Reflection on pynblint

1. What did you learn, in your individual session, about static analysis for ML and the pynblint tool?

   I had no idea that this kind of tools exist. I learnt that pynblint is important tool to make your notebook more readable, comprehensive and share worthy. It helps to achieve real purpose of creating a notebook.

2. Will pynblint be useful to you in your WASP PhD project? Why or why not?

   Yes, I think it can be useful in my research because I have plan to publish my code with publications. Therefore, pynblint can play a handy role to make my notebooks more collaborative, reproducible, and trustworthy.

3. Ideas for how the tool could be improved.

   At the moment, pynblint only analyses the style of coding (more towards the format of code) but it does not say much about coding itself. For instance, It does not rectify the code itself (like duplication of code, dead code etc.). If it adds these features in future then it can become more useful.

4. What do you see as the limits for static analysis tools in ML? For code, models, and for data?

   These does not execute the code and therefore it does not know about the code, data, or models. That is what I think is the limitation of static analysis.