ASPP 2022

MONDRAGON UNIBERTSITATEA

14th Advanced Scientific Programming in Python

a Summer School by the ASPP faculty and
the Faculty of Engineering of the Mondragon University, Bilbao

5–10 September, 2022. Bilbao, Spain

Evaluation Survey Results

Method

The survey has been administered with a web interface created with the LimeSurvey software available at: http://www.limesurvey.org
All answers have been submitted by 3 October, 2022.
No answer was mandatory.
The free-text answers have not been edited and are presented in their original form, including typos.

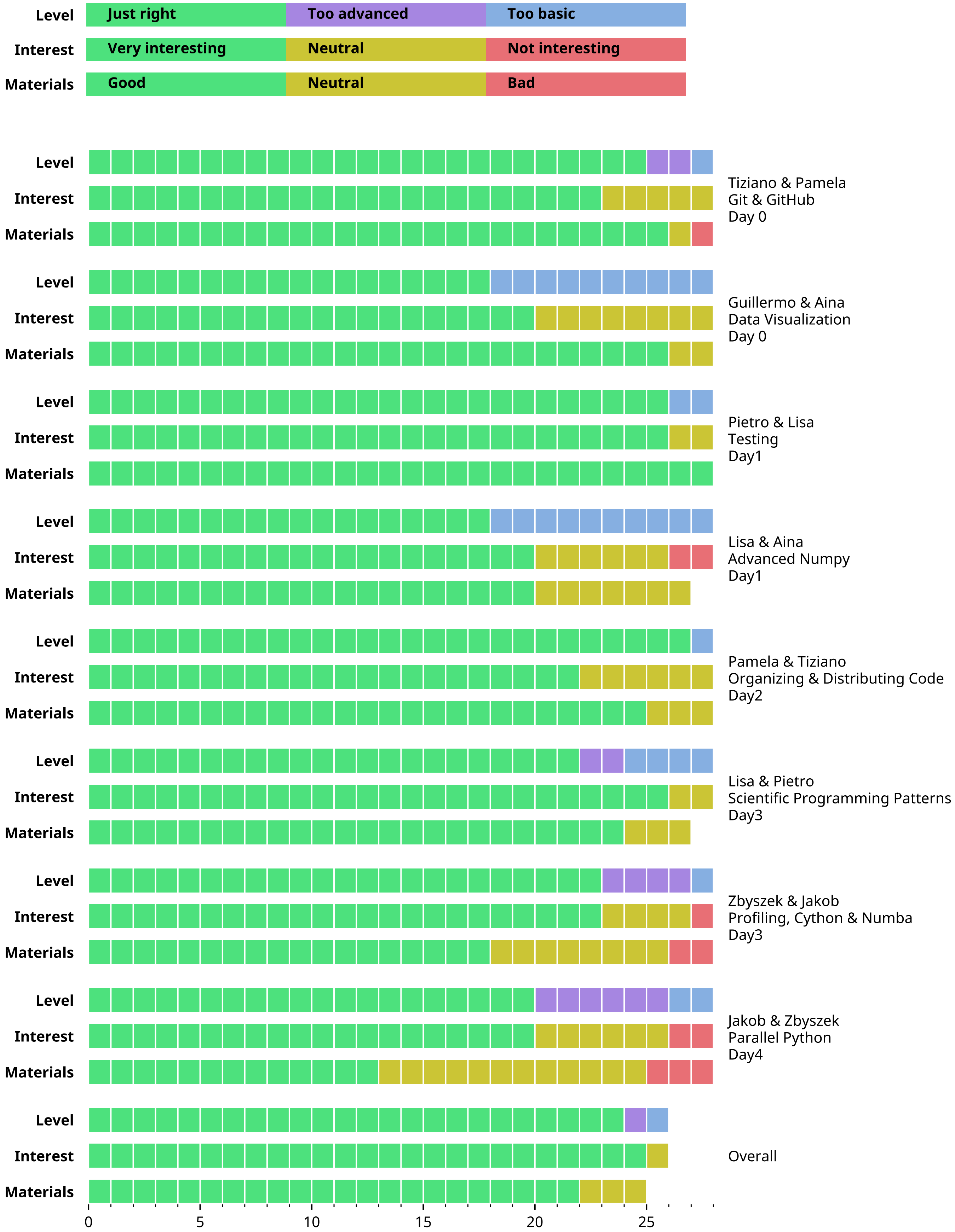
Attendants and Applicants Statistics

	Attendants		Applicants	
	28	17%	163	
Different nationalities	20		45	
Countries of affiliation	11		27	
Gender: other	0	0%	1	1%
Gender: female	16	57%	71	44%
Gender: male	12	43%	91	55%
Already applied	7	25%	14	9%
Bachelor Student	0	0%	3	2%
Master Student	0	0%	30	18%
PhD Students	19	68%	100	61%
Post-Docs	6	21%	19	12%
Professor	2	7%	3	2%
Technician	0	0%	1	1%
Employee	1	4%	6	4%
Others	0	0%	1	2%
Completed surveys	28	100%		

More stats about attendants are available at: https://aspp.school/2022-bilbao/students.html

Lectures & Exercises

Q: *Grade the* ***level*** *of the lectures*
Q: *Grade how* ***interesting*** *were the lectures*
Q: *Grade the quality of the presentation style and/or of the teaching* ***material,*** *e.g. the clarity of the slides/code, the exercises and the solutions, etc.*



Q: Are some of the topics presented in the lectures not relevant for a programming scientist?

1. I thought some of the optimization lecture was a bit too in the weeds on CPU architectures and the like to be truly relevant.
2. Parallel python didn’t seem to relevant for me
3. Optimization by python, numba and parallel python seems something that you will do only when the code is ready to use by others as a software. To end up developing a software during a scientific career is rare, although not impossible (example, ImageJ by Albert Cardona). Therefore, these topics might be only relevant for some scientist and not something that can be generalized. Nevertheless, I enjoyed them very much!
4. I think all the topics were relevant for a programming scientist. The only session I thought was a bit basic was the visualisation one. I think at the coding stage everyone came, we have all had experiences visualising data. So, either “hacks” could be presented or things like interactive visualisations like Dash.
5. All topics are relevant. However, of the topics presented, I'd say the numba/cython/parallel python is arguably less useful. I might have spent more time on topics like programming patters, advanced numpy, and organizing/distributing code.
6. I'd surprisingly say that testing didn't feel much relevant, but it's probably only the matter of taste. It feels too complicated to set up, especially if your code gets larger. Although I don't know a better solution :)

   I'd also enjoy to be told some philosophy about in-code `assert` statements
7. Even though it was interesting, I would probably have skipped the data visualisation class and had more time to digest the other topics.
8. Everything was awesome! :) Please keep updating and developing this format in the future years, this has been the most interesting and relevant summer school I've ever been to.
9. Setting up parallel python on the server was too complex to be really useful within the scope of the lecture. Everything else was just perfect!
10. I think advanced numpy might only apply to certain kinds of data sets, but I am still happy I learned more about it.

Q: Are there further topics relevant to the programming scientist that could have been presented, given that the total time is limited. Please also mention which topics should be replaced by the new ones.

1. The topic that I was most interested in was the data visualization section. That has been the lesson which I have taken the most from in my day to day coding, but it didn't get very much time during the course and there were even fewer exercises.
2. Scientific data analysis - big data. --- it is hard to remove any of the current topics, but maybe the parallel python would be the best to replace
3. Working with Pandas
4. Although visualizing data is very important, doing the right statistical analysis and showing the plot is much more relevant in science. I would suggest adding that aspect to the current data visualization day and altogether making it a bit more complex.
5. It might have been useful to have an introduction to any particularly good stats packages etc. I know lots of people do stats in python, but I don't know of any especially comprehensive and or user friendly packages.
6. I could suggest "how to make use of pandas for your data" -- although it has a rather messy api, I've seen a few areas making use of pd.DataFrame object for their particular kind of data (rdkit, reciprocalspaceship). I'd love to hear some philosophy behind the general approach of it, honestly
7. I thought the advanced numpy lecture and data visualization could perhaps be combined in a way that you have to perform some calculations with numpy before visualizing them and spend more time on topics that were (for me) were more interesting and novel (perhaps the Testing, Organizing & Distribution of code or Scientific Programming Patterns)
8. Presenting more scientific libraries (for statistics, data frames, modelling), for me profiling was an excellent topic, but Cython or Numba is not really applicable
9. No, I think the number of different topics was just great!
10. I would mention the pandas library but it's relatively easy to use when comparing to presented topics. So no need to replace anything.
11. It would have been nice to spend more time on Organizing & Distributing Code, but I am not sure which topic should be replaced (they were all fantastic!)
12. I wish to humbly suggest the addition of atmospheric satellite remote sensing data. Parallel group project work and exercises, with some topics on big data from the atmospheric satellI hite remote sensing data

13. I know time is limited and many choices depend also on the field and therefore the kind of data we're handling, but I will mention a couple of aspects that often come up as bottlenecks in my experience:

    * memory limits, I/O operations from remote servers or slow storage in general, handling big files in an efficient way and possibly parallelizing the operations (how to use queues, stacks etc). This is also with reference to the profiling and optimization lesson, which was very interesting but I almost never need to optimize numerical aspects as there are plenty of libraries that often do that pretty well.

    * processing data streams

    * more advanced OOP operations such as inheritance, polymorphism... it's often the case for me that I need to define a class that inherits from python standard dictionary for instance, or other basic data structures, in order to make them work for my needs

    * some functional approaches, e.g. the functools library
14. Replace Advanced Numpy with Data Analysis using Pandas and SciPy
15. Profiling for memory usage and how to write more efficient code for working with large datasets would be great to add. I would suggest removing the data viz lecture because most of the things we learned can be acquired through the Matplotlib website and the problems for data viz can be quite individual depending on the complexity of the dataset. The Numpy lecture was a little bit basic, but quite relevant for understanding memory usage and computation efficiency.
16. Pandas and dask
17. I think I would have benefited from a brief lesson on plotting data in general before the rest of the data visualization lecture / exercises. In addition, it would be great to have learned more statistics, but I realize this topic could be a whole summer course on it's own.
18. I would have loved to have gotten further in the Data Visualization lectures, but having the materials online after makes it an excellent reference to have. I could say that of the whole course, that having the materials to reference back in my own forks. I loved the classes lectures though I think maybe a more basic exercise of just making a basic class with a few attributes and playing with that workspace a bit longer would have helped me better understand there before moving on to the class methods. I don't think any topics should be replaced but I would love to see what a second year curriculum of the course would look like of the next steps! Everything taught was super useful. If there's one thing to make space then I suppose potentially the python parallel cluster bit could be taken out. It's very cool, and I loved that bit, but, I think it's something I'm unlikely to implement especially as most scientists already have access to a pretty big cluster. Still it was very cool and I loved it.

Q: Do you think that pair-programming during the exercises was useful?

Yes, I have learned from my partner / I have helped my partner	100% (28)
No, it was a waste of time for both me and my partner	0% (0)
Neutral. It was OK, but I could have worked by myself as well.	0% (0)
Other	0% (0)

Q: What do you think of the balance between lectures and exercises? When answering, please keep in mind that the overall time is limited ;-)

Lectures were too long, there should be more time for exercises	14% (4)
Lectures were too short, there should be more time for lectures	7% (2)
The time dedicated to lectures and exercises was well balanced	71% (20)
Other	7% (2)

Other:

1. Overall unbalanced
2. There were a few moments when time for exercises was not enough. This had to do with questions from students that dragged on, so perhaps the time for questions should be balanced more and personal questions can be answered instead.

Q: Any further comments about the lectures and exercises?

1. The lectures were well prepared and presented enthusiastically, which made it easy to listen to and understand. It was were very useful to immediately apply the learned material in the exercises.
2. The lectures and the tutorials are not really separate, but are really closely tied together, which is great.
3. While I understand that time was limited, have the feeling that some presentations were rushed. The balance between lecturers and exercises was good though.
4. The content of most of the lectures was too ambitious so most classes didn't finish the presentation and exercises were proposed as homework.

   Some exercises should be a bit more guided so that students don't 'waste' that much time.
5. I just want to thank you for all your efforts! The summer school was excellent and I appreciated the knowledge of the tutors and their friendliness. The learning atmosphere was great, the cohort was diverse and we learned a lot from one another. I can also see how much effort has also gone into designing the repos for each topic, their exercises and of course, Pelita! I am happy that we were able to fork the repos to have access to them in the future in case we need to remember some topics.

   Thanks for a wonderful experience!
6. It was the first time I experienced pair programming and I just loved it! It is a great experience both for learning new skills and also for learning my own reactions to a new situation/interaction. Clearly, this was not the first rodeo for most of the lecturers! They have really found the perfect balance of topics for this event.
7. I was amazed how well picked the topics of the lectures were. I didn't find a single lecture or excercise irrelevant for my PhD. Not only the topics were well picked, but also the implementation into the course was super nicely made and there was always a good mix of lectures and excersices in which we applied our freshly gained knowledge.
8. Great job, I couldn't imagine a better course, thank you all!
9. I think longer excercise slots and shorter lectures would be good for the following reason.

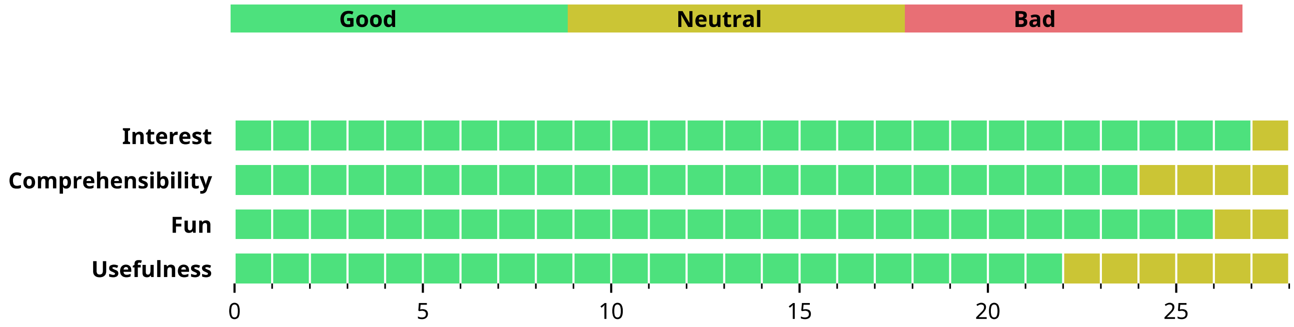
   I was a few times in the situation of being paired with someone with less coding experience who, when the time for the excercises came, had not understood the concepts just described enough to immediately start with applying them. In these cases it would be nice if there was enough time both for the less experienced person to understand and for the more experienced coder to have an opportunity to teach and explain.

   I tried both going slow enough for the other person to follow everything (at the cost of being caught up to the lecture) and trying to explain while I completed the excercises in the allotted time (at the cost of the understanding of the other person.
10. The time dedicated to the lectures and exercises was well balanced, but I do wish there was more time overall! It was an amazing course and I am grateful that everything is available online.
11. I humbly suggest the inclusion analysis of atmospheric data from satellite as ASPP is a world hub of python programming so data analysis from other science domains apart from neurosciences could be considered in future series of ASPP
12. I loved the approach. However, since programming needs practice, and practice means stumbling on difficult problems and trying to solve errors that come up in the most absurd situations, I would have really made the most of the summer school if, for instance, we also had some time to work on (hopefully enough challenging) exercises on our own. The ideal situation for me would have been to have lessons in the morning and time for studying or just solving problems in the afternoon. I know this means doubling the length of the summer school but it would also mean a lot more confidence on the acquired skills at the end of the two weeks and not just a lot of knowledge to handle on my own later on.
13. ASPP manages to strike the right balance between lectures and exercises. I found all lectures to be intellectually stimulating and highly relevant to my work. I was able to easily ask questions and benefit from the knowledge of the instructors.
14. This course was incredibly helpful. All lectures and exercises were well designed not only for learning efficiently, but also for enjoying every step along the way.
15. This program exceeded my expectations! I can't think of a better way to learn these topics than the programming project that we did. In addition, this is a truly amazing group of instructors to learn from. A huge thank you to the organizers!
16. The course was excellent and filled a gap in my knowledge that certainly will help my research moving forward. Learning to properly use git, class objects, and how to organise and distribute code is a game changer for me that I feel will let me not only write better code, but do better and more reproducible code. All of the exercises were highly relevant to the content of the lectures, and through the combination of lecture and practical exercises with peer programming I feel that my time in the course greatly expanded my programming capacity. I thoroughly enjoyed this course, and take from it not only new friends and colleagues, but an incredible formative experience that ensures I'll understand and that my code runs years after it's written.

Programming Project

Q: Evaluate the programming project.

Interest: How interesting was the programming project?
Comprehensibility: How clear and comprehensible was the code and the available documentation? Was it easy to work on the programming project
Fun: Was it fun to work on the programming project?
Usefulness: Was it useful to work on the programming project? Do you think you may re-use what you learned?



Q: Do you think the team-programming experience is relevant to your work as a programming scientist?

Yes: 86% (24)
No: 14% (4)

Q: Do you think that the project should be about a real-world scientific problem instead of a video game?

Yes: 11% (3)
No: 89% (24)

Q: Any further comments about the programming project?

1. The project was extremely fun to do. As many of the topics during the course, it was relevant to the tasks of a scientist due to the required time-management, teamwork, and communication skills. The video game was useful to see that a completely new topic could be understood and improved upon quite fast.
2. Fun programming project at the end, which still helps with practicing some of the skills learned earlier in the week, especially git.
3. Programming Pelita bots was one of the things I liked most about the Summer School. It gave us the opportunity to bring together everything we had learnt - from Git to testing and to a bit of optimising code! The base code is well documented such that it was easy to understand all the objects and variables enough to meaningfully programme the bots. Above all, working as a team on the project gave us a feel of industrial programming paradigm where teams are the norm. I liked the fact that it wasn't just the programming, but also strategy, team work and ability to meet deadlines.
4. If anything, I would present the project one or two days before so that students have more time to think about it and leave the last two hours of each day for team work.
5. Using and programming Pelita was very fun and we got to apply the topics we learned. I enjoyed the fact that it was a game rather than a real-world scientific problem, since it was a good way to keep us motivated to do well in the live competition- not to mention getting an awesome hat if we won. We were able to build complexity in our programming of Pelita if we so chose, which reflects that the exercise is educational enough.
6. The programming project was really fun and engaging! I wish we had had more time to develop some of the features that were left out due to the approaching deadline, though. Overall it was an interesting, new way of working in teams that is very rare in science.
7. Project is super fun and surprisingly useful as an excercise. My expectations from it were much lower, but then I was amazed how nicely we can apply git to work together as a team, and it really worked -- and the final competition was fun too :)
8. I think the programming project was excellent as it taught us how to program as a group, co-ordinate the code together and using git and GitHub in a proper way which I think is very good skills to have even if they're not so relevant to me at this moment for my work.
9. I think that working on a topic completely detached from our own research is the key to success. In this way, we did not have to rely on pre-existing scientific knowledge and just enjoy the pure programming environment. And it was super fun too! I would have liked to have a bit more time to really understand all the code we used in the project, but there are clearly time constraints. Thumbs up!!

10. It was so fun!!!! Pelita best game ever :)
11. The pelita game is fun and fits well at the end of the intensive course!
12. I humbly suggest the inclusion analysis of atmospheric data from satellite as ASPP is a world hub of python programming so data analysis from other science domains apart from neurosciences could be considered in future series of ASPP

    I wish to humbly suggest the addition of atmospheric satellite remote sensing data. Parallel group project work and exercises, with some topics on big data from the atmospheric satelIite remote sensing data
13. The project was super cool and I can't wait to code some more bots on my own! I realized how I have issues on programming with other people, as I'm just not used to do that. It would be super cool if the project part also included some software engineering paradigms to implement (i.e., how do we agree on interfaces? how do we split the work? What are best practices? I assume this is something that can be learned as well and it's in some sense part of an "advanced programming" course)
14. The project was a lot of fun and working in a team was both invigorating and challenging. The real world application of the game is limited in my field but the programming and soft skills are invaluable.
15. Pelita rocks
16. It was a perfect combination of learning, team experiences and a lot of fun.
17. I think a video game programming project is a good way to bring together scientists from different backgrounds.
18. The programming project was the perfect balance of fun and challenge. I think it required us to work together using version control and shared repositories, to develop a an organizational logic for our different functions, and to strategize our game plan as a team. These skills are highly relevant to my field and I'm really proud of how far I have come in one week and how much we got done as a group!
19. The programming project was very fun, and I feel it was great to have something so neutral to everyone as no team had an advantage based on domain knowledge. It allowed me to use git on a practical level, and run continuous integration in a way that I hadn't to this point, and I learned a lot from it. The game itself was a blast, though I do think the API could have been slightly more explained before we got started. I feel that had we done this in conjunction with the class lectures as well it could have let us put into practice the conceptually most difficult part of the course. I think that working in a team this way was also great as it showed me a new way of working, and how we can program together as opposed to alone to make a project bigger than ourselves.

The School in General

Q: How do you overall evaluate the school?

Good: 100% (28)
Neutral: 0% (0)
Bad: 0% (0)

Q: How do you evaluate the general level of the school? Was it too advanced/too basic with respect to your expectations?

Too advanced: 7% (2)
Just Right: 89% (25)
Too basic: 4% (1)

Q: How do you evaluate the general level of the school? Was it too advanced/too basic with respect to what was advertised in the announcement?

Too advanced: 4% (1)
Just Right: 96% (27)
Too basic: 0% (0)

Q: Did you learn more from attending the school than you would have learned from reading books and online tutorials alone?

Yes: 96% (26)
No: 4% (1)

Q: How do you evaluate social interactions and social activities at the school?

Good: 96% (27)
Neutral: 4% (1)
Bad: 0% (0)

Q: Would you recommend this course to other students and colleagues?

Yes: 100% (28)
No: 0% (0)

Q: How did you hear about the school?

Google Search: 2
Professor/Tutor/Supervisor: 5
Colleague/Friend: 12
Mailing list: 10
Other: 3

1. I joined the ASPP mailing list after meeting some of the organisers in another event they organised
2. somewhere on twitter I believe
3. from a colleague at SciPy who distributed the invite to our data science club

Q: Any further comments or suggestions?

1. ASPP was an amazing experience. The lectures were interesting and engaging, so that the time spent learning passed very fast. Nevertheless, now that I am back I am very applying most of the learned topics confidently to my own work. The organisers and the other students were all amazing people that created an enjoyable atmosphere inside and outside of the classroom.
2. I found, that I benefited most from the topics where I was already experienced, but not yet expert, which is great. Maybe the best aspect of the summer school was the committed faculty, and that all of them stayed for the whole week, not just when they teach.
3. I understand that given that the history of the Summer School and its links to neuroscience, word about the Summer School easily spreads to the Neuroscience community. However, I think it would be good to have more diversity in the students' background.

4. I really enjoyed ASPP and learned a lot! Thanks so much for all your efforts!

   The only suggestion I have is for the lunch break to be planned in advance, because the first day was a bit of a scramble to find where to go with enough capacity. But thanks so much to Aitor, because after realising this, he and the rest of the team swiftly made sure that every other break was well-organised!

   Also, small thing for Spaniards in general (not ASPP organisers), but I think they need to deal better with dietary restrictions. ;)

5. I thought the course was fantastic! I learned so many useful tools that I wish I had known at the beginning of my PhD and that I know want to implement. I would strongly recommend this course to anyone who's starting with python and especially if they're starting a new project when setting up the design is easier.

6. This was a great event! Clearly, the lecturers were there because they love what they are teaching: this was obvious from their first sentence on day 1. Moreover, they managed to create a super relaxed and user-friendly atmosphere that contributed to the success of the school. The other students were also super nice but also knowledgeable: I've learned so much from them too. Finally, staying all at the same location made social interactions way easier! Thank you so much for doing this!!

7. Thanks to all the organizers for this wonderful week! I learned so so much and got to know so many awesome people. It was a way too amazing experience to find the rights words! <3

8. Best course and atmosphere ever!

9. This was an amazing course, thank you for the opportunity! I left ASPP with renewed enthusiasm, new skills, and treasured memories. Thank you to all of the organizers :-)

10. I humbly suggest the inclusion analysis of atmospheric data from satellite as ASPP is a world hub of python programming so data analysis from other science domains apart from neurosciences could be considered in future series of ASPP

    I wish to humbly suggest the addition of atmospheric satellite remote sensing data. Parallel group project work and exercises, with some topics on big data from the atmospheric satellite remote sensing data

11. I loved the school, for real. I've probably written it in every single text area and I'll do it again here.

    Time-wise, probably the lunch break was too long considered that we managed to have lunch inside the building almost always, and shortening it would have saved us a little time at night. The last lecture of the day was probably a bit too long, given the complexity of the topics.

    It's amazing how everyone was just the right person in the right place, and this is true for both faculty and students.

    I would love to see the project evolve in very many ways, as I believe that the topics that are part of the school are highly needed both inside and outside academia. It would be super useful to have it as a permanent seminar for instance, longer and more diluted than a summer school, or even a proper university level program.

    I'd like to remark how the "open source" approach was a huge feature of the experience and this, in my opinion, should be stressed as a crucial aspect and strength of the school. And with "open source" approach I'm referring to the choice of technologies but also to the general sharing philosophy that every single participant and organizer seemed to embrace. It's hard to explain, I know you'll get what I mean.

    I'd definitely enroll in (and pay for) a program that teaches me how to properly and efficiently do scientific programming, with open source approaches, both inside and outside academia. Having someone in the lab that can really code and understands the scientific approach at the same time, hired precisely to do that, is a need that has come up in probably every single conversation I had with my colleagues over the years: this kind of "lab technician" does not exist in my field, but at the same time there often is the "hacky" person that, alongside their research work, helps everyone with coding and maintaining code and managing the server issues etc. It would be amazing to have a proper training to be that person.

    For the first time in years, I felt the willingness to be involved and help out and be kept in the loop. I feel this is a great sign that something special has happened during the summer school week ;)

12. ASPP was an invigorating experience this year! It feels like someone crawled into my brain and figured out exactly what I needed to become a more efficient scientist and programmer. The programme has a wide appeal to both scientists in industry and academia. It is not an opportunity to be missed!

13. These instructors are incredible! I can't believe how much I learned in one week. Each of these skills is a huge benefit to my ability to conduct and analyze my research data. I highly recommend this program to anyone who is attempting to learn more about scientific programming.

14. I absolutely loved it. Keep doing a great job and hopefully I can be involved later at some point. It was such an amazing group of people and learning experience that you can't help but want to come back as a tutor. Thanks so much to all of the organisers, it's a phenomenal course.