Welcome to The Carpentries Etherpad!

This pad is synchronized as you type, so that everyone viewing this page sees the same text. This allows you to collaborate seamlessly on documents.

Use of this service is restricted to members of The Carpentries community; this is not for general purpose use (for that, try etherpad.wikimedia.org).

Users are expected to follow our code of conduct: https://docs.carpentries.org/topic_folders/policies/code-of-conduct.html

All content is publicly available under the Creative Commons Attribution License: https://creativecommons.org/licenses/by/4.0/

Instructor slides

Day 1 - https://docs.google.com/presentation/d/1GtFiZBslxbtazB7r9xoN3ZQjgN2kziF1QpuAsqc12ho/edit#slide=id.gdc0b03ec89_0_929

Day 2 - https://docs.google.com/presentation/d/1EITo3GLB2Ry3Ypy7mQa-QnXApw1UNjihutfReujKcN8/edit#slide=id.gdc0ad2ae93_0_588

15:30
Afternoon Break
15:45
Launches and Landings
16:25
Putting It Together
16:45
Wrapping Up
16:55
Post-Training Survey

Day 2

Carpentries Zoom Room:

- Direct link: https://carpentries.zoom.us/my/carpentriesroom1?pwd=VElzMTlMcXh1bjF6YXl3L1NsTlldz09
- Password: 202020

Sign in: Name (Pronouns), Institution, Email & Twitter (optional)

1. Bret Lambert (he/him), University of Oslo (student), bretsl@math.uio.no
2. Peter Verhaar, Leiden University, p.a.f.verhaar@hum.leidenuniv.nl
3. Susanne Blumesberger, University of Vienna, susanne.blumesberger@univie.ac.at
4. Christoph Ladurner (he/him), Graz University of Technology, christoph.ladurner@tugraz.at
5. Bernadette Fritzsch (she/her), Alfred Wegener Institute for Polar and Marine Research, Bernadette.Fritzsch@awi.de
6. Kai Schwenzfeier (he/him), TU Vienna
7. Selda Sherifova (she/her) Graz University of Technology, sherifova@tugraz.at
8. Saskia Wepner (she/her) Graz University of Technology, wepner@tugraz.at
9. Edith Halvarsson (sher/her), Bodleian Libraries, University of Oxford, edith.halvarsson@bodleian.ox.ac.uk
10. Dirk Barbi(he/him), Alfred Wegener Institute for Polar and Marine Research, dirk.barbi@awi.de
11. Helene Hoffmann (she/her), Helmholtz-Zentrum Dresden-Rossendorf, helene.hoffmann@hzdr.de
12. Zongru (Doris) Shao (she/her), Helmholtz-Zentrum Dresden-Rossendorf/CASUS - Center for Advanced Systems Understanding, z.shao@hzdr.de
13. Sebastian Starke (he/him), Helmholtz-Zentrum Dresden-Rossendorf (Helmholtz AI), s.starke@hzdr.de
14. Alexander Gruber (he/him), Graz University of Technology, alexander.gruber@tugraz.at
15. Chenzi Xu (She/her), University of Oxford, chenzi.xu@ling-phil.ox.ac.uk
16. Hermann Schranzhofer (he/him), Graz University of Technology, hermann.schranzhofer@tugraz.at
17. Christian Bischof, University of Vienna, christian.bischof@univie.ac.at
18. Tereza Kalová (she/her), University of Vienna, tereza.kalova@univie.ac.at
19. Paul Gierz (pgierz@awi.de), Alfred Wegener Institute for Polar and Marine Research (he/him)
20. Giuseppe Barbieri (he/him), Professional Support, Casus Science, g.barbieri@hzdr.de

Excerise:
Write down 3 things you learnt yesterday.

- that the material on shell teaching was updated
- that it might be a good idea to think about a concept map when planning a workshop+
- that their are ways to steer a learner mindset to a growth mindeset
- Definition of "fluid representation"
- formative assessment+
- positive error framing
- Avoid Stereotypes
- Give positive feedback
- Think on all members, in the sense of problems you self don't have
- How to give good feedback+++
- How mental models work++
- How "bad" shortterm memory works ;-)+
- The ways that students can become demotivated by certain behvaiours+
- Mental load and the importance of chunking lessons (you do not want to 'speak at' someone for more than 15 minutes tops)
- The core values of the carpentries
- concept maps, inclusivness, how important feedback is
- people can only hold up to 7 items in the brain at the same time;
- steoreotypes get in the way of learning (they can increase mental load);
- a teacher best has a growth mindset: for him/her self (become better teacher through feedback)

AND for the students (so they can learn better)
- it is important to understand which mental models the students are using which might get in the way of learning
- teacher should minimize extranous load and increase germane load (e.g. by giving opportunities for reflection (like this exercise) helps)
- Feedback is important, there are lots of online tools that can help, think more about inclusion / accessiblity
- please mind the (expert) gap++
- live coding is harder than it seems
- be aware of things like presentation font size, ask for immediate feedback
- do formative assessment often to be aware of confusion/overload of learners+
- about fixed mindset++++++
- Pre-workshop survey is helpful to know learners' expectation
- Avoid dismissive language+
- Give Feedback using the matrix model
- introductions are important
- effectively use working memory and do formative assessment every 10-15 mins;
- concept maps are good for understanding and illustration;
- provide constructive feedbacks.
- equity vs. equality, encourage a growth mindset, ++
- Expert gap, the idea of concept maps, short duration of memory
- be slow rather than leave someone behind
- code of conduct is a very good basis to remind myself _how_ to communicate what is needed to give inclusive lessons
- Manage cognitive load
- Being too slow (and potentially boring some people) is better than being too quick; as this may de-motivate learners+
- Stimulate growth mindset
- giving constructive feedback
- ways to support memory retentionexpert awareness gap++
- Learners can be overwhelmed by various things (be aware)
- I need an English course!+
- mental maps are quite difficult
- feedback is essential, i really enjoy to got feedback
- structure is essential
- motivation is essential

Exercise (15 min):

In small groups, read and discuss one of the three checkout procedures described on this page. (https://carpentries.github.io/instructor-training/20-checkout/index.html if you are viewing this text elsewhere)

Make notes in the Etherpad:

1. What points do you think it is most important or helpful for people to remember?
2. What questions or points of confusion do you have, or think others might have?

When you are done, report back to the full group about that stage of the process.

Room 1: Task 3 - 5 minute teaching
-Step 1 is to register in the Etherpad (also the zoom link is found there)
https://pad.carpentries.org/teaching-demos
- a full class needs to be prepared
- and then a single segment is teached for 5 min
- there should only be minimal deviations from the material
- you also need to give feedback along the two axes to others in the group

Questions:
-what types of deviations are allowed?
-when we teach the segment do we start at the beginning or can we be asked to start in the middle?

Room 2: task 1 - contribution:
1. Data or Library carpentry contributions can be github or email; Software carpentry, community develeoped lessons, gloasrio must be done via github; there is a help wanted page
2. need to have an idea about git, need to have a github account (how to contribute, see contributed guide: swc_github_flow/for_novice_contributors.md at master · dmgt/swc_github_flow · GitHub)
3. need to notify checkout process team via email about the contribution (send the link to whom? specific trainers?---checkout@carpentries.org)
4. need to review CONTIBUTING.md
5. Proof reading must be followed by adding a new issue (how about if there are no errors?)

Room 3:
1.
- remember 90 days period for checkout
- community contribution
- there's material available for live coding session
2.
-basic understanding of GIT is substantial in order to complete the instructor training
- Not clearly formulated what a 5min presentation includes -> on the example pages for coding: it would make sense to repeat the information that the session lead for instructor training will pick one part of the live coding examples which should then be taught by you in five minutes

Room 4:
live coding demos
the live coding session will be coordinated with the lead teacher in advance, right?
Will you find out which bit of the lesson you are teaching beforehand?

Room 5:
We looked at Step 1: Make a Contribution to a Lesson/Glossary
- git knowledge is required to make a contribution to the lessons (might be hard if no prior knowledge of this)
- there are already "Help wanted" notes with issues that people can help with
- contribution does not have to be accepted in order to become an instructor

- GitHub is already able to send you emails when opening issues/pull requests, why does this need to be done again seperately?
- subjective "good commit message" direction

Room 6:
How is the operativ way to do all these three things. Is there a registration form or just using email? Seems to be necessary to have git know how!

Exercise: Schedule a discussion or demo (5 min)

- Visit the discussion Etherpad (https://pad.carpentries.org/community-discussions) to sign up for a session.
- If you would prefer to do your teaching demonstration before your discussion, visit the demo Etherpad (https://pad.carpentries.org/teaching-demos) and sign up there.
    - This demo rubric (https://data-lessons.github.io/instructor-training/demos_rubric/) is provided as a guide for Trainers evaluating potential new Instructors during the teaching demonstration.

Exercise: Up and Down

List some advantages and challenges of participatory live coding from both a learner's and an instructor's point of view in the Etherpad.

This discussion should take about 5 minutes.

- Challenge: People have different speeds and familiarity with typing+++
- Following the instructor and typing kind of blindly (because eyes on the instructor) might be challenging+
* Instructor may make mistakes; learners can then learn from these mistakes (and particularly from error messages); they learn how to 'debug'+++
* Learners can see step-by-step how comnands are constructed
* Typing in commands creates a 'neural pathway', as was explained yesterday. So it is important that learners don't only watch; they also need to do the typing themselves++++
* instructors can repond to questions immediately and show different options. There often questions such as 'What happens if I change this parameter?'

- advantage (learner): see every single step that is needed and learn why++
- advantage (instructor): have the opportunity to provide immediate help; see what learners struggle with instead of "guessing" where the issues might be
- challenge (instructor): don't be too fast when typing commands so that others can follow++
- challende (learner): interrupt instructor when they are questions without feeling disturbing
- advantage (both): live coding can serve as a possibility to "negotiate" because if instructors feel (a bit) stressed since it's live, they might have more understanding/patience for "slower" learners+

- [+] for students it is an advantage that is it slow enough to process. also when the students participate they are actually DOING somthing and not just listening -> easier to memorize.
- [+] students can make mistakes -> instructor can then focus on these actual mistakes instead of theoretical ones a student "could" make +
- [-] for the instructor this might be more difficult than power point, as things can go wrong, errors etc
- [-] a challenge for the instructor is what he/she should verbalize while typing+
- [-] it must be made sure that students have the software installed. this might take some time from the lesson.

* [+] Instructor is forced to slow down++++++++
* [+] the learner can see what is really done (ideally)
* [-] if things go wrong beyond the instructors expertise, the learner can end up more confused than before++++
* [+] the course is more interactive and the instructor can easier react to questions he/she did not anticipate
* Running into an error
* Technical Problems, which need a bit of time to solve
* Trouble finding a particular error+
* Changing set-up and therefore missing dependencis etc.+
* Far slower progress content wise (Problem with big courses)++
* Instructor challenge: the instructor may not know how to solve some errors on learners' laptop due to different operating system, different versions of software/packages, etc.+
* learner challenge: Keyboard shortcuts.
Instructors may type too fast or use keyboard shortcuts that learners may not see or follow; also different+ operating system has different short cut options.++
Different lanugage keyboards have different locations for special characters+++

Listening, watching and typing at the same time is more likely to retain attention ---> moving lessons into long-term memory
* Participants can follow processes better, but more stressful for the presenter+

difficult to explain abstract concepts e.g. object oriented programming, or even to explain a loop without a chart
why this religious normativity, why not use best of different learning concepts, what is so bad on a presentation or a graphic explanation+
to install a runtime environment for users at home

Instructor may need look away from the camera rather often to check if all the steps are covered+
Instructor may have difficulty following the steps as designed - If the steps don't follow the design learners may get confused when recapping on the lesson page
multitasking (watching and typing) might be a challenge ++
Getting everyone to start with the same basic environment may be challenging++
Instructor may not be able to get around the room efficiently if someone gets stucks+
demonstrate an example of how to code
instructors need to remember the code very well--> they have the material and can use that while coding+
Participants may fall behind if there's an issue and feel demotivated (importance of helpers ensuring that no one is left behind)
e-accessibility can be a challenge, because of the small prsentation+++++

Excerise:

- Split into groups of three.
- Assign roles, which will rotate: presenter, timekeeper, note-taker.
- Have each group member teach 3 minutes of your chosen lesson episode using live coding. For this exercise, your peers will not “code-along.” Before you begin, briefly describe what you will be teaching and what has been learned previously. Do not record this exercise.
- After each person finishes, each group member should share feedback (starting with themselves) using the same 2x2 rubric as yesterday. The timekeeper should keep feedback discussion to about 1 minute per person; this may leave some time at the end for general discussion. The note-taker should record feedback in the Etherpad.
- Trade off roles.

Copy and paste for each room - write down room number

Room 1:
python data types (Alexander)

- +good pace, clear voice ++
- - small fonts +

unix shell - filters and pipes (Kai)

- good to understand, describes what we want to do, explains everything
- slowly and clear ++
- calm voice & atmosphere
- explained what was previously done (directory structure) +

unix shell - navigating the directory tree (Bernadette)

- [+] deliberate pauses after chunks of information
- calm voice & atmosphere++
- slow but very much input
- I actually liked the ammount of input ;)+

SPSS - use of syntax for data preperation (Christian)

- + error handling++
- + step by step explanation
- - some confusion due to fast switching between windows
- + calm voice, loud and clear & good pace

Room 5:
Unix Shell
Clear presentation, good speed,
Content determined by the exercise
More practice / enthusiasm

How close to the script to we need to stay?
-Very good pace
very fast for novice learner
maybe repeat the command while typing
good pace and clear
some small details about more detailed explaination could be improved
was nice to call a command "your friend" to make clear that it is really helpful
Amy: Good pace, nice to set the goal before, what does "unzip" mean
alternatives to terminal
- slash instead of hyphen
was pwd introduced beforehand?
used opportunity to teach tab completion organically
say waht command stand for to make it easiert to remember

Room2:
Shell
- great presentation
- clear
- explaining every singol command
- missing reference about the tab
Python
- clear
- maybe too fast
- more representative variable names
- why we are doing the loop, the benefits
- title representing the idea
- great presentation
how to use pandas to load csv file
- slow paced (positive)
- reading out what was printed
- cursor covering relevant informations
- would be better to show the working directory
set up a project with gradle
- focus on terminal
- not see what the person should focus on
- explain more details
- clear visual effect with opened windows

Room 3:
Paul: git config introduction
Didn't explain the -e flag (also didn't explain what "flag" is: technical jargon)
Environment may have been too "customized"
Suggestion: offline teaching with youtube? record sessions?

Peter
Starting with data wrangling in R
Single quotes and double quotes; explanation followed by demonstration; Sometimes there are mistakes;

this is good

Susanne

calm voice
using the mouse for a better understanding,
explain the phases of the archiving-process

Room4:

Positive
Presentation

- Sebastian: good fluent english
- Sebastian: good speed
- Hermann: very detailed
- Herman: good pace
- Saskia: showed file content, motivate why we do things
- Dominik: good that previous feedback was used (explain both glob patterns), screen size good

Content

- Sebastian: be more verbous about how to open ipython
- Sebastian: shows directory
- Herman: mentioned corner cases what is allowed in variable names, also that they are case sensitive
- Saskia: good that parameters to the commandline argument were explained
- Dominik: showed example of wrong glob pattern

Constructive
Presentation

- Sebastian: /
- Herman: demotivating comments "this is not interessting"
- Saskia: internet connection problems, "tab separated" might be jargon? ask often if presenter can still be heard (here we had big problems with connection, delay of video and stopped screenshare -> not the presenters fault)
- Dominik: explain how the copy+paste was realised

Content

- Sebastian: not explained patterning
- Herman: shift enter -> only for jupyter
- Herman: exoalin case sensitive
- Dominik:

Room 6
Positive

Presentation

- talking while working
- good structure

Tereza: seemed comfortable did not come across as nervous; good story tellingx2 :), step by step, slow and loud enough voice, use of question to involve the learner, seems experienced, patient for a few secs after posing the question
Nikos

- good introduction to the topic, with real life example

Selda
Very doof speed and voice
Selda: speaking slowly and calmly

Content

- explaining the topic (pandas)
- showing advantages
- fixing a mistake and commenting on it

Tereza: mainly structured, explained abbreviations
Nikos

- quite fast
- Vent was disturbing

Selda
Confident and logical build up
very well prepared and structured

Constructive
Presentation

- sharing the screen
- loud keystrokes - using a headset might solve this
- "as we all know"

Tereza: nervosity can be improved, 'not easy to remember' - makes content scary
Nikos

- Wording 'never mind skip that'

Selda

Content

- using jargon

Tereza: using jargon, c-shell vs bash-shell (confusing jargon)
Nikos

- not ideal planing of timeline (mostly iontroduction)

Selda

Exercise: Evaluate Learning Objectives

Select one learning objective from the episode you've used for teaching practice. Copy it into the etherpad then add numbers below your objective to address the following:

1. Identify the *action* specified in the learning objective.
2. Suppose a learner had met this objective, and wanted to do more or challenge themselves on the same topic. What do you think a next level objective might look like? How is the action different?
3. Suppose a learner struggles to meet the specified objective. Is it possible to write an objective at a lower level than the one in your lesson? What would that look like?

This exercise should take about 10 minutes.

Image: A six level pyramid labelled from bottom to top: remember, understand, apply, analyze, evaluate, create. https://data-lessons.github.io/instructor-training/fig/Bloom_taxonomy.svg
Version with action words: https://carpentries.github.io/instructor-training/fig/Blooms.png
https://cdn.vanderbilt.edu/vu-wp0/wp-content/uploads/sites/59/2019/03/27124326/Blooms-Taxonomy.jpg

- http://datacarpentry.org/lessons/
- https://software-carpentry.org/lessons/
- https://librarycarpentry.org//lessons/

Room 1
The unix shell - pipes and filters
Objective: Construct command pipelines with two or more stages.
1. "understand" and "apply"
2. "analyze" as (a) the students could analyze more options of the commands and be creative to come up with different combinations (b) test if anything could go wrong such as a directory does not exist and analyze the error message
3. stepwise chaining and observing the output of each step

The pyramid is taken too far out of context and it is a misconception to even try to apply it here.

Room 2
Objective: **Use** a library function to get a list of filenames that match a wildcard pattern.
1 Use -> "apply" level
2 One level up: read the documentation for that function, check its implementation
3 One level down: try to understand what the function is doing, discussing potential problems that came up when using it in the group

Objective: **Write** a for loop to process multiple files.
1 "Write"
2 "apply"

3.

Room 4:
Objective: Explain the purpose of functions.
1. Explain (Bloom's Taxonomy: Understand)
2. The student should in a next step call build in functions (Bloom's Taxonomy: apply)
3. Repeat the definition of a function (Bloom's Taxonomy: Remember)
4. Further step could be: Define a new function (Bloom's Taxonomy: create)

Room 6

Objectives

- Use shell commands to work with directories and files
- Use shell commands to find and manipulate data

1. apply
2. "analyze" as (a) the students could analyze more options of the commands and be creative to come up with different combinations (b) test if anything could go wrong such as a directory does not exist and analyze the error message analyze - experiment; ability to look at own misstakes and go through your work and locate it
3. learn and remember the commands to navigate

Room 5
Unix Shell

- Create a directory hierarchy that matches a given diagram. --> Apply.
- Create files in that hierarchy using an editor or by copying and renaming existing files. --> Apply
- Delete, copy and move specified files and/or directories. --> Apply
- Next level analyze: Create diagram / replicate commands for existing unknown directory tree
- Lower level: explain the working /naming of the commands, concept of folders and files

Room 3
1. Understand how to use operator to many variables in an automated "operation"
2. Applying an operator to a nested list, or matrix. Due to the addinal dimension you need a second loop
3. Remember what prerequisite are needed for loops to assign variables

Exercise: Round Two

1. Before splitting into groups, read the rubric that is given to Instructor Trainers as a suggested framework for evaluating the online teaching demonstration sessions that are part of Instructor checkout. https://data-lessons.github.io/instructor-training/demos_rubric/
2. What questions do you have?
3. Return to your groups of 3 and repeat the previous exercise. This time, the presenter should incorporate changes based on feedback received, and everyone should try to ‘level up’ their feedback using the rubric for teaching demos.
4. When you are finished, add some thoughts on this process to the Etherpad: What did you change? Did it work better or worse with the change? How might you do it if you were to teach it again?

This exercise should take about 10 minutes for rubric discussion, 25 minutes for teaching, and 10 minutes for de-brief.

Room 1:

Room 6:

Selda: (+) content & presentation: covered mimicking the learner's environment, slowed down the content & speech: easier to follow; calming,
(-): pronounciation 'variable' :)
Tereza: (+) good text font size; spoke slower, not as nervous, did very good :), did not look nervous at all
Doris: clear voice, good to understand, but bit too fast in talking nad showing things on the screen, clear structure and a helpful introduction, slowing down when speaking would be helpful

Room 2
- Peter:
- Good overview beforehand
- Explained all commands and typed slowly
- Explained what we saw in the terminal
Edith:
- Explained all commands and wrote slowly
- Easy to follow
Guiseppe:
- Explained all commands and options
- Could have provided more background to the task
- Terminal window a bit small

Room 3:
Linus:
+ good explanation, detailed information, even understandable for "non-coders"
- only minor comment on small, small formulation
Saskia
- first pipe to shell then to file, explain shell '>' operators
+ very good speaking speed and presentation
Paul
+ calm and good presentation style
- for man commmand, it could help to put one little example option of the output and comment on that

Room 5:
+ good pace, good introduction, close to script
- font size
+ more energy in voice
- typing speed

Room 4:

Exercise: What Are the Challenges?

What are some of the challenges you might expect when teaching learners with a broad range of expertise? Add your thoughts in the Etherpad.

This discussion should take about 5 minutes.

- In group excerises more advanced users who already knows the anwsers may lower the particpation of other groups/mean that other learners do not get the chance to work through issues
- Some may be borde and some will not keep up++++++++++++
- keeping students engaged if they are bored as you might need to be slower
- keep learners motivated while explaining something for the third time
- handle advanced questions that will confuse novice learners+++
- learners at very different learning levels++
- usually, the content I teach is way more complex than the examples in this workshop. it might be hard to transfer all the good knowledge that we've gained within the last two days to such courses.
- Too fast for some, too slow for others and just right for a few.++
- Danger to loose both the slowest and the fastest participate to frustration / boredom++
- As everyone brings a laptop, we might loose some to their email client :D + (or slack)+
- Difficult to get the pace right: slowing down will frustrate the advanced students and speeding up may mean that the slower students simply give up++++
- the advanced student may not show up at a later session
- while some students might have troubles understanding more advanced students are bored. I think a solution could be thta more advanced students can explain to the more novice students
- examples the trainer choose should be relatable for prople from a range of disciplines/experts to keep the learners engaged
- advanced learners might want to answer all the questions or answer in a complex way, which could be discouraging to beginners+
- how to deal with learners who do not fulfill prerequisites++++
- if recognized in pre-workshop survey that student is too advanced, should we still grant access to the course?+++
- students with a higher level may digress if the teacher focusses on the ones with a lower level and those who have a lower level might digress when they do not understand the content of the course

giving feedback will be complex (especially if you have to grade the students)

"The task of a teacher is to guide two groups, one composed of olympic athletes and of disabled people through difficult terrain to three different locations at the same time without any of them to be bored or strained too much"XD - you can both have a disability and be an olympic athlete... People in wheelchairs can be damn fast with a proper hand bike. There was no attribution intended to which group is bored and which is strained ;)

Exercise: Know Your Resources

1) Take 5 minutes to read through the Code of Conduct Incident Response Guidelines:

https://docs.carpentries.org/topic_folders/policies/incident-response.html
2) Discuss what you have read in small groups. As questions arise, you may wish to refer to our complete Code of Conduct section in The Carpentries Handbook:
https://docs.carpentries.org/topic_folders/policies/index_coc.html

- What would you want to discuss or have your instructional team agree upon in advance of your workshop?
- What questions do you have about CoC enforcement?

3) Write some notes in the Etherpad.

This discussion should take about 10 minutes.

check/inform your team about the local legal guidelines.
personally i have difficulties imagining when or how that can happen.
in advance to the course, everyone should read and check that they've read and agreed on the CoC

Exercise: Teaching Together - Nuts and Bolts

With a partner, imagine that you are planning a workshop together. For this exercise, you may assume that your workshop has a separate, designated Host.

- How would you prepare to teach a workshop together?
- How would you coordinate with other members of your instructional team (e.g. Host, Helpers)?
- What kinds of things will you do to support each other during the workshop? What won't you do?

Record some notes, and share your thoughts with the group. This exercise should take about 10 minutes.
Room 2: dividing the task according to who is more comfortable with teaching with part - find common grounds
brainstorm about anticipations of the pace & potential troubles (ideally base dn teh preworkshop survey- who is summarizing the response)
practice presentation/lesson (if one of the intructors feels necessary) to collect feedback
coordinate with helpers
house keeping rules

Room 5
- Decide on roles and responsibilities, scheduling breaks, use of tools, etc.
- different chat channel for hosts, to ensure that we can communicate o the helpers if a learner is struggling and needing support (for online teaching)
- one talks as a presenter, the other one writes (e.g. notes) and takes care of technical stuff

Room 3
- have a VC or personal meeting with helpers and write down what was agreed upon -> then send around as mail so everybody can read later what was agreed upon (fix roles and responsibilities)
- instructor that is not teaching at the moment should take over organizational stuff (keep pad up to data, look who needs help)
- instructors: don't interrupt each other while teaching, don't frequently correct/criticise the teaching of the other instructor in front of the whole class+

Room 1
1. - Termino Survey
2. Set the communication channel+
3. set roles+
4. write a schedule and a script
5. Dry run for tools+

Exercise: Practice With The Carpentries Infrastructure

For this activity, your Trainer will put you in groups, but you may choose whether to work together or independently. If you work independently, you can still use your group as a resource to ask questions as they emerge.
Go to the workshop template repository: https://github.com/carpentries/workshop-template

- If you have a GitHub account (or don't mind creating one) and are comfortable doing so, follow the directions to begin creating a workshop website using your local location and today's date.
- Alternatively, have a look at the video tutorial linked on the instructions page. With any time remaining, have a look at the websites for upcoming Carpentries workshops on our website: https://carpentries.org/upcoming_workshops/
- Add your questions and thoughts on this process to the Etherpad. If you created a workshop website, add the link there as well.

that was NOT trivial
https://github.com/TheCabbageBaggage/2021-06-18-TUW

Exercise: What is in an Introduction?

Get into small groups (3-4 people) and discuss the questions below. Take notes on your answers in the Etherpad.

1. What do you hope to accomplish in a workshop introduction?
2. What information do you need to include in an introduction to accomplish these goals?

After 5 minutes, come together, and combine ideas as a large group

Room 2:
1. get to know each other (break the ice), make sure everyone is at the right place, what do we want to

achieve during the workshop, how to act, check if everybody is ready to start

2. clear instructions on min info :), display the workshop/course nameagenda,when and how to ask questions, recap of requirements (setup)

Room 3:
1. create a safe and friendly atmosphere
2. how things work (e.g. "you can always ask questions" vs. "there will be a slot reserved for queations"), how to call people (problem in german: Mrs. X or with first name)
3. communicate learning goals for the workshop?

Room 5:
1.
- get to know each other (including background)
- reference to code of conduct and explain quickly
- clarify prerequists,
- questions are fine
- 'mistakes' are ok/learning opportunity

2.
- Have people done the 'homework' (downloaded what they need)
- prepare links (e.g. to websites) in advance

Room 1
- enough time for introductions and to discuss the motivation and expectations
- talk about the preparatory materials
- agenda, timeline including the breaks

Room 6
Get to know the people is important for communication
What are the aims of the workshop
Include people directly and find out where they are

Exercise: Picking up the Pieces

In small groups or on your own, make a list of all the concepts and skills you have encountered in this training. Your list can include everything from educational/teaching theories to specific in-classroom practices.

- Giving feetback in constructive way
- Live Coding++++
- how to ~~not~~ overfill a workshop with content++++++++++
- --> important hint: we always use double the time for our workshops than is stated on the carpentry website
- Listen to feedback given by participants+++
- Never teach alone - have a instructional team+++
- feedback rounds, assessment, online tools to help+, structure for workshop material, code of conduct,

- github.io pages from github.com might be really useful++
- reduce cynicism (a bit) <--- but maintain/aquire humour+
- Find out how to use Python, GitHub (many thanks to Helene!!!), the Carpentries material, and some other tools
- maybe sing important parts of the material (also thanks to Helene) +++
- How to motivate people
- making mistakes is not necessarily a bad thing++++++++++++
- having a Growth Mindse both as a teacher and also for the students is key+++++
- interleaving content (chuncked) with exercises & formative assesment
- found out how important breaks are/would be in an online class
- preparing class: starting with goals, break down to checkpoints which can be used for formative assesment. check if the steps between checkpoints actually work out (e.g. does the code work?)
- how to better do live coding in order to foster understanding on the part of the learner
- About cognitive overload and the importance of chunking up training into smaller parts++++++
- include helpers
- if someone is really quick and tends to get bored, ask them for helping others ++
- giving constructive feedback, dismissive language
- dealing with diverse student groups+
- short-term memory is very limited! ++++
- About the processes of (sucessfully) moving short-term memory into long-term memory (good approaches include following along in coding demos, discussing issues in teams etc.)
- Set the tone! Ensuring that you encourage participatory culture early on in the workshop ++
- housekeeping rules are important++
- breaks are important!!!+++
- using that `+`-sign to signal "thumbs up" in a collaborative text document++++++👍

Exercise: One Up, One Down

Provide one up, one down feedback on the entire Instructor Training course. Remember:

- Say only one thing, and try not to duplicate. This gets harder for those who come later!
- Trainers should try not to respond, only record responses (e.g. in the Etherpad). This is also hard, but important!

- Teachers were well prepared
- Long days
- Design was well considered with logical build up
- Time in the breakout sessions were short
- Structure was good
- Many info
- Good materials are prepared
- Sometimes the instructions for the groups were not clear and needed more discussion. Copied in the Etherpad earlier
- A lot of good new information
- Better to have a way to display the task to be executed by the group
- Many thanks to trainers, it is not easy
- introduction should have been longer talked about what we wanted from the course

- - too many tabs open and links were confusing to manage
- + liked the atmosphere, open and warm
- - after lunch the structure was lacking behind yesterday
- + good that we were flexible to our needs for breaks
- + appreciated our enthusiasm as trainers we were motivated
- - it would be better to provde structure introduction to the website, it is really scattered. Info on how to naviagte the website would have been helpful
- + i liked that we had a lot of opportunity to connect with each other in the break out session
- - 2 min break every now and then to free our minds.

This exercise should take about 10 minutes.

https://www.surveymonkey.com/r/instructor_training_post_survey?workshop_id=2021-06-17-ttt-online-CEST

Resources from Feedback of Day 1

You can find more information about:

Motivation here - https://data-lessons.github.io/instructor-training/08-motivation/index.html
Mental models: https://data-lessons.github.io/instructor-training/aio/index.html
Memory and Cognitive load - https://data-lessons.github.io/instructor-training//05-memory/index.html
Linking new and old knowledge (not part of the curriculum but I found this resource) - http://teachthought.com/; https://files.eric.ed.gov/fulltext/EJ1231249.pdf

**info on Universal Design in your teaching

In the 1990s, the Center for Applied Special Technology (CAST) brought Universal Design into Education with the Universal Design in Learning (UDL) Framework. UDL places responsibility for accessibility on the course designer rather than on the learner. It states that the most inclusive approach to education is to design instruction with diverse learners in mind from the beginning. UDL is not about finding the one, best way to teach everyone. The key to UDL is creating redundancies such that learners have multiple options in how they:

1. receive
2. engage, and
3. share information.

Day 1

Carpentries Zoom Room:

- Direct link: https://carpentries.zoom.us/my/carpentriesroom1?pwd=VElzMTlMcXh1bjF6YXl3L1NsTlldz09
- Password: 202020

Sign in: Name (Pronouns), Institution, Email & Twitter (optional)
Please sign in so we can record your attendance.

1. Bret Lambert (he/him) University of Oslo (student) bretsl@math.uio.no
2. Selda Sherifova (she/her), Graz University of Technology, sherifova@tugraz.at
3. Giuseppe Barbieri (he/him) Casus Science (Professional Support) g.barbieri@hzdr.de
4. Hermann Schranzhofer (he/him), Graz University of Technology, hermann.schranzhofer@tugraz.at
5. Helene Hoffmann (she/her), Helmholtz-Zentrum Dresden-Rossendorf (Helmholtz AI), helene.hoffmann@hzdr.de
6. Kai Schwenzfeier (he/him) Technical University Vienna, schwenzfeier@iap.tuwien.ac.at
7. Alexander Gruber (he/him), Graz University of Technology, alexander.gruber@tugraz.at
8. Zongru (Doris) Shao (she/her), Helmholtz-Zentrum Dresden-Rossendorf/CASUS - Center for Advanced Systems Understanding, z.shao@hzdr.de, (Görlitz)
9. Sebastian Starke (he/him), Helmholtz-Zentrum Dresden-Rossendorf (Helmholtz AI), s.starke@hzdr.de
10. Linus Kohl (he/him) Technical University of Vienna (TU Wien), linus.kohl@tuwien.ac.at
11. Dirk Barbi (he/him) Alfred Wegener Institute for Polar and Marine Research, dirk.barbi@awi.de
12. Christoph Ladurner (he/him) Graz University of Technology, christoph.ladurner@tugraz.at
13. Bernadette Fritzsch (she/her) Alfred Wegener Institute for Polar and Marine Research, Bernadette.fritzsch@awi.de
14. Edith Halvarsson (she/her), Bodleain Libraries, University of Oxford, edith.halvarsson@bodleian.ox.ac.uk
15. Susanne Blumesberger (she/her), University of Vienna, susanne.blumesberger@univie.ac.at
16. Peter Verhaar (he.him), Leiden University, p.a.f.verhaar@hum.leidenuniv.nl, @pverhaar
17. Chenzi Xu (she/her), University of Oxford (student), chenzi.xu@ling-phil.ox.ac.uk, @ChenziAmy
18. Dominik Thalmeier (he/ him) Helmholtz-Zentrum Munich (Helmholtz AI), dominik.thalmeier@helmholtz-muenchen.de
19. Saskia Wepner (she/her) Graz University of Technology wepner@tugraz.at
20. Paul Gierz (he/him) Alfred Wegener Institute for Polar and Marine Research (pgierz@awi.de)
21. Tereza Kalová (she/her) University of Vienna, tereza.kalova@univie.ac.at
22. Christian Bischof (he/him) University of Vienna, christian.bischof@univie.ac.at
23. Nikos Gänsdorfer (he/him), University of Vienna, nikos.gaensdorfer@univie.ac.at

Please fill out the pre-training survey at https://www.surveymonkey.com/r/instructor_training_pre_survey?workshop_id=2021-06-17-ttt-online-CEST

You can keep track of the time in your current timezone at https://timeanddate.com/worldclock.

We will be using this timer - https://cuckoo.team/instructortraining

Welcome
https://data-lessons.github.io/instructor-training/01-welcome/index.html

Questions:

- What is The Carpentries and how do we approach teaching?
- What should you expect from this workshop?

Objectives:

- Identify common ground with some of your fellow workshop participants.
- Understand a general structure and core goals of The Carpentries.
- Predict what will and will not be covered in this workshop.
- Know where to find The Carpentries Code of Conduct and how to report an incident.

Code of Conduct:
To make clear what is expected, everyone participating in The Carpentries activities is required to abide by our Code of Conduct.
https://docs.carpentries.org/topic_folders/policies/code-of-conduct.html

Exercise: Take 1 minute to read Part 2. The Carpentries Code of Conduct (2.1, 2.2, 2.3)

Any form of behaviour to exclude, intimidate, or cause discomfort is a violation of the Code of Conduct. In order to foster a positive and professional learning environment we encourage you to:

- Use welcoming and inclusive language
- Be respectful of different viewpoints and experiences
- Gracefully accept constructive criticism
- Focus on what is best for the community
- Show courtesy and respect towards other community members

If you believe someone is violating the Code of Conduct, we ask that you report it to The Carpentries Code of Conduct Committee by completing this form: https://goo.gl/forms/KoUfO53Za3apOuOK2

Exercise: Getting to know each other (5 min)

- Each participant tells their name and from where they attend the training
- In breakout rooms, get to know each other by telling:
    - One aspect of teaching you are passionate about or you really enjoy
    - One thing of the Code of Conduct you really liked

Exercise: Reviewing The Carpentries Experience and Goals

For the multiple choice questions below, please place an "X" next to the response(s) that best apply to you. Then find yourself a spot in the Etherpad below to write a short response to the last question.

Have you ever participated in a Software Carpentry, Data Carpentry, or Library Carpentry Workshop?

- Yes, I have taken a workshop. +++-++
- Yes, I have been a workshop helper.+ ++-
- Yes, I organized a workshop.+-+
- No, but I am familiar with what is taught at a workshop.+ +++++
- No, and I am not familiar with what is taught at a workshop.+-++++++++

Which of these most accurately describes your teaching experience?

- I have been a graduate or undergraduate teaching assistant for a university/college course.++ ++++++++++-
- I have not had any teaching experience in the past.+-
- I have taught a seminar, workshop, or other short or informal course.++++++++++++++++
- I have been the instructor-of-record for my own university/college course.++++-
- I have taught at the primary or secondary education level. ++
- I have taught informally through outreach programs, hackathons, libraries, laboratory demonstrations, and similar activities.+++++++++-

Why are you taking this course? What goals do you have for today and tomorrow?
How best to teach new content.++++
Get some new inspiration about teaching a workshop (especially for teenagers), but generally mostly to officially become carpentry instructor
How to balance conveying content and not overloading students
It looks interesting to expand my backgrounds+++
How to teach using Git (live coding?)
How to teach basics of Research Data Management++
How to best teach programming/data management/data science based materials+
How to teach carpentries in an online environemnt
Learn about didactic skills++
to improve my teaching and learn more about learning
To improve my teaching skills (same)++++
How best to balance different learning speeds with multiple students+
How to teach coding to students with non-tech backgrounds?
Learn how to teach well. Get a scaffold for a workshop & best practices.
How to design a workshop on a topic I am not too familiar with (esp for git)
How to create a good learning atmosphere+
Improve/refresh instruction skills; learn how Carpentries workshops are officially run+
To learn some new interactive ways of teaching data management.+

This exercise should take about 5 minutes for responses, with an optional 10 for additional discussion as time permits.

Exercise

In the Etherpad, write your primary research domain or area of expertise and some aspects of the mental model you use to frame and understand the topic.

Statistical survival analysis: estimate probabilities of obtaining disease from observed patient data
Library science. Aspect of mental model: organizing, aggregating, transforming and surfacing information
Speech data analysis, build models for acoustic data to understand speech structures about a language
My Job: Using Data Analysis, Statistics & Machine Learning to help researchers answering their research questions. Some aspects of my mental model: Models (Regression, Classification, Unsupervised), Data Transformations, Data Cleaning, Labels, Structured vs unstructured Data, Tabular Data vs Time Series
Climate modelling. Aspects of mental model: connecting various programming languages and technologies to solve technically difficult numerical problems. This requires some sort of "mini" knowledge of many different parts, and it is difficult to master all elements. Knowing that it is difficult may be hard for students to understand, and they then get frustrated if something doesn't work right away.
High Performance Computing and Model Coupling; Workflow Software Development and Standardization
Industrial Data Science for creating prescriptive maintenance models by analysing structured and unstructured data and combining those with semantic technologies. Domain Problem -> Sub (nuclear) Problem -> Look up relevant approaches from other domains -> modify and apply them
Systemsbiology: for a dynamic system I have influx and efflux which can be described by different therms in a differential equation to describe the overall dynamics.
Automatic Speech Recognition; signal processing, perception experiments, statistics, machine learning
Data Steward giving scientists support for Data Management
Datamanagement
High Performance Computation - Technology (HArdware), Usage in Science (Parallelization --> different approaches, Optimization of Codes), Analysis of Bottlenecks, Development of Software
Soft tissue biomechanics: experimental design, data collection & analysis (interdisciplinary-many dots to connect: mechanics, biology, microscopy, medicine, image processing...)
Datamanagement, sociological data analysis
Jvm languages, Java and Kotlin, 3d real time rendering, native libraries ports/bindings
physics and reseach data management: literature, interviews, exchanges and asking for advice
Digital Humanities; Text and Data Mining; Natural Language Processing; Data Hermeneutics
Repositorymanagement
My main expertise is around setting up a repository with respect to the programming part of the requirements
Library and Information Science, Data Management.
Library and Information Studies, Data Librarian
Deep learning, time series forecast, natural language processing, text mining.

Surface forces, but mainly method development from coding to machine design. Mental model: Try to get a mostly complete picture of every part in my mind and then working on it.

Exercise

Part 1: On a piece of paper, draw a concept map of the same concept you discussed in the last activity, but this time without the analogy. What are 3-4 core concepts involved? How are those concepts related?

Part 2: In the Etherpad, write some notes on this process. Was it frustrating? Do you think it would be a useful exercise prior to teaching about your topic? What challenges might a novice face in creating a concept map of this kind? This exercise should take about 5 minutes.

Very hard to identify different parts of the area, i.e. parts of concept maps. I'm not even sure about what a 'concept' is
It was hard to find out what are the key concepts, i hadn't done that yet
I think the concept of the 'concept map' might not be appropriate for the audience I am teaching. Risk of overwhelming participants.
Quite frustrating, as I don't have a concept map of concept maps ;-) So didn't really know what to do.+++ +++++
very usefull - makes clear how the different parts are related to each other, gives you a idea how to teach
It can be a useful exercise; it enables me to identify the main focus; it does demand a good understanding of the domain and I can image that it can be difficult for novices to identify main concepts and connections
It was frustrating as I have no clear area of expertise, but before teaching a concept to others it would be really helpful to make up my mind about how this concept actually looks like and how all aspects relate to each other. The concept map, however, is nothing the learners should be confronted with.
it seems as if nowadays everyone has to be an "expert" in something; it was quite a challenge to think of what you are an "expert" in, when you don't feel to be an expert yet but still learning things; but for teaching / prior teaching - it makes sense to think on your teaching topic; but I can imagine that novices would have the same problem as I had to feel as an expert and to reflect on that++
Yes, it was a little bit frustrating. I think I need some more time to get into this. But maybe it could be helpful.
Quite frusttrating: I dont know what the concepts are or how they are connected. I drew twp graphs, that where very different. no idea which one is better. I did not actually know what to do.
Somewhat frustrating; had to invent a specific task to draw a concept map of. Possibly good prior to teaching to get a list of topics to cover during instruction. A novice would be hard-pressed to know how concepts are related.
can be useful but also overwhelming
Not so easy,as I thought! It is hard to find the right words and connections.+
There are many different ways how concepts can be connected, it would take many iterations to get a final version for a specific purpose. This final version, however, would be very helpful to give an overall picture of what's aimed at a lecture/course/field and how pieces (concepts) can be connected
It
It was quite frustrating and we could have used a concrete example, maybe your own concept map?++
Very useful. A novice might not have a full picture or fail to see the connections++
difficult to break down to subtopics
It takes time to summarize with a concept map. But it is very helpful to derive a solid understanding.
When you do not know about anything oder only sparsly about a problem it is hard to create a concept map
I know what a concept or mind map is, but I dont have any mapping of my expertise to a map.

Exercise

Formative Assessments
Based on your previous educational experience (or even this training so far!) what types of formative assessments do you know about?
Write your answers in the Etherpad; or go around and have each person in the group name one.

This exercise should take about 5 minutes.

Quizzes; graded/reviewed homework; snap questions during class+++
Mentimeter
Emojis (Teams, Zoom etc.)
collect feedback in shared document, polls in a video conference, room to talk/share experiences/ask questions in the main group,
Pre-workshop survey++++
Online Multuasd
Deriving a solution to a problem on blackboard with immediate feedback++
Questions++
Seeing faces showing that they don't understand++++
Results of Exercises+
Small exercises. e.g. solving a math or physics problem. and may be then present it to class.+
1) Having people read error messages and tell me what they think is going wrong, 2) Blank/broken/incomplete code for one person to fill out in front of the group++
Mentimeter (or similar): short Q&A, polls and discussion of answers afterwards, e.g., which is the correct answer, why is it easy to mix up with one of the wrong answers
Online quizzes, questionnaires, (hands on sessions in person give direct feedback)
Let the people tell me about their experiences and their literature
let them write reflections about workshops
Peer feedback+
Mentimeter, Kahoot, Quizzes, Surveys, Questionnaiers, polls, notes
Questionnaires; Collaborative documents (e.g. Etherpad or Google doc) asking stiudents to list one thing they liked and one thing they did not like; polls; General 'live' discussions during workshops
MC Quizes online, Collaborative Jupyter Notebopoks, mentimeeter
let learners find "mistakes" themselves; peer feedback
quickly querying the audience to check if they are on board with the current talk +++++

Exercise: Identify the Misconceptions

Choose one of the wrong answers to the question below and write in the Etherpad what the misconception is associated with that wrong answer.
Q: what is 27 + 15 ?
a) 42
b) 32
c) 312
d) 33
This discussion should take about 5 minutes.
b) 20+10=30 and you forget that 7+5 makes 12 not just 2, wich will raise the solution to 42+

b) only added 5 to 27 which would lead to 32, forgot the 10 of 15
c) 312 comes from adding digits in first place, then append summation of digits in second place++
b) forgetting the report
c) 3 = 2+1, 12 = 7 + 5, don't add but append the results
c) 312, comes from additing 2 and 1, and then appending 12 (5 + 7)
b) forgot to carry the 1.++++
d) looks nice! :D :D
b) ignored/forgot the 1 in the 15
* to be fair 42 is the answer to everything +++++
b) forgot the +10
to be honest it is too abstract for me and I don't get the problem since maths isn't my topic at all - so for my simple understanding there is a right and a wrong answer in maths ;-)
d) Missreading how many + - added 6 (1+5) to 27?

Exercise: What Is An Expert?

What is something that you are an expert in? How does your experience when you are acting as an expert differ from when you are not an expert?

- the confidence level is different when answering questions, more background information can be given if needed but maybe that bears the risk of overwhelming the learner and being less clear and to the point
- if I'm not an expert and teach something, I'm a little afraid that questions come up that I can't answer
- if I'm not expert and I need to teach the topic, it is easier to anticipate some questions & empathise because I have just had the similar thoughts myself
- If I'm teaching in a topic I'm expert in, it can get more challenging to answer a question to the point, as I am thinking of many ways it could be answered
- an expert knows how to answer questions in many different ways+
- an expert know where to go to look up information
- an expert knows the limits of their knowledge and expertise
- being no expert makes it easier to understand the learners who also are no experts
- thinking that you are an expert in something makes you less open to to other opinions in this topic
- as an expert you have more experience that you can share and learners can learn from
- don't be afraid to say if you don't know something rather than pretending to know and giving a nonsense/wrong answer ++++
- ability to work with concepts instead of details
- an expert is not afraid to ask for help
- feeling of what learners might not know
- more relaxed when teaching, seeing more connections to fields of knowledge outside the actual area of expertise, not afraid of questions+
- maybe gets bored / aggressive when confronted with wrong stuff more easily?
- becoming an expert needs a long time and it often not helps for be a good teacher+++
- as expert you know your own limits+
- As an expert, I normally attempt to teach general knowledge to get the leaners to build their own solutions, as a learner, I immediately try to apply new information ot my own specific problem, and maybe don't see the "big picture"
- An expert does things unconciously. Can hinder in explaining well.+++++++++

- An expert has the tools to achieve things.
- An expert knows how to make complicated things seem easy; less experienced teachers may overcomplicate things and may find it it difficult to distinguish between main things and less relevant tnings (get lost in details)

This discussion should take about 5 minutes.

Exercise: What do you use interchangeably?

In the Etherpad, share an example of words or notation that you sometimes use to accomplish or refer to the same thing. If possible, try to think of an example that might occur in a Carpentries workshop.

Building awareness of how you can represent the same concept in multiple different ways will help you avoid doing so without explanation while teaching.

This exercise should take about 5 minutes.

Script, program, code++++++++
pitch, f0, fundamental frequency OR loudness, level, volume (which are not all exactly the same but often used as synonyms; which is even worse...)sometimes intensity+
text mining, NLP,/machine learning, data mining
git: general lingo: is it "checked in", is it "commited" + also "pushed" and "uploaded"++++
' ' and " " ++++
dialects, accents, language varieties
"fitting" a model and "training" a model+++++++
In data science, there are many terms which means roughly the same: row/record/observation or column/field/variable. A tidyverse 'Tibble' is more or less the same as a 'Dataframe' in base R
directory/folder+++++
data; research data; research data management+
The likelihood of a model can be called evidence, when there are latent variables involved.
various ways how to import functions in python (import package and refer with dot or import function directly, rename the function...)
metadata/field/entry/record
In physics: a function and its approximation when certain things become small...
jvm language/java/kotlin, gradle/building tool, 3d graphics/APIs
talking about data could be a good example,
mathematical model / implementation of a model / coupled model / earth system model
intensity/counts/amplitude/...

Exercise: Awareness Gaps

Think about the area of expertise you identified for yourself earlier. What could a potential awareness gap be? What do you feel is easy for you, but difficult to learn for a novice.

This exercise should take about 5 minutes.

I still say everything, from "how do I connect to VPN again" over "what is an environment / bashrc?" to "i don't think i have github on my computer"...++
Abbriviations are often not known by novices++++++++++
Jargon language, and terms that are used to describe multiple concepts (example: RDF as both a working group, a standard, and shorthand for an implementation)
Understanding the domain problem and beeing able to apply Data Science and therefore programming skills to solve this
why documentation 'on time' is important when running experiments and/or changing codes (too much trust in memory)++++
That children don't need to be teached to find their way around the world, but they learn on their own if given the right environment
Not being scared of actually reading error messages and seeing what goes wrong+++
Some critical theoretic points+
interplay of different components of a large network
Assume that novices have installed relevant packages
Novices need much more time for understandig the topic+++
climate models come without any documentation, so typically new colleagues need personal training
Understand what could happen and what never could happen
knowing the big picture ++++
that depending if I write import numpy vs import numpy as np the code might look different but does the same+
why to set aside an independent test set, and what means independent
how to open the command line+
moving between directories on command line
machine learning is no witch craft +
assuming that learners understand the relevant terms the same way as I do++
finding a good granularity of concepts if I don't know which concepts are already present while trying not to overwhelm the learner with too many new things I think are needed to understand the problem.
the abstract model behind a "word" being used every day +
the workflow of datamanagent is easy for me, because i talked about it serval times, but for novices it is hard to understand the different steps, the same with Datamanagement Plans
keyboard shortcuts+++

Exercise: Test Your Working Memory

This website implements a short test of working memory.
https://miku.github.io/activememory/
What was your score? If you are comfortable, share your answer in the Etherpad.

- 3 (like a boss) +++
- 4+
- 5+
- 2 and then 5 (helps when you keep repeating in mind)
- 6 (first 3) helps when connecting them in a sentence
- 5
- 8 (categories help)

(in one test before, I remembered 7 words when it only showed 7 words. too much imformation may confuse working memory)

If you are unable to use this activity, ask your Trainer to implement the analog or adapted version of this test.
This exercise should take about 5 minutes.

Repeat the exercise above:

5 - I cant make up the sentences so fast +++++++++++++++
tbh that was at least partly a language problem for me, i think
3 - trying to think about sentences makes it more difficult to get the new words in my head, when I'm still thinking about the others :D
6 (5 before) putting sentences made me confused which words were actually there and which I invented myself to build a proper sentence +
6 (5 before) but coming up with a story so fast is just overwhelming my brain cells :D
6 (before 3) - story keeps changing with new words :)
Sorry, for me it's making things not better :-(
8 - thinking up a sentence that makes me laugh
8 repeating + sentencing + imagining object/persons known+
maybe we're too old ;)++ (i am for sure)++:-)
me too for sure
am I allowed to take quick note? I remembered more than 10 with quick notes
9 (categories /stories - I can share my story: a community with power risk budgets less; a guy has already writen less about bar character)
now i had pictures in my head, it worked better!
General note: I would be curious how much this depends on the speed of the words? If it is only slightly slower, would we be better? yes, i think it would be better+++

END HALF DAY / LUNCH (1 hour)

Exercise: Authentic Tasks: Think, Pair, Share

1. Think about some task you did this week that uses one or more of the skills we teach, (e.g. wrote a function, bulk downloaded data, built a plot in R, forked a repo) and explain how you would use it (or a simplified version of it) as an exercise or example in class.
2. Pair up with your neighbor and decide where this exercise fits on a graph of “short/long time to master” and “low/high usefulness”.
3. In the class Etherpad, share the task and where it fits on the graph. As a group, we will discuss how these relate back to our “teach most immediately useful first” approach.

This exercise should take about 10 minutes.

Make your notes here:

Room 1:
Pipeline processing in the shell. Teach first how it works, and a simple application such as grep | sed. Once that basic understanding is down, expand it one more command to show that it expands to effectively infinite processing. Most immediately useful after learning how to run a command, and with very little time to teach.

Room 3:
First Task: How can I import data into my analysing tool (e.g. Python or R); we think this is a quit usfull thing and it would not take a lot of time
Second Task: Writing a function for preprocessing the data; is also very usfull but will take some more time to explain working with functions and their parameters
Third Task: Clone a project using git, open it in Idea, wait for Gradle to sync and download all the dependencies and then run one of the test samples

Room 2:
Second Task: git clone Workshop material for git workshop (upper left side)
Third Task: List Comprehension in python to create a new list from an old list that contains only a part of the old list. E.g. filter out names that start with the letter "A"
Using print to print a variable. Improve output then with pprint

Room 4:
Big project: Latex
many small steps needed in between: how do I write headings, how do I add graphics, how do I make tables, usw. Can be very useful for certain kinds of research publication (maybe not social sciences and humanities), and can be a good example of how to break down a complex problem into small, learnable steps. Getting it to work is tricky, so starting with a working template will help. Encourage learners to break things to see what happens.

Room 5:
write a function (in whatever language)
y-axis: usefulness (reusability -> saves time; reduction of erros sources -> no need to correct the same error 20 times)
x-axis: time to master

Room 6:
build a machine learning model
task 1: linear regression -- most of time useful
task 2: decision tree -- another branch
task 3: ensemble methods -- based on task 1 and task 2

Exercise
How Can You Affect Motivation?

Exercise: Brainstorming Motivational Impacts
Think back to courses you have taken in the past and consider things that an instructor has said or done

that you found either motivating or demotivating. Try to think of one example in each case, and share your example under the appropriate heading in the Etherpad.

This exercise should take about 5 minutes.

Exercise

Motivating

- many repetitions of important, 'elementary' information
- use drawings to explain concepts, repeting things and making sure to get everyone on the same page again before starting on a new topic
- interesting content
-Slow phase/time to digest and reflect
- using enthusiastic and amazed voice when presenting the most abstract physics++
- interaction between participants
- examples which I know already with new methods
- professor said the Euler Equation is the proof for the existence of a platonic world
- an example which I can immediately relate to: how biological tissues grow & remodel in response to mechanical loading: imagine walking on the beaches of Crete in a hot sommer for several hours a day
- took time for questions

Jokes, informal talk, different tones, looking the audience all around, gesticulate

Demostration through experiments/multimedia (subject-specific)

- demonstrating an obvious interest in the material being taught
- telling relevant anekdotes (class about technical chemistry with a story about a chemistry plant blowing up :D)
- Discuss interesting applications; good examples+

beste practise examples (something useful and practical)

- everyday-life examples (link of matter and relevance)

many examples

- using real-life examples given by the participants during the course+

Demotivating

- going too fast during live coding without a chance to catch up and later building on that code to work+++
- using processor specifications and short names in an HPC course as if they should be familiar to everyone+
- starting every second sentence with "but" (prevents a mental model to develop maybe)
-Math Lecture Linear Algebra: Teacher said he finds linear algebra trivial and boring.
- going too slow and reiterating basics again and again.+++
- Skipping over steps/going too quickly
- "well, that's all written in the book, haven't you read it!?"++
- too slow, or something I already know+++
- "You should know that by now"+++
- talking about something I completely miss the background for and don't understand anything+
- "you know that from school"
- painstaking detail of the easier precursor material but rushing through the harder material it was the point of the lecture to master ++
-Discuss material too quickly and leave no room for questions+

- Giving not enough time for the exercieses
- The feeling that 'I would rather not be here'
- belittling relevant experience or opinions of the students
- "that's trivial!"
- got tired
- only participated for the certificate, so didn't even start motivated in the first place
I had a bad internet connection in python course...
Too dificult content, dont get the big picture+
- Overtime++
combined: very motivated and going fast - can be demotivating for some students as the teacher is going faster than they can proceed, whereas other students who process information fast can be hyped about it

Exercise: Helping Learners Learn From Mistakes
A learner at your workshop asks for your help with an exercise and shows you their attempt at solving it. You see they've made an error that shows they misunderstand something fundamental about the lesson (for example, in the shell lesson, they forgot to put a space between ls and the name of the directory they are looking at). What would you say to the learner?

In the Etherpad, describe the error your learner has made and how you would respond.

Room1:
showing them the mistake, or in the course forum giving a more generic answer how to deal with such a problem while solving a similar mistake

Room2:
Error: The student has forgotten to declarate a variable name
Respons: Check the errormessage and have a closer look at this variable name; --> he/she should find the solution by himself/herself

Room 3:
Look at the error message and explain what it is telling you, this way they can recognize it the next time it happens. Remember to tell that errors are normal and happen, and try to explain what the computer thought you were trying to do.

Room 4:
positive feedback: working in the right direction, almost there
ask questions what 'lsfoo' is supposed to mean?

Room 5:
-Ask student to explain step by what she/he has done and why. Line by line.
-explain what components there are. e.g. in shell example: how does computer know what the components are+
-

Room 6:
- read the error message with the person -> interpret with them
- Let the person explain what he/she did
- experience: often while explaining they notice that they did not understand something or they find their error themselves

Exercise: Discuss The Carpentries Core Values

Take a moment to read through the Core Values on this page: https://carpentries.org/values/

Choose one core value that resonates with you. What is a decision you might make in a workshop that could look different if you were actively considering the core value you chose?

This exercise should take about 5 minutes.

Always learning - It is my job to teach the sylabus, but often students give excellent examples or new input, therefore I too learn while teaching. Or they ask questions I have not considered my self, so the challenge is beeing open for the student input. ++++1++++++
Empower One another - I learnt a lot/was empowered from the tech community and would happy to contribute and empower others+++
access for all: one time I used a colored illustration and then was told by a student that he was color-blind. I could have prepared the illustration more accessible ++++++
access for all: Consider different types of workshops (i.e. spread over multiple days) that would be inclusive for people with caring committements/disabilities. + Think about multiple breaks. + potentially make material available before sessions for learners that require it.++
Empower One Another++++
Someone tells me privately that he felt discriminated by something I said. I act openly if I talk about this in the main room (without mentioning the name), apologizing and encouraging others to speak up as well. +
strength through diversity – valuing different perspectives is a topic that is a bit hard to tackle in the "technical world" as there's something like wrong&right; still we could value when someone makes a suggestion which actually IS wrong, but contains interesting aspects+++
"value all contributions" is certainly something I would need to work on, maybe by teaching more people that are not in my responsibility anyway, reduce cynicism+++
Act openly++++++++++
Always learning - Someone tells me they are just too stupid to code -- it is a matter of practice and getting the grasp of the thought process - they can do it!+++
strength though diversity: we have different point of views and different requirements++

act openly - I think it is not allways easy to do that if you feel insecure, e.g. if something goes wrong (an error is showing up) -> taking this as a learning oportunity I think is not easy but very valuable!+++
new ways of explaining something when I see that I am not reaching everyone+

act openly: admit if one has no answer for a question raised and try to find an answer after the workshop and spread it around later++

inclusive for all -- creating a safe enviroment for people to express themselves and their identity+++

Accessibility

Exercise: What Happens When Accessibility is an Issue?
Think of a time when you have been affected by, or noticed someone else being affected by barriers to accessibility. This may have been at a conference you attended where the elevator was out of service, or maybe a class you were taking relied on audio delivery of content. Describe what happened, how it impacted your (or someone else's) ability to be involved and what could have been done to provide better accessibility in this case.

This exercise should take about 5 minutes.

My study cirrucla was only in german, my scientific job, at the same university is only in english. So I am still learning the vocabulary
In my old job, important details where explained in german, but some people barly spoke german.
There are some old buildings in my university where the toilet is not accessible to wheelchair users. My friend had troubles.
A lecturer in an online course, switching back and forth between the current task and other parts, eg the answers already given.
Not directly affecting a person, but during a fire safety briefing, wheelchair users were forgotten as possible people affected. The person in charge was surprised, but then revised it :-)
Mainly language problems, often with students who just arrived from (Asian) countries. Should have asked a colleague to help as an interpreter.
I asked a colleaque to proofread an text, which I had marked in green and red and he couldn't read ist, becaus of his colour-blindness. I started to think about these barriers.
A conference I attended did not have any seats and quiet areas (for rest or sitting while eating). A collegoue who has a disability for which they require quiet breaks of rest was not able to participate in the full conference. More breaks and a quiet space would have enabled them to stay for the full day. +
a teacher once exposed a student by saying that all the learner said was more or less bullshit because they had some wrong/broken concepts in their mind

Lecture notes typically much more difficult to access/follow for dyslexic learners. Also difficult for dyslexic instructors to prepare text handouts in the most commonly used methods
Lecturers in German-speaking countries routinly use slides in English while speaking German, which makes it very difficult to understand when neither is your native tongue+
i prepared a presentation on my linux notebook but due to university infrastructure based on microsoft, i was not able to get my presentation running

Invitations to workshops sometimes come only in german; I often translate them and send it to our people speaking only english.

Exercise: Giving Feedback
We will start by observing some examples of teaching and providing some feedback.

Watch this example teaching video as a group and then give feedback on it.
https://www.youtube.com/watch?v=-ApVt04rB4U Put your feedback in the Etherpad. Organize your feedback along two axes: positive vs. opportunities for growth (sometimes called "negative") and content (what was said) vs. presentation (how it was said).

Note: there is a version of this video with subtitles in both Spanish and English here:
https://www.youtube.com/watch?v=jxgMVwQamO0

This exercise should take about 10 minutes.
Positive
- Content

- pointing out benefits of the lecture +
- showed how to fix a mistake
- Recap of what you did in the last session+
- to the point
- makes a mistake live and corrects it
- repeating the last point from before the break++
- starting very easy, then extending the example
- showing easy examples first

- Presentation

- loud and clear voice +++
- slef-confident +
- Give a big pciture, why are learning the following content
- asks for feedback
- keep doing lifecoding!+
- unafraid to make a mistake +
- active/lively teaching style, using guestures and modulating voice
- do a short recap after the break is over++
- speaks freely
- confident and informal+
- admits errors+++

Growth opportunities
- Content

- explaining the concepts you teach before you use it +++
- Explain to students what the learning outcomes is of the session before diving into the topics, this will help motivate and provide context to the concepts you are teaching
- unnecessary and extraneous information+
- examples should be clear before the presentations
- it would be great to have a bit detail on the summary of last time's lesson to make it easier to remember and get back on track
- surprising step towards functor, explain why
- doesn't explain how the fix was done++++
- No need to mention that excel users can do this too
- Saying “this is really simple” and “even Excel users can understand it” can de-motivate students
- Use of jargon - terms such as “polymorphic” or “lexical binding?” +
- if you explain basic stuff like function please do not use words the students cannot understand at this point like "lexical binding" or "polymorphic"

- Presentation

- larger script for better readability++++++++++++
- skipping things like "it's really easy" might be more inclusive++++

- Using your smartphone during your lecture+++
- Asking for sit down was a little bit angry; shaking the head a little bit
- tone - it's crucial to be polite to the students
- seems to get stressed when something doesn't work, could just laugh it off
- Enage with audience: check in with students to see if they are following along
- dismissive language, aggressive+ ("Sit down now" at the beginning did not sound very friendly, also 'even Excel users can do this')
- please try to relax, it makes me nervous that you are nervous
- it was too dark in the room+
- focus on your audience (no turning the back to the audience, check for long faces)
- you can follow the content better if you speak more slowly
- verbalise the steps and tips apart from typing
- please take more time to explain details instead of telling "trust me on this"++
- in the beginning reacts very authoritarian towards a student - not very nice!; looks at whiteboard and shows there; font way to small to see something; non-inclusive language ("simple stuff", "easy to understand", "trust me");
- Looking at phone while teaching: best to switch it off++
- Turning back on the audience and does not look at participants much (teacher was mostly looking at the screen)
- "Any questions?" Doesn't encourage asking any
- "don't worry"

Exercise: Sharing Feedback

The prep time for this exercise is intentionally short – the point is to practice giving and receiving feedback, not to create a perfect presentation. Imperfect presentations will give you more to work with!

Trainings where trainees are co-located:

Split into groups of three.

Individually, spend 5 minutes preparing a 90-second introduction to the topic of the lesson episode you chose before the start of the training course. You will not be live coding; you can use a whiteboard or other visual aids if available (but this is not required!).

Get together with your group and have each person teach their segment to the group, while one person records this (video and audio) using a cell phone or some other handheld device. Keep a strict time limit of 90 seconds per person (one person should be responsible for the timekeeping).

After the first person finishes, rotate roles (they become the videographer, the audience becomes the instructor, the person who was recording becomes the audience) and then rotate roles again.

After everyone in the group of three has finished teaching, watch the videos as a group. Everyone gives feedback on all three videos, i.e., people give feedback on themselves as well as on others. Keep an eye on the time during feedback, especially if your group has more than 3 people, to be sure to leave time for everyone.

After everyone has given feedback on all of the videos, return to the main group and put everyone’s feedback about you into the Etherpad.

Distributed trainings: Your Trainer will split the group into virtual break-out rooms. Follow the instructions above but do not record each other. Instead, give each person feedback immediately after

they finish their turn teaching.

This exercise should take about 25 minutes.

Room 1:
Feedback - good speed and clear structure
Feedback - negative used the word markup without explaining what it is

Content
positive - explaning terms that were used first
live coding,
Presentation
positive - good, clear structure

- - confindent

Feedback: good start with a quick summary. poor audio quality
explain content first
+ made presentation feel positive
+ good tempo
+ confident presenter
- Why am I taking this course? (motivation)
- Did not verify attendees could access the shell

Room 2
-

Room 6
Feedback - Positive: Explain background + purpose of excerise, explain why it is useful, read out what a command stands for
Feedback - Negative: Careful of support words like "mmmm", take more time to show commands on screen

Room 7
Feedback: Positive: Life Coding Example, on the whole clear what was done. Negative: type slower, better explain what can be seen on the screen
Feedback: Positive: Clear voice; Enthusiams; Negative: Felt rushed (mainly due to 90 seconds rule); the Powerpoint slide contained some information that was not discussed

Feedback: clear voice, no rush; but no direction, where the presentation should go to. for me the feedback was good and right. and I am used to get feedback

Room 4 (feedback has been verbally delivered)
- improve: don't say that a library (glob) has a weird name, or at least explain where it is coming from+

- take care of large enough font size when screen sharing

- positive: pace, pleasant tone, profesional handling of audio problems, clear instructions were given, introduction with outlook what is going to be learned+

Exercise: Using Feedback

Look back at the feedback you received on your teaching. How do you feel about this feedback? Is it fair and reasonable? Do you agree with it?

Identify at least one specific change you will make to your teaching based on this feedback. Describe your change in the Etherpad.

This exercise should take about 5 minutes.
happy with the feedback - clear communication of what to keep doing and what can be improved. I would explain if there are many windows to be opened autamtically - so that the learners don't get confused with pop ups; be better prepared next time for screen sharing :)
something was honoured that I didn't do on purpose; now that I know it helps, I'll intentionally include it in my future teaching+
I was aware of the feedback, and know that it is a result of not being prepared enough. I will make an effort to slowdown and practice before teaching this topic.
I found that I had the urge to explain myself when receiving constructive feedback. It was reasonable and I didn't feel offended. +
I will take more time when typing / explaining the output on screen, not assuming that everyone knows what is happening
I have to critically review my materials and try to take an outside perspective in order to identify all concepts I just assume without properly explaining them
Fully agree with all the feedback that was given ; I shall make sure to talk slowly even when pressurised to deliver the message within 90 seconds
I am happy with the feedback, thank you guys!++++++
Choose perhaps only one aspect of a topic instead of explaining it in all its complexity
I was happy with the feedback I got, but found it difficult to really condense something down into just 90 seconds. +++++++
Feedback is absolutely fair, put time into improving language
Very pleased with the positive feedback I've received :)
My English is really bad :-(

Got only one short feedback because time was over, but this one was only positiv on my presentation. double check accessibility before presentation, e.g., audio, visual, etc.++++

Homework:
https://data-lessons.github.io/instructor-training/12-homework/index.html

WIth no coding
https://librarycarpentry.org/lc-open-refine/

Minute cards Day 1: https://forms.gle/c6KKUCp2gK9nXb3c8