A Risk Management Process for Machine Learning Projects

Ben Cohen

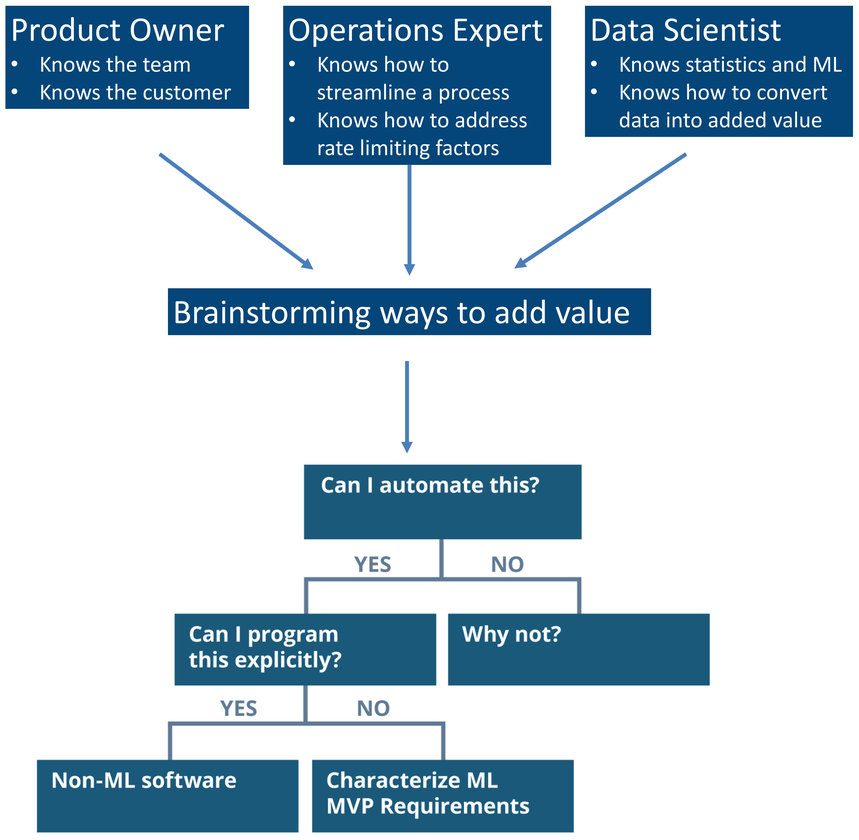
Many machine learning projects fail, underwhelm, or backfire. In the present work, I have identified three common risks in machine learning projects. To mitigate these risks, I describe a general risk management process that takes place before a project launches.

Risk #1: Poor problem solution alignment

Examples include using overly sophisticated or hard to interpret models. In some cases, a problem is best solved with software, not machine learning.

Risk management for risk #1

Focus on adding value, not adding machine learning. To add value via machine learning, three key players can exercise their creativity to find ways to improve a system.



Considerations for MVP

- Customer needs, organization's mission
- Compute environment / time / memory constraints
- Optimizing metrics, strategy
- Performance metrics, types of errors
- Interpretability

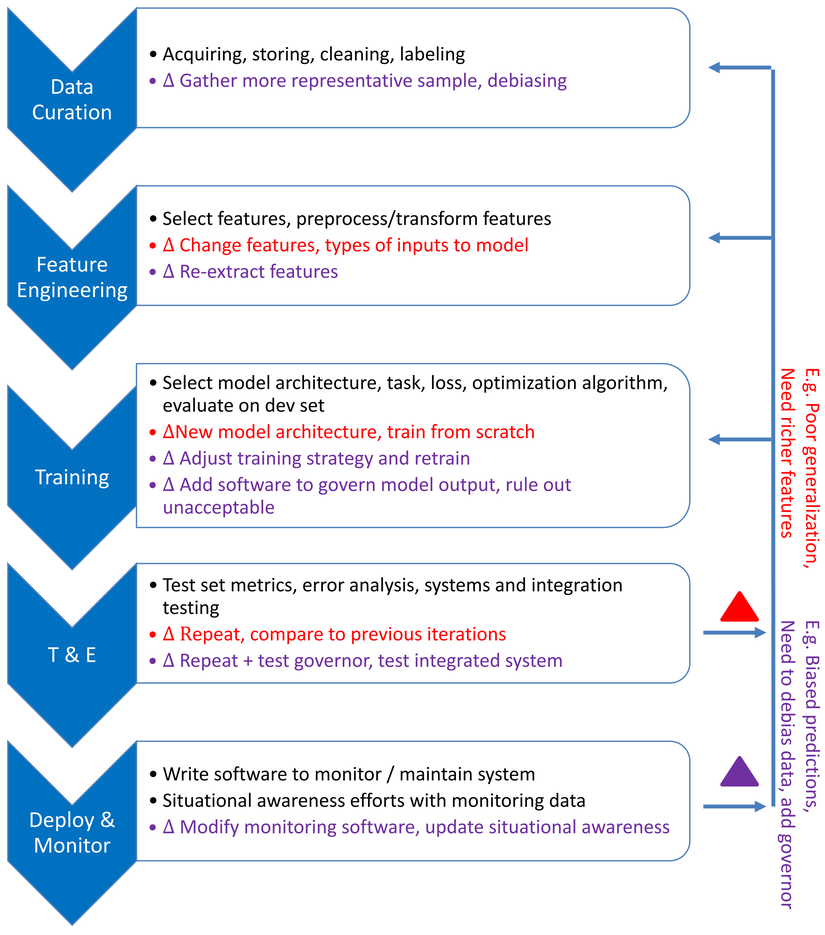
Risk #2: Incurring excessive cost

The Agile methodology for software engineering embraces changing requirements. Unfortunately, the CASE[1] (changing anything changes everything) principle of machine learning raises the cost of adapting to changes on the fly.

Risk #3: Unexpected behavior in the wild

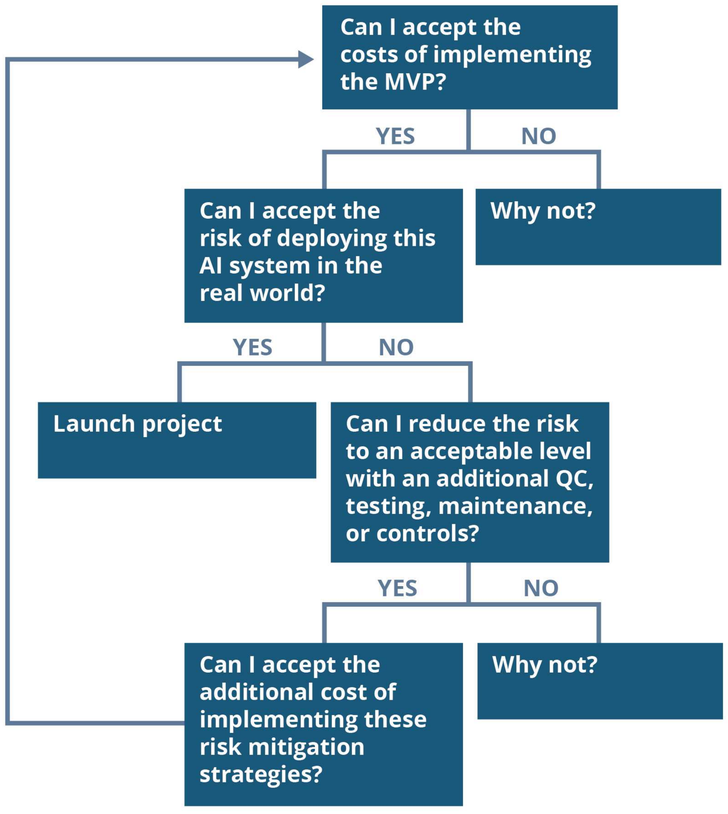
Common causes of surprising or undesirable behavior include differences between development and production environments, complex interactions between ML and non-ML components of a system, and biases in the data.

Risks 2 & 3 in the ML lifecycle[2]



Risk management for risks 2 & 3

Plan for change by walking the MVP through the ML lifecycle before launch. We can't anticipate every change, but we can estimate the likelihood or cost of some changes. This sets realistic expectations for a project and informs how risky it is.



Citations

1. Sculley *et al*, NIPS, 2015
2. Amerishi *et al*, *ICSE, 2019*