Biotechnology data analysis training with Jupyter Notebooks

Ulf W. Liebal [1], Rafael Schimassek [1], Iris Broderius [1], Nicole Maaßen [1], Alina Vogelgesang [2], Philipp Weyers [2], Lars M. Blank [1]

[1] iAMB - Institute of Applied Microbiology, ABBT, RWTH Aachen University, Worringerweg 1, 52074 Aachen, Germany; [2] CLS – Center for Learning Services, RWTH Aachen University, Kackertstr. 15, 52072 Aachen, Germany

Corresponding Author: Ulf Liebal, ulf.liebal@rwth-aachen.de

Abstract

Biotechnology has experienced innovations in analytics and data processing. As the volume of data and its complexity grows, new computational procedures for extracting information are developed. However, the rate of change outpaces the adaptation of biotechnology curricula, necessitating new teaching methodologies to equip biotechnologists with data analysis abilities. To simulate experimental data, we created a virtual organism simulator (*silvio*) by combining diverse cellular and sub-cellular microbial models. *silvio* was utilized to construct a computer-based instructional workflow with important steps during strain characterization and recombinant protein expression. The instructional workflow is provided as a Jupyter Notebook with comprehensive explanatory text of biotechnological facts and experiment simulations using *silvio* tools. The students conduct data analysis in Python or Excel. This instructional workflow was separately implemented in two distance courses for Master's students in biology and biotechnology. The concept of using virtual organism simulations that generate coherent results across different experiments can be used to construct consistent and motivating case studies for biotechnological data literacy.

Keywords: Biotechnology, Data Analysis, Jupyter Notebooks, Python, Recombinant Expression, Systems Biology, Mathematical Model, Cloning

Running Title: Biotech Data Analysis with Jupyter

27   **Conflict of Interest Notification Page**

28   The authors declare that no conflict of interest exist and that the funders had no role in the course

29   material and setup.

30

31   **Data Literacy in Biotechnology**

32   Biotechnology is increasingly generating vast amounts of complex data that require advanced computational analysis

33   (*1*). The experiments include, among others, multi-omics investigations (*2*), high-throughput techniques, or

34   multiplexed experiments (*3*) and computational modeling approaches demand advanced data skills (*4*). In parallel,

35   data science provides new tools to facilitate data analysis in the form of accessible machine learning tools, for example,

36   via the Python SciKit learn library (*5*) and data analysis environments like Jupyter Notebooks (*6*), Galaxy (*7*), KBase

37   (*8*), or KNIME (*9*). Particularly data analysis with Jupyter Notebooks are becoming popular (*10*) and have been used

38   to guide metabolomic data analysis (*11*), metabolic engineering (*12*, *13*), or gene expression (*14*). However, the

39   developments of analysis have rarely managed to receive adequate attention in the curriculum of biotechnology.

40   Increasing the share of data analysis and bioinformatics in the biology and biotechnology curricula are

41   excellent means for developing critical 21st-century skills by fostering inquiry-based interdisciplinary learning (*15*).

42   Motivating and activating students is particularly important because the topic can deviate from many students' prime

43   interests and skills. Several reports discuss problem-based learning and flipped classrooms in computational biology

44   (*15–18*). Also, Jupyter Notebooks are popular teaching resources in engineering (*19*, *20*) and guidelines for their setup

45   are available (*21*, *22*). A particularly useful tool for motivation can be *gamification* and *serious game* elements (*22*,

46   *23*), which contribute more strongly to positive self-assessment than exam results (*24*).

47   Several virtual organism simulations allow precise phenotypic simulations of microorganisms. Models of

48   organisms are increasingly extending from the fundamental mathematical representation into programmatic

49   environments (*25*). Python is particularly popular because of the extensive community that supplies and maintains

50   easily accessible packages to support general tasks from machine learning up to specific biological solutions (*25*). An

51   outstanding recent development of virtual organism simulations is *Vivarium* as a modular environment that enables

52   multi-scale models for realistic simulations (*26*).

53   Here, we report about a computational data analysis course for biotechnology, including growth curve

54   analysis and recombinant gene expression, in a remote seminar over three hours. The data simulation is based on a

55   collection of models that are combined into a unifying Python environment. The course is designed as a tutorial in a

56   Jupyter Notebook and although the courses were conducted distantly, small student groups jointly worked on

57   simulations and data analysis. An evaluation of the course was performed by anonymous feedback about prior

58   knowledge, competence inventory and experience, and usability of the Jupyter Notebook.

59

60   **Procedure**

61   **The Simulator of Virtual Organisms**

62   The microbial phenotypes are simulated in Python with an ensemble of models connected with the *silvio* (**si**mulator

63   of **vi**rtual **o**rganisms) framework. *silvio* generates surrogate molecular and microbiological data without the attempt

64   to reproduce real phenotypes. Instead, at each start of *silvio*, a new virtual organism is initiated with unique parameters

65   to generate realistic data regarding type, structure, and complexity. The data is simulated by models of the associated

66   experiments, including mechanistic models, *i.e.*, growth laws, and statistical models, *i.e.*, random forest for gene

67   expression prediction. Figure 1 provides an overview of the architecture of *silvio*.

68
69   _____
                                    Figure 1
70   _____
71   *silvio* is organized in a parent object (*exp*) from which all experiments and genetic manipulation is performed.

72   Two types of functions can be distinguished, one for changing host parameters and another for performing analytical

73   experiments. The host object class contains all biological relevant information, for example, the host strain (*e.g.*, *E.*

74   *coli*, *P. putida*), the maximum biomass of cultivation (strain-specific, 30-145 g_{CDW}/L), the optimal temperature (25-

75   40° C), the optimal primer length for cloning (16-28 nucleotides), a correction factor for the expression strength (30-

76   50), as well as budget information. The budget represents money which has to be invested in the initial lab equipment

77   and in each experiment. The limited budget strengthens the importance of rational experiments and precludes a

78   massive random search for the solution. The amount of investment in the initial lab equipment determines the

79   probability of experiment failures. In addition to the budget resource, the time dimension is considered such that

80   experimental functions are designed to last a few seconds. Host-changing functions alter the values stored in the host

81   object. Cloning, the only current host-changing function, duplicates the host properties with changes in the genetic

82   information. The data is stored in separate objects to facilitate processing, visualization, and export.

83   *silvio* contains several models to simulate biological processes which are measured or optimized during

84   biotechnological strain engineering and include growth, the growth constant, DNA melting temperature, and gene

85   expression (Table 1). A logistic growth model based on the Verhulst equation simulates the growth experiments to

86   determine optimal growth temperature. The maximum biomass (K) is randomly initiated between 30-100 and 45-145

87   g_{CDW}/l for *E. coli* and *P. putida*, respectively. The temperature dependence of the growth constant (r) is calculated via

88   a normal distribution and the optimal temperature is randomly initiated between 25-40° C.

89
90   _____
                                    Table 1
91   _____

92   Empirical formulas are used to calculate the DNA melting temperature and the gene expression strength. The

93   optimal primer length is randomly initiated between 16-28 nucleotides (*27*). Two equations are used to calculate the

94   optimal melting temperature, the first equation for primers <25 nucleotides, and the second equation for larger primers

95   (Table 1). Gene expression is predicted based on the promoter sequence with 40 nucleotides (nt) upstream of the open

96   reading frame. The regression is performed by a random forest machine learning module trained with measurements

97   of a σ^{70} dependent synthetic promoter library expressed in *E. coli* and *P. taiwanensis* (*14*)(Neves *et al.*, manuscript in

98   preparation). The *silvio* simulation environment is available at https://git.rwth-aachen.de/ulf.liebal/silvio and the

99   Jupyter Notebook is available at https://github.com/uliebal/RecExpSim/.

100   **General teaching setup**

101   An instructional workflow was developed to convey biotechnological principles of recombinant expression within a

102   modern data analysis environment. To increase motivation for the students, we developed a story about a fictional

103   biotech company that aims to engineer a strain for vaccine production of a coronavirus protein subunit including a

104   competition for the highest production rates. The steps included in the simulation are (1) choice of the host organism

105   and its characterization, (2) design and cloning of a promoter sequence, and (3) measurement of the final expression

106   rate. The final rates depend only on the promoter sequence and the precision by which the students measure host

107   growth properties. The principle learning outcomes are described in Table 2 and include biotechnological knowledge

108   as well as computer science procedures. During the simulation, different experiments are performed by using

109   predefined functions appropriately. Each experiment requires resources: the money-budget is limited and some

110   experiments are time-consuming to stimulate economic and effective use of the simulated experiments.

111
112   _____
113   Table 2
114   _____
        The teaching unit was conducted as part of a biotech training for biology and biotechnology Master students

115   at the RWTH Aachen University and the Westfälische University of Applied Science, respectively. The course was

116   conducted each as Zoom-Meetings with an initial introduction and a walk-through of the simulation, including the

117   solution to all steps (45 minutes). Following, participants were allotted to two-person groups in breakout rooms to

118   work autonomously for ~2 h on solving the simulation with regular (~20 minutes) support by a supervisor. The

119   participants managed to test at least one up to four promoter sequences for vaccine expression. In the last ~20 minutes,

120   all participants joined a conclusion round, during which the statistical relationship of promoter sequence and

121   expression was examined, the production rates were compared and the group best production was honored.

122 **Data analysis in recombinant expression**

123 The recombinant engineering workflow is simulated with four selected steps and data analysis tasks (see Table 3).

124 The users start by choosing a host bacterium, predefined as either *E. coli* or *P. putida*. Both are popular bacterial hosts

125 and related to important pathogens. The host choice affects the possible maximum biomass concentration and the

126 predictor for the promoter activity but does not affect the final rate. Along with the host, the user also decides how

127 much money is invested in the laboratory equipment. An increasing investment correlates hyperbolically with

128 decreasing experiment failures. The students started with 10,000 EUR starting capital, of which 10-20% are optimally

129 used for the equipment.

130
131 Table 3
132
133 **Host characterization**

134 In the second step, the user identifies the optimal growth temperature and the associated growth rate and biomass. The

135 optimal temperature and the maximum biomass are randomly initiated and the maximum growth rate at is 1/h (see

136 Methods). The user inputs a vector with the test temperatures as an argument to the experiment function from *silvio*

137 (*simulate_growth* in the experiment class) to simulate growth experiments to identify the optimal temperature. The

138 experiment for each temperature costs 100 EUR and takes ~3 s simulation time. With a low probability of ~10%,

139 depending on investment in equipment, the growth fails and remains at the inoculum level. The outcome of this

140 experiment is a CSV-file with a data formatting as generated by the instrument *GrowthProfiler* (*Enzyscreen BV*). The

141 students could choose subsequent data analysis in *Excel* or *Python*. In *Excel*, the data has to be correctly imported,

142 followed by logarithmic operation and visualization. The temperature data with the strongest biomass increase is then

143 subjected to linear regression within the linear regime of the logarithmic data to determine the growth rate and the

144 average biomass during the stationary phase in the original data. To encourage Python-based analysis, we provided

145 disordered the lines of code (Parsons puzzle). The students have to understand and rearrange the script. Two Python

146 blocks exist: for visualization (Figure 2) and linear regression to extract growth rate and biomass. Analogously to the

147 *Excel* procedure, the user identifies the optimal temperature along with growth rate and biomass.

148
149 Figure 2
150
151 **Promoter construction and cloning**

152 After characterizing the host organism, the students identify a suitable promoter sequence and cloning is simulated.

153 The key learning objectives are knowledge about the structure of a standard sigma70 promoter, including boxes that

154 control prokaryotic gene expression, computation of DNA melting temperature, and appreciation of the fickleness of

155 cloning. A 40 nt promoter reference sequence is provided:

156 GCCCA**XXXXXX**AXGCXXXCXCGTXXXGG**XXXXXX**TGCACG,

157 *X* represents positions to be replaced by standard nucleotides and the bold printed X represent biologically important

158 –35 and –10 boxes, respectively. The optimal composition of the box positions is explained in the accompanying text

159 and literature links (*28*). The cloning step simulates the integration of the promoter into the host cell. The primers for

160 cloning start with the first nucleotide of the promoter, whereas the optimal primer length is unknown and randomly

161 initiated between 16-28 nt. The user then calculates the melting temperature according to the basic formula given by

162 the equation in Table 1. The cloning experiment has a higher error rate and identical configurations will lead to

163 different results with relatively frequent cloning failures. The cloning process is time-consuming and weakly

164 predictive and thus mirrors somewhat the process of learning of a beginner. The students are motivated to try several

165 versions of promoters to improve expression.

166

167 **Simulation results and cross-connection**

168 Following the successful design of promoters, the users measure vaccine expression values, examine correlations of

169 GC content and expression and identify the most effective host. The learning objectives are that multiple promoters

170 need to be constructed to find a good one and that the GC content is not predictive of expression strength. The users

171 first measure the promoter strength of each promoter construct. This evaluation is based on a Random Forest model

172 trained on promoter libraries as described in the Materials and Methods section (*14*). The final simulated experiment

173 reports the vaccine production, but only if correct values from the host characterization experiments are provided

174 (optimal temperature, biomass, and growth rate are within 10% of the correct values). As each group will only have

175 tested 1-4 promoter variants, combining the results across the groups can lead to further insight. We investigated

176 correlations between GC-content and vaccine expression strength for all tested promoters. An online audience

177 response system (ARS NOVA) showed a blank plot with GC-content on the x-axis and expression strength on the y-

178 axis with an underlying Hot-map feedback function. The students clicked on the corresponding value pairs of their

179 tested promoters, which were stored and resulted in the plot in Figure 3. The students could appreciate that the

180 combination of all data showed no direct correlation between GC-content and vaccine expression. (ARS NOVA has

181 meanwhile been converted into a commercial product and the Hot-map function was discontinued altogether.)

182
183 Figure 3

184
185   _____
Discussion

186   The course was supported with an evaluation to estimate the didactic benefit of the teaching concept. Evaluation is a

187 critical component in optimizing and improving study programs and circumstances, as well as teaching in general. In

188 all aspects of quality development processes, feedback is an important tool: whether as an inventory at the start of a

189 project, selective feedback during the process, or as a final review of the effectiveness of measures - the achievement

190 of objectives can be tracked at all stages. Established questionnaires in this study were used to obtain comparable,

191 accurate and trustworthy data. The teaching evaluation took place at the end of the course, between the 21th of April

192 and the 25th of May 2021, via an online questionnaire implemented on the platform *SoSciSurvey*. The results were

193 evaluated with the statistics programme *R*. In total, 16 students took part in the survey. Due to the small sample size,

194 a statistically reliable analysis of the results is not guaranteed.

195

196 **Survey instruments for teaching evaluation**

197 To evaluate the educational benefit of the Jupyter Notebook based course, we assessed the prior knowledge,

198 competence experience and competence gains as well as the increase in the topic related interest. We were also

199 interested in how the course participants received the user-friendliness of the Jupyter Notebook and how they rated

200 the course overall. Competence experience was assessed using a KIM questionnaire (Short scale of intrinsic

201 motivation) (*29*). The questionnaire contained twelve perception-related statements (*e.g.*, joyfulness, satisfaction,

202 distress) and students rated the extent of agreement on a scale of 1 (full oposition) to 5 (full agreement). Competence

203 gains were explored using the Graz Evaluation Model of Competence Acquisition (GEKO) for higher education

204 courses (*30*). The GEKO contains a list of six different areas, for example "*the acquisition of theoretical, subject-*

205 *related knowledge (e.g., theories and their contexts)*" or "*the ability to work in a team (e.g., coordination of*

206 *cooperation with fellow students, division of tasks)*", reported, again on the 1-5 scale. This scale of assessment was

207 also used to evaluate increase in topic interest and the prior knowledge of students (novice to expert). A newly designed

208 question section assessed the usability: experienced independence, the user-friendliness of the interface, the topic-

209 related knowledge, and the complementarity to the parallel lecture (1-5 scale). The survey ended with the students'

210 overall evaluation of the course by seven self-developed questions (1-5 scale).

211

212 **Evaluation of the Survey**

The evaluation revealed that, on average the students stated that they had rather little prior knowledge before attending the course ($M = 1.75$, $SD = 1.13$), and that the competence gain was rated relatively high ($M = 3.66$, $SD = 0.56$). The course also succeeded to raise interest in the topics and competencies covered ($M = 3.88$, $SD = 1.02$). In particular the acquisition of competences in teamwork received the highest rating and testifies the effectiveness of digital collaborations ($M = 3.94$, $SD = 1.12$). The evaluation of the KIM questionnaire showed that the course increased intrinsic motivation and the experience of competence on average ($M = 3.37$, $SD = 0.49$). Regarding the usability, the evaluation revealed that students were initially concerned about struggling with the computational tasks, but in the end, they found it enjoyable and the simulations were rated overall as user-friendly ($M = 3.67$, $SD = 0.55$). The total score rating of the course also indicates that the students enjoyed the course ($M = 3.58$, $SD = 0.43$). (Figure 4). Overall, the feedback demonstrated that the Jupyter Notebook based course led to a generally positive learning experience.

_____

Figure 4

_____

Conclusions

We have developed a model ensemble to simulate biotechnological relevant experiments for microorganisms (*silvio*). These experiments were combined in an instructional Jupyter Notebook to teach selected data analysis steps in a recombinant gene expression project. By adding new models to *silvio*, new biotechnological experiments can be simulated and converted into instructional, motivating data analyses projects in the form of Jupyter Notebooks. Our approach decouples the computational data analysis from often lengthy, expensive, and sometimes unavailable experiments. We stress that our simulations are not meant to replace the physical experience of conducting the experiments in the laboratory but rather to support analytical lectures by providing practice in data handling and evaluation. Finally, our survey documented the positive learning experience by the students.

Acknowledgments

References

1. A. L. Oliveira, Biotechnology, Big Data and Artificial Intelligence. *Biotechnol. J.* **14**, 1800613 (2019).
2. U. W. Liebal, A. N. T. Phan, M. Sudhakar, K. Raman, L. M. Blank, Machine Learning Applications for Mass Spectrometry-Based Metabolomics. *Metabolites*. **10**, 243 (2020).
3. M. Wehrs, A. de Beaumont-Felt, A. Goranov, P. Harrigan, S. de Kok, S. Lieder, J. Vallandingham, K. Tyner, You get what you screen for: on the value of fermentation characterization in high-throughput strain improvements in industrial settings. *J. Ind. Microbiol.*

Biotechnol. **47**, 913–927 (2020).

4. X. Fang, C. J. Lloyd, B. O. Palsson, Reconstructing organisms in silico: genome-scale models and their emerging applications. *Nat. Rev. Microbiol.*, 1–13 (2020).

5. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine Learning in {P}ython. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

6. T. Kluyver, B. Ragan-Kelley, F. Perez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, C. Willing, J. D. Team, in *Proceedings of the 20th International Conference on Electronic Publishing, ELPUB 2016* (IOS Press, 2016; https://ebooks.iospress.nl/doi/10.3233/978-1-61499-649-1-87), pp. 87–90.

7. E. Afgan, D. Baker, B. Batut, M. van den Beek, D. Bouvier, M. Čech, J. Chilton, D. Clements, N. Coraor, B. A. Grüning, A. Guerler, J. Hillman-Jackson, S. Hiltemann, V. Jalili, H. Rasche, N. Soranzo, J. Goecks, J. Taylor, A. Nekrutenko, D. Blankenberg, The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. *Nucleic Acids Res.* **46**, W537–W544 (2018).

8. A. P. Arkin, R. W. Cottingham, C. S. Henry, N. L. Harris, R. L. Stevens, S. Maslov, P. Dehal, D. Ware, F. Perez, S. Canon, others, KBase: The United States Department of Energy Systems Biology Knowledgebase. *Nat Biotechnol*. **36**, 566 (2018).

9. M. R. Berthold, N. Cebron, F. Dill, G. R., T. Kötter, T. Meinl, P. Ohl, K. Thiel, B. Wiswedel, KNIME - the Konstanz information miner. *ACM SIGKDD Explor. Newsl.*, 58–61 (2009).

10. J. M. Perkel, Why Jupyter is data scientists' computational notebook of choice. *Nature*. **563**, 145–146 (2018).

11. K. M. Mendez, L. Pritchard, S. N. Reinke, D. I. Broadhurst, Toward collaborative open data science in metabolomics using Jupyter Notebooks and cloud computing. *Metabolomics*. **15**, 125 (2019).

12. G. W. Birkel, A. Ghosh, V. S. Kumar, D. Weaver, D. Ando, T. W. H. Backman, A. P. Arkin, J. D. Keasling, H. G. Martín, The JBEI quantitative metabolic modeling library (jQMM): a python library for modeling microbial metabolism. *BMC Bioinformatics*. **18**, 205 (2017).

13. J. G. R. Cardoso, K. Jensen, C. Lieven, A. S. L. Hansen, S. Galkina, M. Beber, E. Özdemir, M. J. Herrgård, H. Redestig, N. Sonnenschein, Cameo: A Python Library for Computer Aided Metabolic Engineering and Optimization of Cell Factories. *ACS Synth. Biol.* **7**, 1163–1166 (2018).

14. U. W. Liebal, S. Köbbing, L. Netze, A. M. Schweidtmann, A. Mitsos, L. M. Blank, Insight to Gene Expression From Promoter Libraries With the Machine Learning Workflow Exp2Ipynb. *Front. Bioinforma.* **1** (2021), doi:10.3389/fbinf.2021.747428.

15. B. Fahnert, Be prepared – Learning for the future. *FEMS Microbiol. Lett.* **366** (2019), doi:10.1093/femsle/fnz200.

16. P. Compeau, Establishing a computational biology flipped classroom. *PLOS Comput. Biol.* **15**, e1006764 (2019).

17. Z. Li, B. Liu, S. H. J. Li, C. G. King, Z. Gitai, N. S. Wingreen, Modeling microbial metabolic trade-offs in a chemostat. *PLoS Comput. Biol.* **16**, e1008156 (2020).

18. M. R. Dillon, E. Bolyen, A. Adamov, A. Belk, E. Borsom, Z. Burcham, J. W. Debelius, H. Deel, A. Emmons, M. Estaki, C. Herman, C. R. Keefe, J. T. Morton, R. R. M. Oliveira, A. Sanchez, A. Simard, Y. Vázquez-Baeza, M. Ziemski, H. E. Miwa, T. A. Kerere, C. Coote, R. Bonneau, R. Knight, G. Oliveira, P. Gopalasingam, B. D. Kaehler, E. K. Cope, J. L. Metcalf, M. S. R. II, N. A. Bokulich, J. G. Caporaso, Experiences and lessons learned from two virtual, hands-on microbiome bioinformatics workshops. *PLOS Comput. Biol.* **17**, e1009056 (2021).

19. A. Suárez-García, E. Arce-Fariña, M. Álvarez Hernández, M. Fernández-Gavilanes, *Comput. Appl. Eng. Educ.*, in press, doi:10.1002/cae.22383.

20. M. Seddighi, D. Allanson, G. Rothwell, K. Takrouri, Study on the use of a combination of IPython Notebook and an industry-standard package in educating a CFD course. *Comput. Appl. Eng. Educ.* **28**, 952–964 (2020).

21. A. Rule, A. Birmingham, C. Zuniga, I. Altintas, S.-C. Huang, R. Knight, N. Moshiri, M. H. Nguyen, S. B. Rosenthal, F. Pérez, P. W. Rose, Ten simple rules for writing and sharing computational analyses in Jupyter Notebooks. *PLOS Comput. Biol.* **15**, e1007007 (2019).

22. N. C. C. Brown, G. Wilson, Ten quick tips for teaching programming. *PLOS Comput. Biol.* **14**, e1006023 (2018).

23. J. Koivisto, J. Hamari, The rise of motivational information systems: A review of gamification research. *Int. J. Inf. Manage.* **45**, 191–210 (2019).

24. J. P. McEvoy, Interactive problem-solving sessions in an introductory bioscience course engaged students and gave them feedback, but did not increase their exam scores. *FEMS Microbiol. Lett.* **364**, 182 (2017).

25. A. L. R. Lubbock, C. F. Lopez, Programmatic modeling for biological systems. *Curr. Opin. Syst. Biol.* **27**, 100343 (2021).

26. E. Agmon, R. K. Spangler, C. J. Skalnik, W. Poole, S. M. Peirce, J. H. Morrison, M. W. Covert, Vivarium: an interface and engine for integrative multiscale modeling in computational biology. *bioRxiv* (2021), doi:10.1101/2021.04.27.441657.

27. L. Y. Chuang, Y. H. Cheng, C. H. Yang, Specific primer design for the polymerase chain reaction. *Biotechnol. Lett.* **35**, 1541–1549 (2013).

28. M. Paget, Bacterial sigma factors and anti-sigma factors: structure, function and distribution. *Biomolecules*. **5**, 1245–1265 (2015).

29. M. Wilde, K. Bätz, A. Kovaleva, D. Urhahne, Überprüfung einer Kurzskala intrinsischer Motivation (KIM). *Zeitschrift für Didakt. der Naturwissenchaften*. **15**, 31–45 (2009).

30. M. Paechter, B. Maier, E. Grabensberger, Evaluation medienbasierter Lehre mittels der Einschätzung des Kompetenzerwerbs: Entwicklung eines Erhebungsinstruments. *Zeitschrift für Medien*. **19**, 68–75 (2007).

305   Tables

306

307   *Table 1: Models in* silvio *for biological processes.*

Process	Model	Details
Growth	Verhulst logistic model $$P(t) = \frac{K}{1 + \left(\frac{K - P_0}{P_0}\right)e^{-rt}}$$	*P(t)*: biomass concentration *K*: max. biomass, random $P_0 = 0.1$ g$_{CDW}$/L, initial biomass *r*: growth constant, temperature dependent
Growth constant	Normal distribution	μ: mean, optimal temperature, random σ = 5: variance
DNA melting	$T_m = 2(A + T) + 4(C + G)$ $T_m = 64.9 + 0.41 \cdot GC(\%) - \dfrac{600}{N_{Nt}}$	A, C, G, T: number of bases *GC(%)*: GC content in % N_{Nt}:
Gene Expression	Random-Forest Regression	*E. coli*, *P. taiwanensis* promoter library, (Neves *et al.*, manuscript)

308

309

310   *Table 2: Principle learning outcomes of the Recombinant Expression Simulation Jupyter Notebook workflow.*

Informative	Recognize promoter architecture of -35 and -10 boxes Explain strain specificity of growth and biomass Relate factors that impact cloning efficiencies
Performative	Distinguish Python variable types Calculate DNA melting temperature Setup growth curve analysis with linear regression

311

312   *Table 3: Computational steps of the Jupyter Notebook workflow. Each step involves a set of different activities and challenges.*

Step	Activity	Challenge
1. Host choice	• Read general introduction	• Python variable assignment
2. Host characterization	• Analyze Excel data • Estimate growth curve parameters	• Variance in biomass concentration • Growth experiment fails
3. Promoter design	• Learn importance of promoter boxes • Calculate DNA melting temperature • Derive primer complementary to promoter	• Optimal primer length unknown
4. Evaluation	• Judge effectiveness of promoter • Test impact of GC-content on expression	• Multiple rounds of promoter design

313

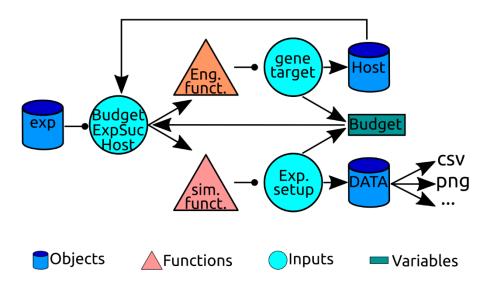314   Figures

315

Figure 1: silvio *code logic. A global experiment object (exp) is the starting point of functions for host engineering and genetic*
manipulation (Eng. funct.) and for analytical experiments (sim. funct.). Both function require input about the host for the
experiment (Host), available money (Budget) and experiment success probability (ExpSuc). The host organism is defined as an
object with user-defined properties (strain type) and randomly initiated variables (optimal temperature, maximum biomass).
Simulation data is stored in data objects with associated functions for post-processing.
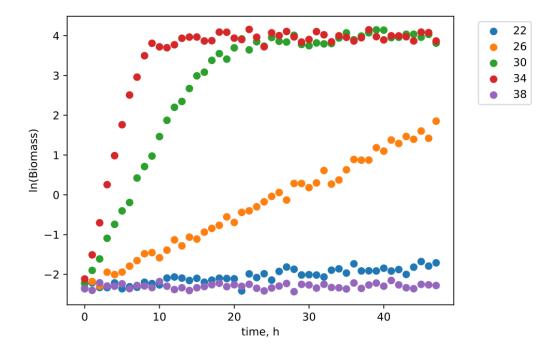
316
317
318
319
320

321

322

323

Figure 2: Example experiments (22-38° C) to determine optimal growth temperature. Five temperatures separated by 4° C were measured, which takes about 15 seconds of simulation time. The experiment at 34° C displayed the most robust linear growth increase until ~7 h. The experiment at 38° C failed, with an expected growth rate comparable to the experiment at 30-34° C. Very slow growth is measured at the suboptimal temperature of 22° C.
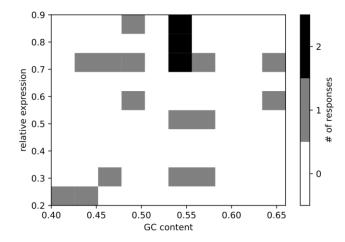
328

329

330

331   *Figure 3: Joint simulation result of all groups with the relationship of GC-content to expression strength. It shows the potential to*
332   *combine results from all groups to arrive at new knowledge when the effort of all groups is integrated. We concluded that GC*
333   *content and relative expression do not correlate linearly.*

334

$F_{\text{Welch}}(5,41.00) = 8.47$, $p = 1.42\text{e-}05$, $\widehat{\omega}_p^2 = 0.44$, $\text{CI}_{95\%}$ [0.16, 0.59], $n_{\text{obs}} = 95$
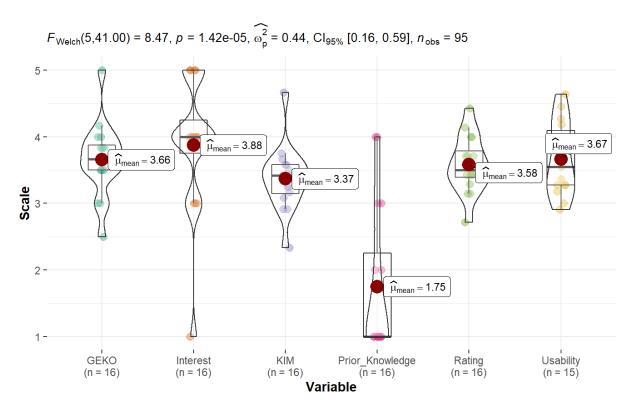
Figure 4: Distributions of the total scores of the scales regarding competence acquisition (GEKO), the students' assessments of the extent to which their interest in the topics was increased by the course (Interest), the perceived competence experience (KIM), how the students assess their prior knowledge before attending the event (Prior_Knowledge), the Overall Evaluation of the Event (Rating) and the Usability of the Jupyter Notebook.