Testing for Data Science

As a data science enthusiast, I thought it would be interesting to discuss how we can apply software testing rules while developing data-focused applications.

I would like to answer my own questions with you guys. How do the following things look like for data science?

Q1: TDD ?
My take: TDD can be used to confirm if our input data matches our expectations - for eg. the kind of dataframe you are expecting while writing a PySpark code by checking for number of rows or types of columns. This becomes important because, when doing a 'transformation' not necessarily means executing an 'action', and you cannot be sure about what structure your current RDD (data) is in.

Q2: Unit Testing?
My take: I have always used python for data analysis. Python has a library for testing called 'unittest'. It looks similar to JUnit and has assert statements. Testing basic statistical hypothesis, like maybe p-value could come under unit test.

Q3: SetUp and TearDown?
My take: If you are writing a spark application, then setUp could include setting up an application, even if you are replicating a cluster on a local machine. TearDown would include stopping the spark process. This becomes important when you your test cases have the same object/dataframe names. If not instantiated each time in each test case, your application might end up accessing older versions of dataframes and give wrong results. I have personally wasted alot of time trying to figure out why my model is giving an error, only to find out my dataframe had changed.

Q4: Test automation?
My take: If you want to write a set of automated tests, which will run each time before deployment, you might want to write tests for checking if the new model has error rate under a certain limit, or has model accuracy falling within a certain range. Accuracy being too low can mean that the new model was not good enough, accuracy being too high can either mean you are really bad-ass, or that you have made really stupid mistakes with respect to processing your data.

Do you guys have any different answers?

References:
http://engineering.pivotal.io/post/test-driven-development-for-data-science/
http://www.predictiveanalyticsworld.com/patimes/four-ways-data-science-goes-wrong-and-how-test-driven-data-analysis-can-help/6947/
https://blog.dominodatalab.com/unit-testing-data-science/
https://www.packtpub.com/mapt/book/big_data_and_business_intelligence/9781786465160/13/ch13lvl1sec78/unit-testing