Bio – INSERT HERE – preferably with picture (otherwise I will choose one ☺

Bojan is a Machine Learning Modeler at NVIDIA. He has been working in Machine Learning and Data Science fields for seven years and has experience with real-world FinTech problems. He is a Quadruple Kaggle Grandmaster and is the first person to be ranked in the top 10 in all four Kaggle categories simultaneously.

A person smiling for the camera

Description automatically generated with medium confidence

Q: What's your favorite kind of competition and why? In terms of techniques, solving approaches, what is your specialty on Kaggle?

A: I love any non-code competition. ☺ This has changed a lot over the years. I used to be really into the image competition, but the sophistication of the engineering stack required to be competitive in these has increased tremendously over the years. For a while I was really into the NLP competitions, but those have always been rare on Kaggle. One constant over the years, though, has been my interest in tabular data problems. Those used to be **the** quintessential Kaggle competition problems but have unfortunately become extinct. I am still very interested in that area of ML and have moved on into doing some basic research in this domain. Compared to the other areas of ML/DL, there has been very little progress on improving ML for tabular data, and I believe there is a lot of opportunity here.

Q: How do you approach a Kaggle competition? How different is this approach to what you do in your day-to-day work?

A: I have always taken the game aspect of Kaggle seriously. ☺ What that means for me, I usually start new Kaggle competitions very playfully – submitting simple solutions, whimsical solutions, modified solutions from other players, blends, etc. These help me get a feel for the problem, what sorts of things work, how far can I get with a few simple tricks, etc. Some of this is also applicable to my day-to-day modeling, but there one important aspect is missing – and that’s the support and feedback from the community and the leaderboard. When you are working on your own or with a small team, you never know if what you are building is the best that can be done, or if better solution is possible.

Q: Tell us about a particularly challenging competition you entered, and what insights you used to tackle the task.

A: The most challenging and the most important competition of my Kaggle career has been the Home Credit underwriting competition. It is the second biggest Kaggle competition of all time, and it happened during a particularly challenging time in my life. Credit underwriting is a very challenging Data Science problem and requires a lot of intelligent feature engineering and a reliable validation scheme. My own personal insight was to use simple linear modeling for feature selection, and it helped our overall model. Our team won that competition, and to this day I consider this victory the highlight of my Kaggle career.

Q: Has Kaggle helped you in your career? If so, how?

A: Kaggle has been the single biggest booster of my ML career. Out of four ML jobs that I have held, three have been a direct consequence of my Kaggle success. It is impossible to overstate how important a Kaggle credential can be in one’s career.

Q: In your experience, what do inexperienced Kagglers often overlook? What do you know now that you wish you'd known when you first started?

A: There are two aspects of all ML problems, and Kaggle competitions in particular, that I have either underappreciated, or not bothered enough with for way too long: feature engineering and robust validation strategy. I love ML libraries and algorithms, and have the tendency to start building the Ml algorithm as soon as I can. But the single biggest impact on your model’s performance will come from very good features. Unfortunately feature engineering is more of an art than science, and is usually **very** model and dataset dependent. Most of the more interesting feature engineering tricks and practices are rarely, if ever, taught in standard ML courses or resources. Many of them can not be taught and are dependent on some special problem-specific insights. But the mindset of looking into feature engineering as default is something that can be cultivated. It will usually take many years of practice to get good at it.

Q: Are there any tools / libraries that you would recommend using for Kaggling?

A: XGBoost is all you need!