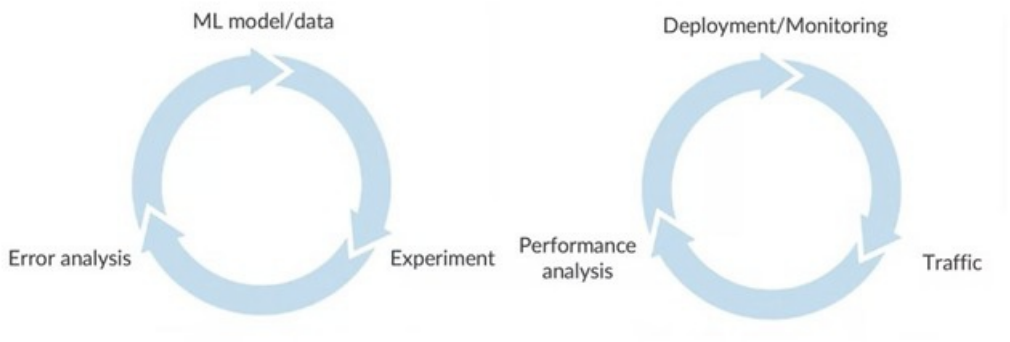
Monitoring dashboard

- The most common way to monitor a ML system is to use a **dashboard** to track how it's doing over time.
- Depending on your application, your dashboard may monitor different metrics.
    - **How to decide what to monitor?**
        - Brainstorm the things that could go wrong.
        - Brainstorm a few statistics/metrics that will detect the problem.
        - It's OK to use many metrics initially and gradually remove the ones you find not useful.
    - **Examples of metrics to track:**
        - ***Software metrics:*** Memory, compute, latency, throughput, server load
        - ***Input metrics:*** number of missing values, *speech recognition*: avg. input length, avg. input volume, *vision*: avg. image brightness $\rightarrow$ essentially check for data drift (change of data distribution)
        - ***Output metrics:*** *speech recognition:* # of times returning NULL, # of times user redoes search, # of time user switches to typing, or CTR, $\rightarrow$ essentially to figure out if your learning algorithm output y has changed in some way (concept drift).

Just as ML modeling is iterative, so is deployment.

It usually takes a few tries to converge to the right set of metrics to monitor.



Set thresholds

Common practice after you choose your metrics to monitor is to set thresholds for *alarms*. You can adapt metrics and thresholds over time.

If your model needs retraining, you can either do:

- Manual retraining (more common)
- Automatic retraining