

## Redegjørelse søknad Merittert Utdanner UiO 2023

Antall ord (uten referanser): 2489.

### En klar utvikling over tid med økende fokus på studentenes læring

Mitt navn er Lex Nederbragt. Jeg er fra Nederland og er 51 år, og har undervist på UiO i 18 år. Min aller første undervisningserfaring var som 20-årig gruppelærer for et kurs i zoologi. Mens studentene så ett tverrsnitt av en kattenyre hver, så jeg mange når jeg hjalp studentene, og opplevde at det å undervise i seg selv er å lære.

Mine første forelesninger på UiO var i 2006, i kurset BIO2120 Evolusjonsbiologi, ([ped.mappe 1.1](#)). Det gikk sikkert bra, men på det tidspunktet var jeg mest opptatt av meg selv og hva jeg som underviser gjorde [[1](#), se også [2](#)].

Det var ikke før jeg som postdoktor måtte lære meg å bruke kommandolinje-baserte verktøy, og programmering at jeg begynte å se nærmere på hvordan man kan undervise og interessere for meg for forskning rundt undervisning. Jeg oppdaget at jeg, når jeg lærer noe nyttig, ønsker å dele den nye kunnskapen med andre. Jeg holdt korte kurs og workshops for kolleger i bruk av disse verktøyene.

I 2012 ble oppdaget jeg The Carpentries, en internasjonal organisasjon av frivillige som organiserer workshops om verktøy for forskere som bruker beregninger og programmering i sin forskning ([ped.mappe 1.3](#)). Etter å ha fått dem til å holde en av deres workshops på UiO ble jeg vervet til å bli instruktør for organisasjonen. Jeg gikk gjennom deres instruktørtrening, som er forankret i pedagogisk forskning: mentale modeller, kognitiv belastningsteori, formativ versus summativ vurdering.

Jeg ønsket å etablere en lokal avdeling av The Carpentries slik at flere UiO-forskere kunne delta på workshops. Sammen med en kollega fikk vi til et samarbeid med Realfagsbiblioteket, og i 2016 etablerte vi Carpentry@UiO, som har organisert mer enn hundre workshops med deltakere fra alle fakulteter, universitetsbibliotek og IT-avdelingen. UiO har flere titalls personer utdannet som Carpentries-instruktør ([ped.mappe 1.3](#)). Denne måten å organisere etterutdanning på dugnadsbasis på fungerer veldig bra, det knytter mennesker med samme behov sammen på tvers, og skaper et praksisfellesskap [[3](#)] rundt undervisning av disse metodene på en pedagogisk gjennomtenkt måte. Carpentry@UiO har bidratt til at mer enn tusen UiO ansatte har fått opplæring i verktøy som er umiddelbart anvendbare. Deres forankring i pedagogikk har inspirert mange.

The Carpentries har betydd mye for meg. Jeg fant likesinnede personer med lidenskap for undervisning av moderne ettertraktete forskningsteknikker. Å bli instruktør, og senere instruktørtrener, endret mitt syn på undervisningen totalt. Dette gjorde meg bevisst på viktigheten av motivasjon for læring, det å begrense kognitiv last, formativ vurdering og studentaktive læringsformer. Jeg skiftet fokus fra meg selv til dem som jeg underviser til, og over til studentenes læring [[1](#)]. Jeg oppdaget at utdanning er en egen vitenskap, og ble overbevist om at, akkurat som tidligere forskning informerer oss når vi utvikler vår egen

forskning, pedagogisk forskning bør informere oss når vi utvikler vår egen undervisning. Dette har guidet mine tanker rundt undervisningsdesign fra den tiden av.

I 2012 fikk jeg min første erfaring som emneansvarlig for INF-BIO5121/9121 ([ped.mappe 1.4](#)). Dette lærte meg alt om hvordan undervisning foregår på et universitet, og viktigheten av emneevaluering.

I 2017 ble jeg med bakgrunnen i min erfaring i å undervise beregninger til forskere spurt om å utvikle et nytt emne: BIOS1100 - Innføring i beregningsmodeller for biovitenskap, et førstesemesterkurs i studieprogrammet Biovitenskap. Emnet skulle gi biologistudenter beregningskompetanse som en del av satningen på 'Computing in Science Education' ([ped.mappe 1.5](#)). Dette var en fantastisk måte å bidra til å modernisere biovitenskapsutdanningen. Jeg hadde selv som forsker vært gjennom denne moderniseringen, og er veldig motivert for å inkludere beregninger/programmering tidlig i utdanningen.

For å designe kurset tok jeg utgangspunkt i relevant pedagogisk teori og forskning om det å lære programmering. Fra forskningen lærte jeg at å lære å programmere anses generelt som vanskelig [4–7]: det er mye å ha i hodet samtidig, syntaks, problemløsningsstrategier, og den faglige konteksten. Min utfordring var å undervise programmering for studenter som (ofte) ikke har valgt det 'myke realfaget' biologi for å lære akkurat det. Når biologi og programmering skulle læres samtidig er det fare for at den kognitive belastningen blir for høy.

Dette førte meg til teorien om kognitiv belastning, som er opptatt med hvordan belastning av arbeidsminne kan optimeres for å fremme læring [8] og er relevant for programmeringsundervisning [9]. Denne teorien har vært elementær for min utdanningspraksis.

Jeg bestemte meg for å ta utgangspunkt i veiledet instruksjon ("Guided instruction") [10], for å begrense studentenes kognitive belastning. Dette er en undervisningsmetode hvor studentene får eksplisitt veiledning og støtte gjennom hele læringsprosessen, heller enn å be dem å finne ut konsepter selv [11]. I tillegg ønsket jeg å bruke studentaktiverende undervisningsformer, som jeg ble mer og mer overbevist om er viktig for læring [1,12].

Det første jeg gjorde for å begrense kognitiv belastning er bruk av samkoding, eller 'Participatory Live Coding': en aktiv læringsteknikk, og form for veiledet instruksjon, der en underviser programmerer sammen med studentene [se også min publisasjon 13] ([ped.mappe 1.7](#)) for å undervise programmeringsteknikker.

Teknikken er et sentralt element i Carpentries' workshops. I BIOS1100 brukes samkoding nå gjennom hele kurset i samkodingsgruppene ledet av gruppelærere, som får opplæring i teknikken og bruker en samkodingsoppskrift som jeg har utviklet. Det lille som er av forskning på live-koding i undervisningen viser at teknikken er like bra som, om ikke bedre enn å bruke statiske kodeeksempler [14,15] og er derfor anbefalt for undervisning i programmering [16,17].

Selv om jeg visste at samkoding kunne være nyttig i BIOS1100 trodde jeg i 2017 ikke at det var mulig å skalere opp samkoding til 200 studenter. Jeg overså at jeg dermed stolte for

mye på selvstendig læring av programmeringen ('minimal guidance' [18]). Dette førte til at mange studenter meldte til studieseksjonen at de var i ferd med å falle av fordi de ikke klarte å lære seg programmeringen. Dette kom som et sjokk over meg, og jeg ble nokså nedenfor at all arbeidet jeg hadde lagt inn i forkant føltes til ingen nytte - studentene lærte ikke stoffet!

Jeg bestemte meg å ta grep og bytte ut undervisningen de tre siste ukene med et frivillig tilbud om økter med samkoding. Tilbakemeldinger fra studenter var overveldende positivt, og mange fortalte at de nå følte at de hadde forstått det vesentlige med programmeringen.

Jeg implementerte deretter samkoding fast i undervisningen. Studentene melder tilbake at de opplever mye læringsutbytte av det, og en gruppelærer sa at 'grunnmuren i programmering sitter mye bedre'. Denne episoden lærte meg viktigheten av å få tilbakemeldinger fra studentene underveis. Det er altfor lett å anta at et gjennomtenkt undervisningsopplegg fungerer i praksis. Jeg er fortsatt stolt over at jeg klarte å snu undervisningen for å 'redde' studentene som var i ferd med å falle av.

Jeg tar også utgangspunkt i kognitiv belastning i valg av programmeringsomgivelse. Jeg valgte Jupyter notebooks som kombinerer tekst og kode, og lar brukeren både skrive og kjøre kode på samme sted ([ped.mappe 1.8](#)). Studentene liker bruken av notebooks og bekrefter at det er lett å komme i gang med.

En utfordring vi hadde var at eksamenssystemet Inspira ikke tilbyr mulighet for å bruke notebooks som innleveringsform, eller teste kode under skoleeksamen. Men, i dialog med undervisere i begynnerprogrammering for Informatikk studenter valgte jeg å akseptere dette. Argumentasjonen var at studentene ved å ikke kunne kjøre kode slapp å bruke mye tid på syntaksfeil, og kunne fokusere på å ferdigstille programmet sitt. Men tilbakemeldinger fra studentene, og min økte bevissthet om baklengs design og meningsfylt samsvar (constructive alignment, [2]) endret synet mitt.

I 2019 ba jeg om og fikk innvilget en egen eksamensinstans for JupyterHub i Silurveien, som kunne brukes av studentene å skrive og levere besvarelser som notebooks. Jeg så på dette som et viktig grep for å øke meningsfylt samsvar. Før var det et stort gap siden studentene måtte skrive eksamensbesvarelsene på en fullstendig annerledes måte enn som de hadde jobbet under kurset. At dette nå var likt har gitt studentene bedre forutsetninger for eksamen, og lar studentene i større grad demonstrere hva de har lært. Mitt håp er at det også har bidratt til mer læring [2].

## En forskende tilnærming

Forskning handler mye om å eksperimentere, og evaluere resultatene i lys av tidligere forskning og teorier. Som sagt tar jeg samme utgangspunkt når jeg designer undervisningsopplegg. Jeg har opplevd hvor viktig det er å bruke tilbakemeldinger fra studenter og gruppelærere.

Jeg ber studentene på slutten av gruppetimene å besvare en online 'exit quiz': "Hva var bra med dagens gruppetime?" og "Hva er fortsatt uklart for deg?" Svarene på disse gir verdifull

tilbakemelding for å forstå eventuelle problemer studentene sitter med. Noe av dette kan adresseres umiddelbart i neste gruppetime.

Jeg bruker aktivt resultatene av kursevalueringer ([ped.mappe 1.9](#)). Dette lærte meg for eksempel at flervalgsoppgaver fungerte veldig bra, men ikke penn-og-papir oppgavene vi prøvde ut en periode, og om behovet for flere mengdetreningsoppgaver i programmering.

Jeg er opptatt av å observere studentene mens de jobber med faget. Det er vanskelig å sette av nok tid til det, men jeg opplever alltid at jeg får mye innsikt i hvordan studentene tenker, og deres misforståelser rundt stoffet. For eksempel kan de lese en oppgavetekst på en helt annerledes måte enn den var ment. Dette hjelper meg å forbedre kursmaterialet over tid.

Ikke minst bruker jeg støttespillerne mine, gruppelærere, som er nærmest studentene og er dermed en viktig informasjonskilde. Jeg ber dem å beskrive deres erfaringer umiddelbart etter gruppeundervisningen i et delt online erfaringsdokument, som diskuteres i fellesskap på de ukentlige gruppelærermøtene. Dette lar meg rette opp feil i materialet som skal gis igjen samme uke, og er grunnlaget for forbedringer til neste gang. Dette gir også gruppelærere eierskap i undervisningsutvikling, noe de setter pris på.

Siden jeg ble mer kjent med pedagogisk litteratur og fikk erfaring med å teste ut undervisningsmetoder har en forskende tilnærming til undervisning vært viktig for meg. Jeg har også fått andre til å være med å vurdere. Jeg deltok i 2019 og 2020 i et tverrfaglig veilederteam for fire lektorstudenter som har tatt en undervisningsrettet master der de undersøkte BIOS1100-studenters holdninger til, motivasjon for, og problemløsningsstrategier i programmering og modellering i biologi ([ped.mappe 1.10](#)).

Jeg har fra begynnelsen av tatt utgangspunkt i at problemstillinger vi jobber med skal være relevante for biovitenskapstudenter. Det var gledelig å få bekreftet at mitt valg av biologisk relevante problemstillinger er støttet av tidligere forskning: dette bidrar til opplevd nytteverdi, som gir ytre motivasjon [19], og økt mestringsforventning [20]. Jeg mener nå også at samkoding som undervisningsform bidrar til mestringsfølelse, noe som er viktig for engasjement [21].

Funnene i masterprosjektene har jeg brukt aktivt for å forbedre kurset, både faglig og hvordan emnet undervises. Eksempler er økt relevans i valg av temaer som for eksempel modellering av Covid-19, og økt fokus på problemløsningsstrategier. Dette arbeidet har også blitt presentert på MNT konferansen og publisert [22,23].

## Kollegial holdning og praksis

Som tidligere nevnt var fellesskapet i The Carpentries viktig for min inngang til et mer kvalitativt blikk på undervisning. Det kollegiale fellesskapet ga rom for erfaringsdeling som jeg har prioritert siden. Jeg har holdt flere foredrag om kurset BIOS1100 og bruk av Jupyter notebooks i undervisningen både innenfor og utenfor UiO ([ped.mappe 1.6](#)).

Siden 2022 leder jeg UndervisningsVerksted ved IBV, også kalt Underverk: “en workshopserie der alle som er involvert i undervisning ved Institutt for Biovitenskap skal få mulighet til å arbeide med alle aspekter av undervisning i et kollegialt fellesskap.” Kollega Tone Gregers ved IBV fikk i 2019 støtte fra fakultetets studiekvalitetsmidler til dette.

UnderVerk har organisert flere workshops, bl.a. om Læringsmål og baklengsdesign, studentaktive læringsformer, digitale verktøy ([ped.mappe 1.11](#)). UnderVerk er en veldig fin måte å gjennomføre erfaringsutveksling på, og viser bredden i undervisningsteknikker som brukes på instituttet, og hvor lett det er å få undervisere til å dele med hverandre. UnderVerk sesjonene har økt kunnskapen blant undervisere på IBV. Som et konkret eksempel ble læringsutbyttebeskrivelser for kurset BIOS-IN5410 tilpasset til å ha aktive verb [24] etter en workshop om dette.

I store kurs overlates mye av undervisningen til gruppelærere. Dessverre er det ikke en utbredt praksis å gi dem opplæring i undervisning. Jeg valgte derfor tidlig å lære opp mine gruppelærere i noen enkle pedagogiske begreper og teknikker. Her gjenbrakte jeg materialet fra The Carpentries instruktørtrening. Til min store glede ble det senere startet opp et Lærings-Assistent-program ved fakultetet. Jeg ba om, og fikk lov til å være involvert i Lærings-assistent-seminar som tilbys til ferske gruppelærere på MSc nivå ([ped.mappe 1.12](#)). Nye gruppelærere i mine kurs anbefales dette seminaret. Jeg opplever at det å tilby noen enkel pedagogikk setter i gang en prosess hos en gruppelærer som i seg selv, sammen med økt kunnskap, fører til at den blir seg mye mer bevisst rollen som underviser, og at det gjør hen bedre i jobben sin.

Jeg er også blitt en del av undervisningsteamet for MNPED9000 – Teaching in STEM, som skal hjelpe PhD studenter med pliktarbeid til å utvikle sin evne til å fasilitere studentens læring gjennom å utforske prinsipper for undervisning og læring i høyere utdanning. ([ped.mappe 1.13](#)).

Vår 2023 ble jeg bedt om å lede prosjektet Digitalt Læringsmiljø 2030. Prosjektet skulle undersøke hvordan UiO bruker det digitale læringsmiljøet og muligheter for forbedring fremover. Dette takket jeg gledelig ja til ([ped.mappe 1.14](#)). Arbeidet ga meg en glimrende mulighet til å lære mye om hvordan undervisning er organisert, og oppleves, på UiO. Jeg kunne bruke mine erfaringer, og påvirke veien videre for UiO i det digitale. Prosjektteamet besto av ansatte fra den sentrale studieadministrasjonen, LINK, og IT avdelingen, og det var veldig interessant å høre mange forskjellige perspektiver. Prosjektet hadde lærerike innspillsmøter med studenter, undervisere, studieadministrative, og fagavdelinger. Undervisning er et samspill av mange aktører. Som underviser ser man ofte ikke de andre sidene av dette, og prosjektet var veldig lærerikt fordi jeg fikk snakket med så mange. Jeg merket at mine erfaringer med undervisning kom til nytte for rapporten. Anbefalinger i rapporten har med en del ting jeg brenner for: at det digitale blir brukt på en pedagogisk måte [25], bruk av relevant læringsteori som utgangspunkt, redusere kompleksiteten i det digitale læringsmiljøet (begrense kognitiv belastning) og et helhetsperspektiv der man ser på det digitale som en del av et større, levende læringsmiljø. Gjennom prosjektet har jeg selv lært mye om hvordan man kan lage et sammenhengende økosystem av undervisningsressurser som studentene kan bruke av det de har behov for [26]. Jeg håper og tror at rapporten kommer til å bidra til bedre undervisning og læring ved UiO.

Min reise som underviser fortsetter. Jeg ønsker flere forskningsprosjekter som studerer effekten av undervisningen min, spesielt rundt MNPED9000 kurset. Vi er godt i gang med integrering av beregninger i biovitenskapsutdanningen, men vi har en vei å gå for en full integrering. Jeg vil gå i dialog med kolleger ved instituttet for videreutvikle

studieprogrammet på dette området. Kunstig intelligens kommer til å endre mye, og jeg ønsker å fokusere på hvordan det kan bidra til at våre studenter blir selvstendige, nysgjerrige, utforskende, lærende mennesker [1].

## Referanser

1. Kugel P. How professors develop as teachers. *Studies in higher education*. 1993;18: 315–328.
2. Biggs J. What the student does: Teaching for enhanced learning. *Higher Education Research & Development*. 2012;31: 39–55. doi:[10.1080/07294360.2012.642839](https://doi.org/10.1080/07294360.2012.642839)
3. Li LC, Grimshaw JM, Nielsen C, Judd M, Coyte PC, Graham ID. Evolution of Wenger's concept of community of practice. *Implementation Science*. 2009;4: 11. doi:[10.1186/1748-5908-4-11](https://doi.org/10.1186/1748-5908-4-11)
4. Jenkins T. On the difficulty of learning to program. *Proceedings for the 3rd Annual conference of the LTSN Centre for Information and Computer Sciences*. Loughborough University; 2002. pp. 53–58. doi:[10.1145/3294016.3294024](https://doi.org/10.1145/3294016.3294024)
5. Robins A, Rountree J, Rountree N. Learning and Teaching Programming: A Review and Discussion. *Computer Science Education*. 2003;13: 137–172. doi:[10.1076/csed.13.2.137.14200](https://doi.org/10.1076/csed.13.2.137.14200)
6. Bennedsen J, Caspersen ME. Failure rates in introductory programming. *ACM SIGCSE Bulletin*. 2007;39: 32–36. doi:[10.1145/1272848.1272879](https://doi.org/10.1145/1272848.1272879)
7. Guzdial M. *Learner-Centered Design of Computing Education: Research on Computing for Everyone*. Morgan & Claypool; 2015.
8. Sweller J, van Merriënboer JJG, Paas F. Cognitive Architecture and Instructional Design: 20 Years Later. *Educational Psychology Review*. 2019. doi:[10.1007/s10648-019-09465-5](https://doi.org/10.1007/s10648-019-09465-5)
9. Caspersen ME, Bennedsen J. Instructional Design of a Programming Course: A Learning Theoretic Approach. *Proceedings of the Third International Workshop on Computing Education Research*. New York, NY, USA: ACM; 2007. pp. 111–122. doi:[10.1145/1288580.1288595](https://doi.org/10.1145/1288580.1288595)
10. Fisher D, Frey N. *Better Learning Through Structured Teaching: A Framework for the Gradual Release of Responsibility*. ASCD; 2013.
11. Kirschner PA, Sweller J, Clark RE. Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist, Discovery, Problem-Based, Experiential, and Inquiry-Based Teaching. *Educational Psychologist*. 2006;41: 75–86. doi:[10.1207/s15326985ep4102\\_1](https://doi.org/10.1207/s15326985ep4102_1)
12. Freeman S, Eddy SL, McDonough M, Smith MK, Okoroafor N, Jordt H, et al. Active learning increases student performance in science, engineering, and mathematics.



Proceedings of the National Academy of Sciences. 2014;111: 8410–8415.  
doi:[10.1073/pnas.1319030111](https://doi.org/10.1073/pnas.1319030111)

13. Nederbragt A, Harris RM, Hill AP, Wilson G. Ten quick tips for teaching with participatory live coding. PLOS Computational Biology. 2020;16: e1008090.  
doi:[10.1371/journal.pcbi.1008090](https://doi.org/10.1371/journal.pcbi.1008090)
14. Raj AGS, Patel JM, Halverson R, Halverson ER. Role of Live-coding in Learning Introductory Programming. Proceedings of the 18th Koli Calling International Conference on Computing Education Research. Koli, Finland: Association for Computing Machinery; 2018. pp. 1–8. doi:[10.1145/3279720.3279725](https://doi.org/10.1145/3279720.3279725)
15. Raj AGS, Gu P, Zhang E, R AXA, Williams J, Halverson R, et al. Live-coding vs Static Code Examples: Which is better with respect to Student Learning and Cognitive Load? Proceedings of the Twenty-Second Australasian Computing Education Conference. Melbourne VIC Australia: ACM; 2020. pp. 152–159. doi:[10.1145/3373165.3373182](https://doi.org/10.1145/3373165.3373182)
16. Brown NCC, Wilson G. Ten quick tips for teaching programming. PLOS Computational Biology. 2018;14: e1006023. doi:[10.1371/journal.pcbi.1006023](https://doi.org/10.1371/journal.pcbi.1006023)
17. Wright AM, Schwartz RS, Oaks JR, Newman CE, Flanagan SP. The why, when, and how of computing in biology classrooms. F1000Research. 2020;8: 1854.  
doi:[10.12688/f1000research.20873.2](https://doi.org/10.12688/f1000research.20873.2)
18. Clark RE, Kirschner PA, Sweller J. Putting Students on the Path to Learning. American Educator. 2012; 6–11.
19. Ryan RM, Deci EL. Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions. Contemporary Educational Psychology. 2000;25: 54–67.  
doi:[10.1006/ceps.1999.1020](https://doi.org/10.1006/ceps.1999.1020)
20. Eccles JS, Wigfield A. Motivational Beliefs, Values, and Goals. Annual Review of Psychology. 2002;53: 109–132. doi:[10.1146/annurev.psych.53.100901.135153](https://doi.org/10.1146/annurev.psych.53.100901.135153)
21. Bandura A. Self-efficacy: Toward a unifying theory of behavioral change. Psychological Review. 1977;84: 191–215. doi:[10.1037/0033-295X.84.2.191](https://doi.org/10.1037/0033-295X.84.2.191)
22. Gregers TF, Nederbragt L. Lektorstuderenter utvikler unik kompetanse og bidrar til økt kvalitet på begynneremner gjennom en undervisningsrettet master. Nordic Journal of STEM education. 2019;3: 23–27. doi:[10.5324/njsteme.v3i1.2992](https://doi.org/10.5324/njsteme.v3i1.2992)
23. Eliassen JE, Bøe MV, Nederbragt L, Gregers TF. Motivasjon for beregningsorientert biologi og sammenhengen med matematikk R2 fra videregående opplæring. Nordic Journal of STEM Education. 2021;5. doi:[10.5324/njsteme.v5i1.3917](https://doi.org/10.5324/njsteme.v5i1.3917)
24. Biggs JB, Tang CS. Teaching for quality learning at university: What the student does. Philadelphia, Pa.; Maidenhead, Berkshire, England; New York: McGraw-Hill/Society for Research into Higher Education ; Open University Press; 2011.

25. Goodyear P. Teaching as design. *Herdsa review of higher education*. 2015;2: 27–50.
26. Luckin R. The learner centric ecology of resources: A framework for using technology to scaffold learning. *Computers & Education*. 2008;50: 449–462. doi:[10.1016/j.compedu.2007.09.018](https://doi.org/10.1016/j.compedu.2007.09.018)
27. Becker EA, Koch C, Word K, Harris RM, Sane M, Nederbragt L, et al. The Carpentries Instructor Training June 2019. Zenodo. 2019. doi:[10.5281/zenodo.3258398](https://doi.org/10.5281/zenodo.3258398)
28. Aldazabal Mensa I, Alexander H, Allen J, Alsheikh-Hussain A, Attali D, Baird D, et al. Software Carpentry: The Unix Shell. Zenodo. 2017. doi:[10.5281/zenodo.278226](https://doi.org/10.5281/zenodo.278226)
29. Achterberg H, Adams J, Adelman J, Allen J, Aranda J, Bae S, et al. Software Carpentry: Programming with Python. Zenodo. 2017. doi:[10.5281/zenodo.278222](https://doi.org/10.5281/zenodo.278222)
30. Koziar KE, Munk M, Greene A, Rasel AA, Bennett E, Stevens SL, et al. Software Carpentry: Version Control with Git 2023-05. Zenodo; 2023. doi:[10.5281/zenodo.7908089](https://doi.org/10.5281/zenodo.7908089)
31. Allen J, Bachant P, Banaszkiewicz P, Bekolay T, Blischak J, Boissonneault M, et al. Software Carpentry: Automation and Make. Zenodo. 2017. doi:[10.5281/zenodo.278220](https://doi.org/10.5281/zenodo.278220)
32. Wilson G, Becker E, McKay S, Michonneau F, Williams JJ, Mayes AC, et al. Data Carpentry Wrangling Genomics Lesson. Zenodo. 2017. doi:[10.5281/zenodo.1064254](https://doi.org/10.5281/zenodo.1064254)