
External links/sources	https://arxiv.org/pdf/2202.02958.pdf
Language	English

Module code	BFVM23PROGRAM6
Module name	Big data computing
Module designers	Bart Barnard, Martijn Herber
Contact	Fenna Feenstra
Grading Teachers	Bart Barnard, Martijn Herber
ECTS and grading	5
Learning outcomes	- You understand the fundamental principles and architecture of cluster computing, including its advantages and limitations compared to traditional computing systems. - You analyze the requirements of a specific problem and choose appropriate cluster computing technologies and frameworks to design and implement a solution. - You design and deploy applications on cluster computing systems using various tools and frameworks, while considering factors such as scalability, fault tolerance, and load balancing. - You develop skills in monitoring and troubleshooting cluster computing systems to identify and resolve issues related to performance, security, and availability.
Description	This course introduces cluster computing systems and their architecture, communication protocols, and fault tolerance mechanisms. It covers cluster job management systems like Slurm, and how to design and deploy applications for these systems. The course also covers Dask, a Python library for parallel computing on large datasets, and best practices in ML-OPS for efficient and scalable machine learning pipelines on cluster computing systems. Throughout the course, students will gain hands-on experience working with cluster computing systems, including designing, and implementing efficient algorithms, distributing data, and managing resources in a distributed environment. By the end of the course, students will be able to use Slurm and Dask for parallel computing tasks and implement ML-OPS best practices. Overall, the course provides the tools and knowledge necessary to design and implement efficient, scalable, and reliable cluster computing systems for various applications, including machine learning and data analysis.
Teaching method	Principles and strategies will be discussed during lectures and tutorials. Throughout the course, students will work on short weekly assignments to reinforce their understanding of the concepts taught. Peer feedback will be given to enable students to improve their skills
Scheduled	Quarter 2, Year 2

Assessment	At the end of the term, students will submit a portfolio of their assignments, which will be graded. In case the portfolio is considered insufficient, specific repair assignments will be given.
Competences	Conduct critical and creative research Model meaningful information Deliver organized solutions Communicate Effectively
Entrance requirements	To enroll in and be evaluated for this subject, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Week 1: Introduction to Cluster Computing and Slurm - Introduction to distributed computing systems and their applications - Overview of cluster computing systems and their architecture - Introduction to Slurm: features, advantages, and limitations - Setting up a basic Slurm cluster on local machines or cloud platforms - Introduction to Snakemake: features, advantages, and limitations Week 2: Advanced Slurm Features - Load balancing and resource allocation in Slurm - Fault tolerance and recovery mechanisms in Slurm - Configuring and optimizing Slurm for specific use cases - Hands-on exercises: deploying and running applications on Slurm using Snakemake Week 3: Introduction to Dask - Overview of Dask: features, advantages, and limitations - Dask as a parallel computing framework for large-scale data processing - Dask array and Dask dataframe for distributed data processing - Setting up a Dask cluster on local machines or cloud platforms - Introduction to Snakemake workflows with Dask Week 4: Advanced Dask Features - Dask task scheduling system: graph optimization and parallel execution - Dask performance optimization and scaling strategies - Dask as a data science tool: data preparation, feature engineering, and model training - Hands-on exercises: using Dask for large-scale data processing and machine learning with Snakemake Week 5: Introduction to ML-OPS - Overview of ML-OPS: challenges and best practices - Deploying machine learning models on cluster computing systems - Monitoring and debugging machine learning pipelines - Hands-on exercises: implementing ML-OPS best practices using Slurm, Dask, and Snakemake Week 6 – 7 work on portfolio
Contact time	42 hr (7 weeks 3 x 2-hr blocks/week; i.e. 1 scheduled 6 hours/week in total, per 5 EC) – including both plenary activities (e.g. theoretical lectures/demonstrations) and supervised practical work (e.g. individual/group exercises).
Literature	Literature resources will be provided on blackboard
External links/sources	External sources will be provided on blackboard
Language	English

Omics and Omics research subjects

Module code	BFVM23PREPOMICS
Module name	Preparatory course omics
Module designers	Jurre Hageman
Contact	Fenna Feenstra
Grading Teachers	Jurre Hageman
ECTS and grading	5
Learning outcomes	- You can understand basic physiological processes. - You know the core components of both prokaryotic and eukaryotic cells and know their role in the context of cell biology. - You know all the actors and components of the Central Dogma of Genetics and can describe what these are and what their role is. - You know the types of biological sequences, their characteristics, and their relationships. - You can understand basic concepts of laboratory "omics" techniques. - You know about the main methods used to analyze sequences (e.g. Blast, alignment, mapping)
Description	The students will be introduced to basic (animal) physiology, cell biology, and molecular genetics – primarily the Central Dogma. Also, the different types of biological sequences will be introduced; their properties, and ways of analyzing them (alignment, mapping, Blast). This is one of the three optional modules of the Preparatory course. It's intended for students without a background in (molecular) life science (e.g. ICT students). The content provides a base for annotating and understanding biological data.
Teaching method	Self-study (112 hours) & tutor (2 times 2 hours a week)
Scheduled	Quarter1, Year 1
Assessment	Exam
Competences	Model meaningful information Communicate Effectively
Entrance requirements	None
Planning	Week 1 – 7
Contact time	28 hr (7 weeks x 2 x 2-hr blocks/week) – theoretical lectures
Literature	https://openstax.org/details/books/biology-2e Chapter Chapter title C3 Biological Macromolecules C4 Cell Structure C10 Cell Reproduction C11 Meiosis and Sexual Reproduction C12 Mendel's Experiments and Heredity C14 DNA Structure and Function C15 Genes and Proteins C16 Gene Expression C17 Biotechnology and Genomics C17 Biotechnology and Genomics C20 Phylogenies and the History of Life
External links/sources	Blackboard https://openstax.org/details/books/biology-2e
Language	English

Module code	BFVM23DSPH
Module name	Data science for personal health
Module designers	Martijn Herber / Marcel Kempenaar
Contact	Fenna Feenstra
Grading Teachers	Martijn Herber / Marcel Kempenaar
ECTS and grading	10
Learning outcomes	- You implement an advanced web-based visualisation. - You design a useable interface to answer a research question. - implement appropriate data(base) technologies given data sources. - You translate design into a project approach and valuable IT solution. - You collaborate with team members to organise the work involved. - You pose an exact and answerable research question (hypothesis)
Description	The objective of this research project is to enhance students' comprehension of the Data Science domain, particularly in relation to personal health. To achieve this, health-related data will be visualized to answer health-related research questions. The students will apply fundamental data transformation and munging techniques to pre-process the data for visualization. Additionally, the project aims to reinforce the programming concepts taught in the preparatory course while teaching students the principles of good UI design. The project follows a cumulative design approach, wherein a basic prototype with a simple data structure is developed first, followed by more advanced iterations on the same theme. This allows students to build upon their existing knowledge and progressively enhance their skills.
Teaching method	Students will be presented with a wicked, complex, multidisciplinary problem that requires an approach to tackle at a master's level. The project is designed to challenge the student programming skills, omics knowledge, and data science proficiency, promoting learning and knowledge development. Regular project meetings will be held to provide feedback on personal development, project progress, research validity, and product applicability in the field. These meetings will facilitate interaction among groups, within groups, and with field experts. Additionally, sprint meetings will be held to ensure the project stays on track and objectives are met in a timely manner. To supplement the other modules, masterclasses on additional theoretical topics will be organized to enhance students' understanding and support their project work. These classes will provide an opportunity for students to expand their knowledge beyond the scope of the project and develop a broader understanding of relevant topics.
Scheduled	Quarters 1, 2, 3, Year 1
Assessment	The assessment will be based on the following deliverables: final presentation (group) scientific article (group) code (individual) conduct (individual)
Competences	Conduct critical and creative research, Model meaningful information, Deliver organized solutions, Communicate Effectively, Being responsible, Being entrepreneurial
Entrance requirements	None
Planning	A detailed planning will be provided on blackboard. Presence during the kick-off and sprint meetings is mandatory. During the kick-off meetings

	further agreements will be made about availability and obligations towards the research team and stakeholders
Contact time	126 hrs, 6 hours a week in general, sometimes 4, sometimes 8 depending on the week task. This includes supervised practical work.
Literature	Recommended: Tufte, E. and Graves-Morris, P., 2014. The visual display of quantitative information.; 1983
External links/sources	Blackboard
Language	English

Module code	BFVM23PRJOMICS
Module name	Integrated omics
Module designers	Lude Franke
Contact	Fenna Feenstra
Grading Teachers	Tijs van Lieshout/ Fenna Feenstra
ECTS and grading	10
Learning outcomes	- You formulate a clear, verifiable hypothesis on the basis of a client's research question; identify and understand possible Omics techniques necessary for answering a research question and hypothesis - You evaluate datasets for utility in answering a client's hypothesis - You pre-process datasets in order to be able to do inter-dataset analysis - You apply and validate data science techniques in pre-processing data and meta-analysis across datasets - You identify business and economics factors applicable to the research question and integrate them with the final conclusion (where applicable) - You present findings in a clear and scientific manner to the target audience (client, researchers, peers)
Description	This course introduces the integration of various "omics" techniques that are used to address research questions that cannot be answered by a single analysis. These techniques are both quantitative and high throughput, generating large datasets that can be analysed using advanced statistical and machine learning methods. The course begins by introducing state-of-the-art lab techniques in areas such as (meta)genomics, transcriptomics, metabolomics, proteomics, epigenomics, foodomics, imaging, and epidemiology. Students will have the opportunity to select a research project from partner research centres such as UMCG, AVEBE, KCBBE, and the Digital Society Hub that provide multiple datasets, and formulate and test hypotheses using appropriate data science techniques (Statistcs/Machine Learnng/Artificial Intelligence). A key aspect of this course is effectively communicating the methods and findings to peers and clients. This course builds upon the technologies mastered in the first research project, and when appropriate, visualizations and web techniques from the first project will be utilized to report and clarify findings.
Teaching method	Students will be presented with a wicked, complex, multidisciplinary problem that requires an approach to tackle at a master's level. The project is designed to challenge the student programming skills, omics knowledge, and data science proficiency, promoting learning and knowledge development. Regular project meetings will be held to provide feedback on personal development, project progress, research validity, and product applicability in the field. These meetings will facilitate interaction among groups, within

	groups, and with field experts. Additionally, sprint meetings will be held to ensure the project stays on track and objectives are met in a timely manner. Masterclasses on integrated omics theory will be provided by the UMCG in the quarter of the project.
Scheduled	Quarter4 Year1, Quarter 1,2 Year 2
Assessment	The assessment will be based on the following deliverables: final presentation (group) scientific article (group) code (individual) conduct (individual)
Competences	Conduct critical and creative research, Model meaningful information, Deliver organized solutions, Communicate Effectively, Being responsible, Being entrepreneurial
Entrance requirements	To enroll in and be evaluated for this subject, at least 30 credits need to be obtained. Furthermore, completion and or granted exemptions of 3 preparatory courses is required.
Planning	Presence during the kickoff and sprint meetings is mandatory. During the kickoff meetings further agreements will be made about availability and obligations towards the research team and stakeholders. The workshops provided by the UMCG are mandatory
Contact time	126 hrs. This includes theoretical background lectures, workshops, midterm presentations and progress meetings with the project supervisor. The first quarter will contain the most contact hours.
Literature	Relevant scientific papers will be provided on blackboard
External links/sources	Blackboard
Language	English

Module code	BFVM23RPS
Module name	Research and Professional skills
Module designers	Mirjam Lurvink, Tsjerk Wassenaar
Contact	Fenna Feenstra
Grading Teachers	Mirjam Lurvink, Tsjerk Wassenaar
ECTS and grading	10
Learning outcomes	- You demonstrate effective communication and collaboration skills using different communication styles. - You demonstrate taking initiative and acting autonomously in the face of challenges, while also being a team player. - You develop a simple business case for a new product or methodology and supervise the execution of a project plan to deliver projects on-time and within prescribed conditions. - You create a stimulating environment to facilitate and enhance the performance of a project team by recognizing and utilizing individual team members' strengths and personal qualities. - You evaluate the relevance and significance of scientific literature in relation to your research. - You evaluate the design and outcomes of scientific research in data sciences and/or life sciences, considering the key elements of the scientific process, applied to own work, work of others, and the relation between them. - You evaluate the reliability, validity, and reproducibility of data analysis methods and results, both of own research and that of others, before and/or after the research is conducted.

	- You evaluate and address possible ethical, legal, and societal implications of a research project. - You evaluate your own learning process and use this to make well-founded choices concerning your personal and professional development.
Description	This course is designed to enhance students' abilities to function as critical, independent, and effective scientific researchers. The course consists of several activities, including a personal assessment (DISC report) to identify preferred communication styles and develop effective communication and collaboration skills. Workshops will cover topics such as code management, version control, business case, IP patenting, and the scrum methodology to teach students to work in a professional environment. Project management lectures will train students to structure project plans, make realistic plans, and integrate effective communication and collaboration skills. The course will also include workshops on lateral thinking, scientific reading and writing, critical thinking, and ethical, legal, and societal implications. All these learning outcomes will be developed and trained throughout the entire master program.
Teaching method	To develop professional skills, students will undergo personal assessments, participate in (peer) feedback sessions, attend (guest) lectures, workshops, and tutor groups. Students can also organize masterclasses. Professional skills workshops are mandatory, and project-based learning is integrated into the course through a series of lectures. Students will receive feedback during scrum and tutor sessions. The course also includes lectures and tutorials to develop research skills. Background theory on the scientific method and experimental design will be covered in (guest) lectures, while critical thinking skills will be developed through active learning methods, including teacher-guided peer-to-peer discussions. The course includes a workshop on ethical, legal, and societal implications (ELSI), as these are an explicit aspect of critical thinking.
Scheduled	Quarter1,2, Year1, 1,2 Year 2
Assessment	Throughout the entire master's program, students will be evaluated based on their ability to meet the program outcomes through a digital portfolio. The student manual outlines the assessment criteria for each program outcome. Students will create a digital portfolio to track their development and demonstrate proficiency in the competencies described in the program outcomes. Each program outcome must have at least one corresponding proof in the portfolio, which will be assessed based on the criteria outlined for each outcome. At the end of the graduation project phase, the portfolio will be evaluated through a criterion-based interview to determine the student's overall proficiency in meeting the program outcomes.
Competences	Conduct critical and creative research Model meaningful information Deliver organized solutions Communicate Effectively Being responsible Being entrepreneurial
Entrance requirements	N/A
Planning	Weekly workshops (1 professional, 2 research skills): Personal assessment feedback session 2 times, 0.5 hour each time
Contact time	92
Literature	Professional skills material is based on: De, B.E., 1985. Six thinking hats. Boston: Little, Brown De Bono, E., 2010. Lateral thinking: a textbook of creativity. Penguin UK

	Optional literature: Gauch Jr, H.G., 2012. Scientific method in brief. Cambridge University Press. Literature will be made available on blackboard
External links/sources	Blackboard
Language	English