[Kalman Filter Is All You Need: Optimization Works When Noise Estimation Fails]

Ido Greenberg, Shie Manor & Netanel Yannay, 2021

The Kalman Filter algorithm (KF)

KF is highly popular for filtering problems (e.g., tracking, navigation and control). It provides optimal predictions under the following assumptions:

- Known linear models for motion (F) & observation (H)
- I.i.d Gaussian noise with known covariance matrix in motion (Q) & observation (R)
- Known initial-state distribution (X_0)

KF parameters tuning

Most of the literature of KF focuses on determining the parameters R, Q from observations $\{z_t\}_t$, without knowing the hidden system states $\{x_t\}_t$. If the training data *does* include hidden states, R and Q can be directly determined through noise estimation: $\hat{R} \coloneqq Cov(\{z_t - Hx_t\}_t)$, $\hat{Q} \coloneqq Cov(\{x_{t+1} - Fx_t\}_t)$. With these $\hat{R}, \hat{Q}$, KF yields optimal predictions of $\{x_t\}_t$ (up to the estimation error of the noise).

So what is wrong?

The KF assumptions practically rarely hold – even in very simplistic scenarios. For example, in the standard problem of Doppler radar tracking:

- Motion model (F) is not linear (nor known)
- Observation model (H) is not linear
- Observation noise (R) is i.i.d in polar coordinates – but not in Cartesian ones
- Initial-state distribution is unknown

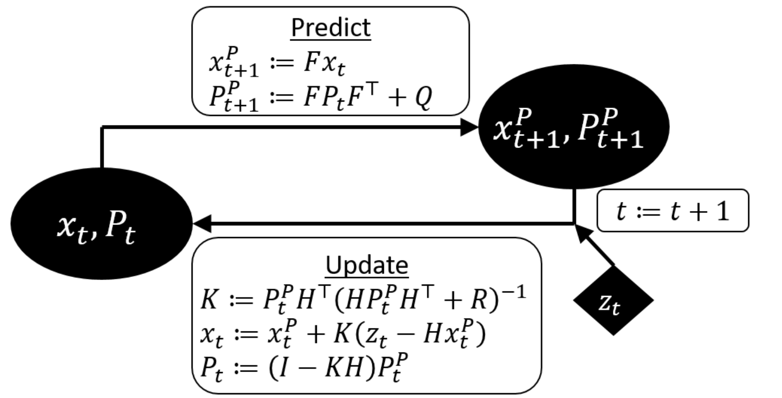
Once the KF assumptions are violated, determining R, Q by noise estimation is no longer equivalent to optimizing the predictions, i.e., the wrong problem is addressed. **This observation re-opens a problem considered solved for decades**.

How can we solve it?

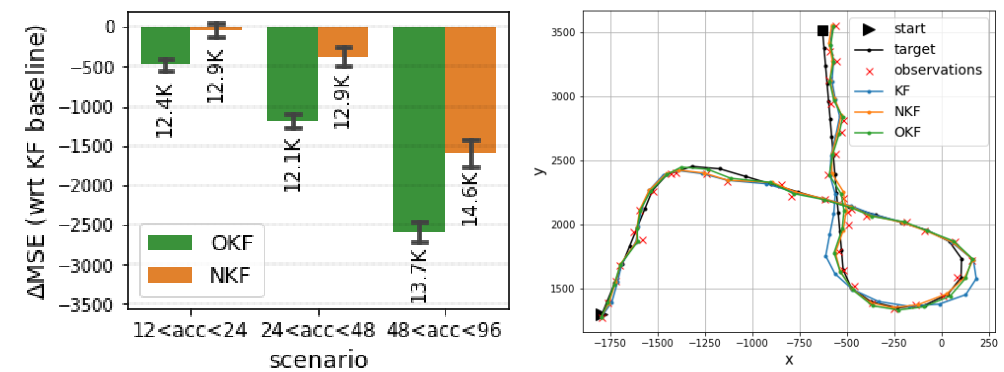
Given states ground-truth $\{x_t\}_t$ in the training data, the KF can be run on the data and optimized by standard gradient-based methods (e.g., Adam) wrt the prediction errors. The optimized parameters (R, Q) represent covariance matrices, thus we use the Cholesky parameterization to keep them symmetric & positive-definite.

Does it really matter?

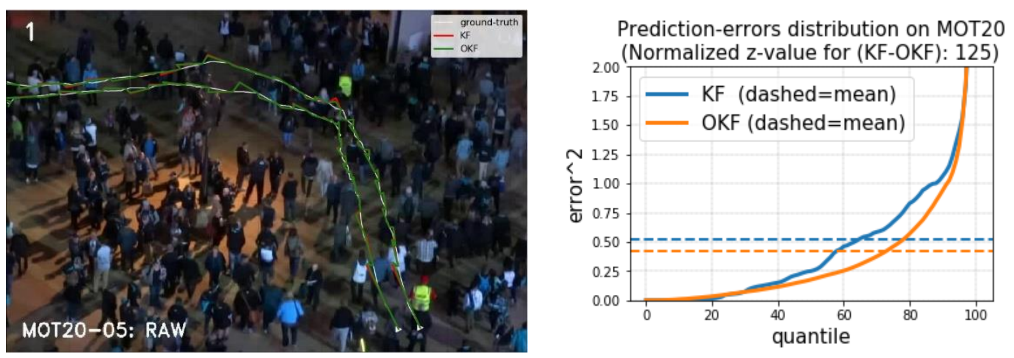
- We show analytically how violating different assumptions causes different changes in the optimal parameters.
- In experiments, optimization reduced prediction errors under any subset of assumptions violations.
- Even when the only violation was non-linear H – optimization reduced the errors by 15%-45%.
- Optimization also compensated for "wrong" design (e.g., Cartesian or polar representation).
- In other experiments, an LSTM reduced the prediction errors wrt a standard KF but not wrt an optimized one. That is, by not optimizing, we may unnecessarily adopt over-complicated models.



The KF algorithm



***Relative* tracking errors of an Optimized KF (OKF) and a KF with LSTM predictor (NKF) – compared to a standard KF. The label of each bar corresponds to the *absolute* MSE. The right figure shows a sample target (projected onto XY plane). All models were learned over targets with acceleration range of 24-48, then tested on targets with acceleration ranges 12-24, 24-48 and 48-96. While the LSTM seems to beat the linear KF – its advantage is entirely eliminated once the KF is optimized.**



Pedestrians tracking (MOT20 dataset): optimization reduces KF's errors by 18%.