Finishing the Kalman filter and the semester

ISTA 410 / INFO 510: Bayesian Modeling and Inference

U. of Arizona School of Information

May 5, 2021

Outline

Last time:

- Intro to the Kalman filter

Today:

- A few more details on the Kalman filter
- Semester recap

The filtering algorithm

After setting initial estimates for $\hat{x}_0^-, P_0^-$, the filtering algorithm proceeds in two steps:

1. Obtain prior estimates $\hat{x}_k^-, P_k^-$ by applying the dynamical system to $\hat{x}_{k-1}, P_{k-1}$.
2. Compute the Kalman gain K_k and adjust estimates to $\hat{x}_k, P_k$.

Notes:

- We're taking advantage of normality and linearity here; all conditional and marginal distributions remain normal, so we can work only in terms of the mean/covariance.
- If Q, R are constant, then the estimate error covariance P_k and the gain K_k stabilize quickly and then stay constant.

Simplest possible example: estimating a constant

A very simple example comes down to estimation of an unknown constant.

For example: we are trying to estimate a voltage, but our instruments are faulty, introducing an amount of noise to each measurement.

This implies the following parameters:

- $A = 1$ (no deterministic time evolution of states)
- $H = 1$ (measuring voltage directly)
- $Q \approx 0$ (assume negligible fluctuation in states)
- $R = R_0$ (fixed measurement error)

Kalman filter equations

We can write down the Kalman filter equations for this version easily:

Dynamics update step:

$$\hat{x}_k^- = \hat{x}_{k-1}$$
$$P_k^- = P_{k-1} + Q$$

We can in principle set $Q = 0$, but we can also adjust it to allow for some fluctuations in the true voltage.

Kalman filter equations

The correction step:

$$K_k = \frac{P_k^-}{P_k^- + R}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - \hat{x}_k^-)$$

$$P_k = (1 - K_k)P_k^-$$

Initial parameters

We need to pick values for:

- Q: we'll use 10^{-5} for something very small but nonzero.
- $\hat{x}_0$: we'll start with 0
- P_0: this one determines how quickly we converge to a stable estimate – we'll try a few examples
- R: this determines how much we "trust" the noisy measurements – we'll try a few examples

Let's try it...

Parameters and tuning

A couple of comments on choosing the parameters Q, R:

- Measuring R empirically is usually practical, because it's a property of our measurement
- Q is trickier. Can come from a scientific model (ideally).
- Q can "compensate" for a poor process model by adding more uncertainty to state estimates – if we don't really know the dynamics, treat them as noise and the filter can still estimate states
- Parameter tuning can be done off-line: use existing data to select and tune Q, R, then use them on new data
  - If Q, R are constant, K and P can also be precomputed

Adding in some dynamics

We can add a little bit of complexity by introducing some dynamics:

- Assume voltage is not constant, but decays over time
- At each time step, system loses r% of voltage

Modifications:

- $A = (1 - r)$
- Update step: instead of

$$\hat{x}_k^- = \hat{x}_{k-1}, \quad P_k^- = P_{k-1} + Q$$

have

$$\hat{x}_k^- = (1 - r)\hat{x}_{k-1}, \quad P_k^- = (1 - r)^2 P_{k-1} + Q$$

Adding in some dynamics

Let's test 3 approaches:

- Use the original filter, unmodified (this is bad)
- Use a filter that knows the decay dynamics (this is good)
- Use the original filter, but increase Q to compensate (this is okay, more or less)

Modifications and extensions

Kalman filter with control

Kalman initially developed the filter for applications in control theory. So, alternate form:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_k$$

(states)

$$z_k = Hx_k + v_k$$

(measurements)

- u_{k-1}, the control input, represents some linear forcing we can do to the system
- B defines how the control input influences the system

Modifying the filter equations

It turns out the only modification we need to make is to the prediction step:

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1}$$
$$P_k^- = AP_{k-1}A^T + Q$$

Nonlinear dynamics

Kalman filters assume linear dynamical systems, but many systems are nonlinear

Option 1: extended Kalman filter

- Linearize the dynamical system around the estimate at each time step
- Advantage: pretty simple, same ideas work
- Disadvantages:
  - Assumptions of normality no longer hold
  - No longer really Bayesian – just an approximation
  - Errors can accumulate over time

Nonlinear dynamics

Other approaches, based on sampling:

- Ensemble Kalman filter: run a large number of Kalman filters, with added noise, and average the results
- Unscented Kalman filter: in the update step, project forward a deterministically chosen set of points to estimate the mean/variance at the next time step
- Particle filter: use importance sampling to maintain a sample of state sequences, pruning and splitting them at each time step

Semester recap

What did we learn?

We learned a lot of stuff over the course of the semester:

- A number of standard probability distributions
- Prior and posterior distributions
- Model evaluation using predictive checks and information criteria/LOO-CV
- Multilevel/hierarchical models
- DAGs, multiple regression, and causal inference
- Computational methods and MCMC
- Interaction models
- Modeling covariance
- Gaussian processes
- Dynamical models: HMM, Kalman filter

The big ideas

Big takeaways:

The big ideas

Big takeaways:

- Bayesian models are generative
  - Based on a description of the data-generating process
  - Can simulate multi-step data generation, with uncertainty/variation at each step

The big ideas

Big takeaways:

- Bayesian models are generative
  - Based on a description of the data-generating process
  - Can simulate multi-step data generation, with uncertainty/variation at each step
- Bayesian models are modular
  - Construct by specifying relationships between variables
  - Can mix and match components – cf. bike share example, Poisson regression + GP
  - Allows incremental expansion of modeling – start simple, add components

The big ideas

Big takeaways:

The big ideas

Big takeaways:

- Models share information across categories
  - Hierarchical models model variation at several scales (cf. bike traffic example)
  - This can, if needed, include explicit modeling of correlations between parameters (cf. Beijing ozone example)

The big ideas

Big takeaways:

- Models share information across categories
  - Hierarchical models model variation at several scales (cf. bike traffic example)
  - This can, if needed, include explicit modeling of correlations between parameters (cf. Beijing ozone example)
- Model evaluation is based on the generative properties
  - Can the model, after learning parameter values, reproduce data that is similar to observed data? (posterior predictive check)
  - Does the model assign high probability to observed data, without too many parameters? (information criteria)

The big ideas

Big takeaways:

- Models share information across categories
  - Hierarchical models model variation at several scales (cf. bike traffic example)
  - This can, if needed, include explicit modeling of correlations between parameters (cf. Beijing ozone example)
- Model evaluation is based on the generative properties
  - Can the model, after learning parameter values, reproduce data that is similar to observed data? (posterior predictive check)
  - Does the model assign high probability to observed data, without too many parameters? (information criteria)
- All of this can be estimated using MCMC
  - Modern probabilistic programming frameworks put together Monte Carlo samplers for you

The big ideas

Big takeaways:

The big ideas

Big takeaways:

- Interactions: effects are conditional on the values of other variables (i.e. just more conditional probabilities)
  - Different slopes in different categories; products of predictors; etc.
  - Relation to multilevel models (cf. ruggedness and African GDP example)

The big ideas

Big takeaways:

- Interactions: effects are conditional on the values of other variables (i.e. just more conditional probabilities)
  - Different slopes in different categories; products of predictors; etc.
  - Relation to multilevel models (cf. ruggedness and African GDP example)
- Causal inference: we can infer causal effects from observational data if we have a sufficiently detailed model of relationships
  - Key idea: conditional independence structure
  - DAG: graphical representation of these independences
  - Confounding and non-causal associations appear as indirect paths through the DAG
  - Paths can be closed or opened by conditioning on the right variables

Summary

Today:

- Discrete-time Kalman filter
- End of the semester (yay!)

Next time:

- There is no next time! Thank you for sticking with me!