Stranger Forecasts

Maxwell Peterson

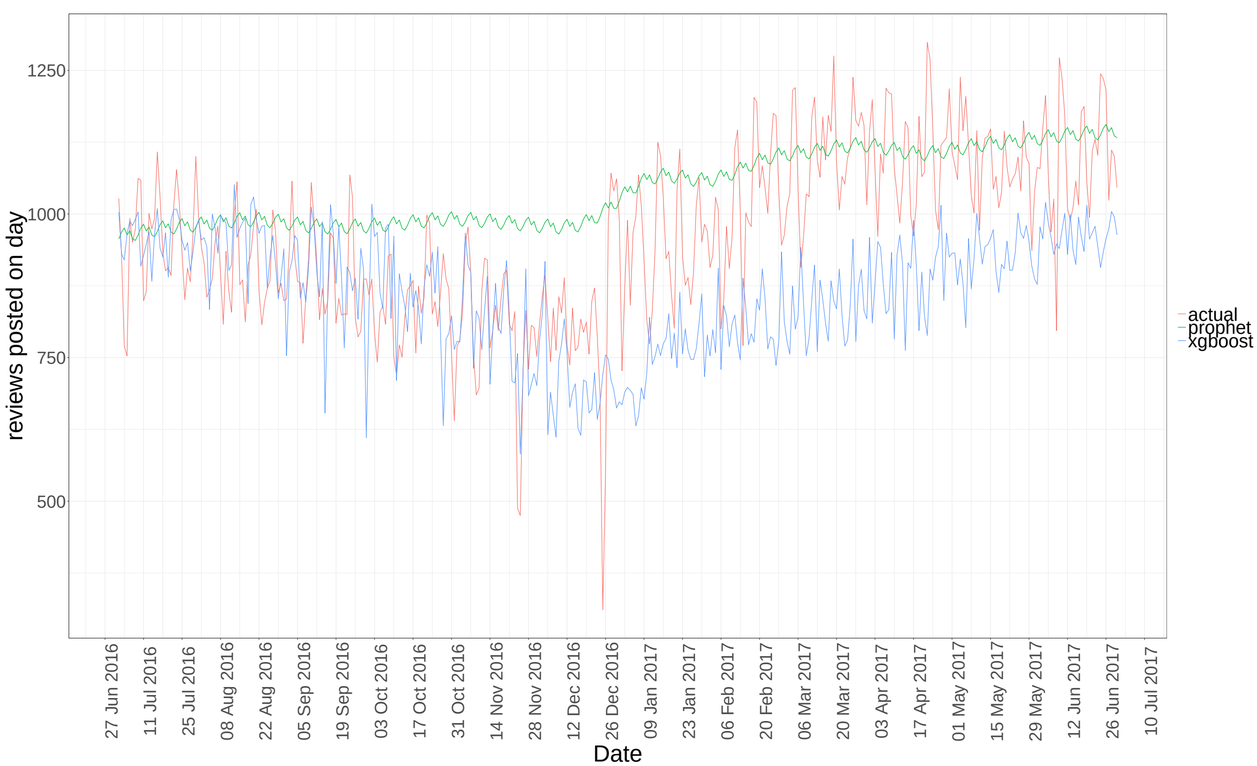
This is an experiment with timeseries prediction using the boosting algorithm xgboost. We focus on Arizona Yelp reviews-per-day.

The idea is to train an xgboost model where the input features to predict the value of the timeseries at y[t] are N lags of y up to time t. For example, if we used 3 lags, then to predict a future out-of-sample value y[i], we would use the 1-by-3 matrix [y[i - 1], y[i - 2], y[i - 3]].

For forecast horizons longer than 1 day, not all lags are known. For example, when forecasting the day k days after the final known date, there are k - 1 unknown lags of y[k]. Thus we use a one-step-forecast-and-feed procedure: starting at the day after the final known day, we predict one value at a time, and use the predictions as if they were known lags of the series. With this scheme, the kth lag of y[k] is the single value predicted at step y[k - 1].

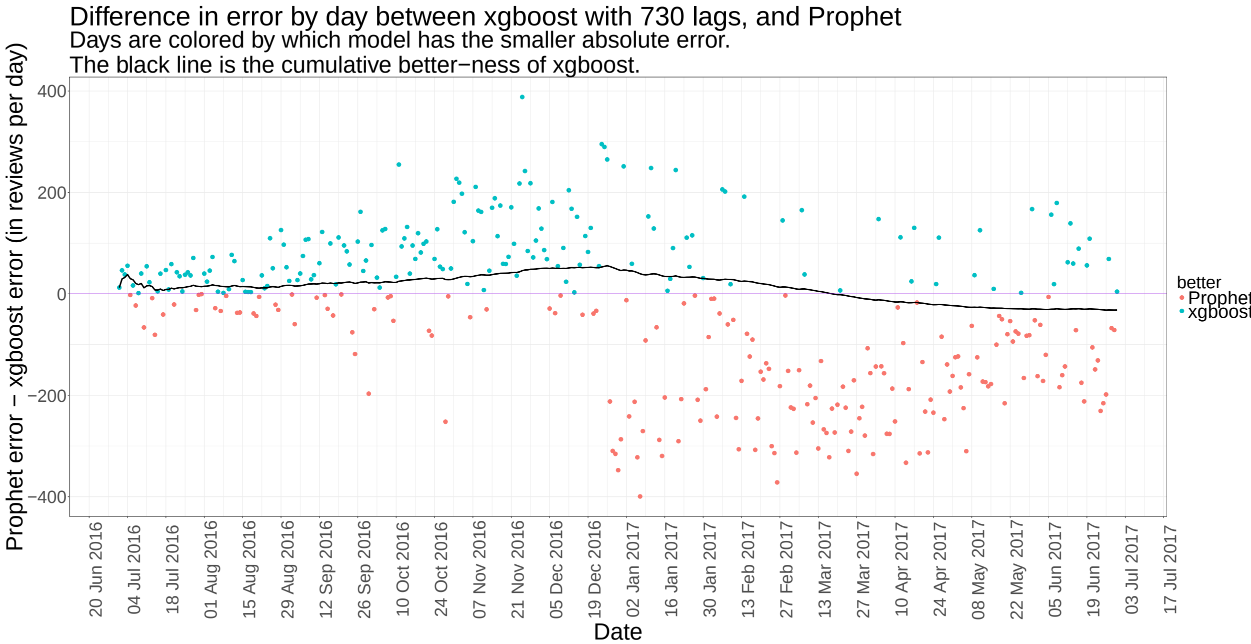
We train on data from December 24, 2004 to June 30, 2016, and forecast over a horizon of 365 days after June 30, 2016.

Using 730 lags as features, the following forecast results. The forecast from a prophet model trained on the same data is also plotted.



What interesting behavior! And what differences between the two methods! The prophet forecast is stodgy and safe, preferring to skip attempting to capture the day-to-day variation in favor of attempting (but not succeeding) to stay around the local mean of the series; the xgboost forecast wants it all, and reaches all around day-to-day. The immediate suggestion is that for long horizons, prophet may be a safer bet; but that for day-to-day variation attempts, this xgboost method has much more promise.

Xgboost is the better method for this series for the first 6 months or so, then Prophet begins to do better:



This is because xgboost failed to increase enough at the beginning of 2017, and was always shooting low.