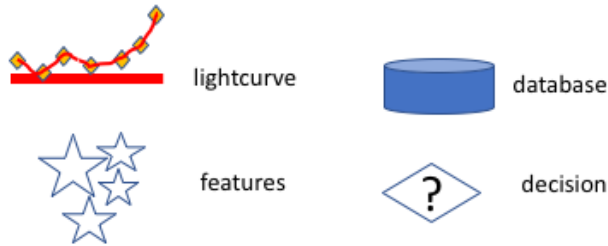


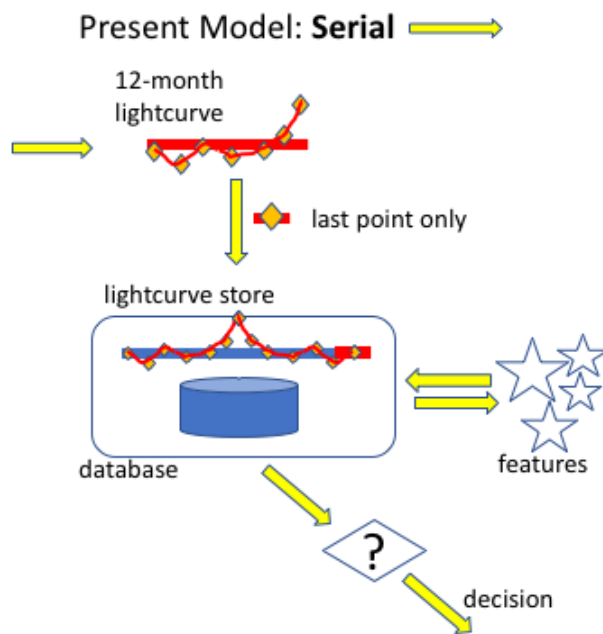
Handling Light Curves

Created by Roy Williams
Last updated Mar 09, 2020 • Analytics



A *lightcurve* is a history of the brightness of a source, and is built from multiple *detections* of that source. From a lightcurve, we wish to compute *features*, some simple such as min/max magnitude, magnitude slope, most recent

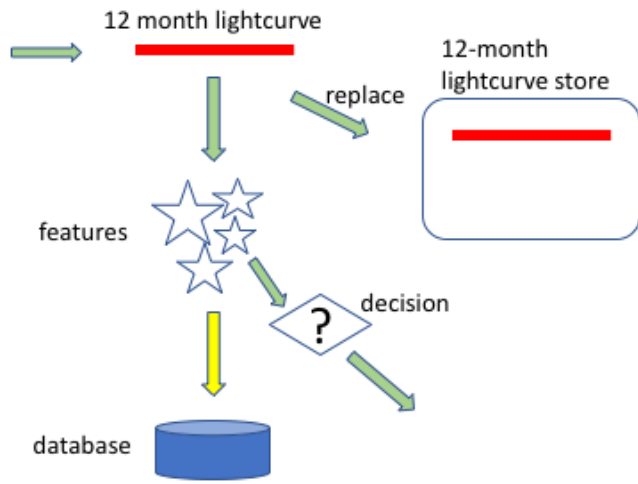
magnitude, etc, and some more complex, such as periodicity, machine-learning classifications, etc. In general, it is these features that go into user-built queries, rather than asking about individual detections.



Each alert carries not just a single detection, but a whole lightcurve. With ZTF there is a month of lightcurve, and LSST will provide a year of lightcurve. In the current (ZTF) implementation, we keep lightcurves much longer than a month; we take the last detection only from the alert and add it to the existing lightcurve in the database, then compute features and make decisions.

However, the database presents a serial bottleneck, and this process is difficult to parallelise.

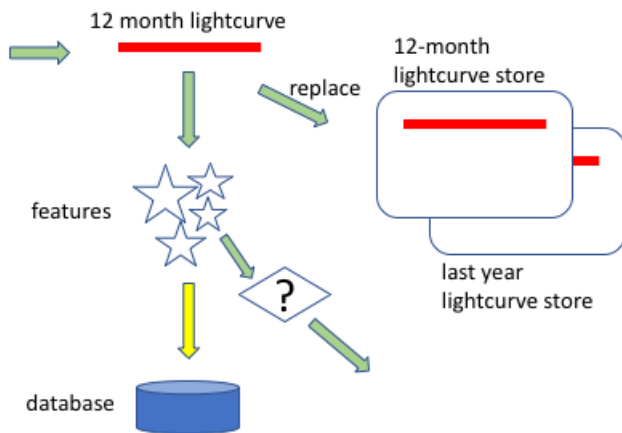
Future Model: Parallel



In order to respond rapidly to a firehose of alerts, we need to make decisions based on the lightcurve that comes with the alert. We compute features and test those against user criteria before streaming out the winning alerts to those users. Then we put the alerts into our prompt product database, so that users can experiment with these criteria against history. One way to do this is by replacing the existing SLC for

a given object with the new SLC.

Better Future Model: Parallel



After we have been operating Lasair-LSST for a year, we can "freeze" the existing set of SLCs, and start a new collection. After 2 years, each given source will have two distinct light curves, one for each year, and each will have its own set of features. Integrating the two to a single 2-year light curve would be straightforward.

Like Be the first to like this

No labels