Applying Kalman Filter to Denoise GPS Data

Kai Liu [KL25756]

The Institute for Cellular and Molecular Biology
The University of Texas at Austin

Abstract

In this project, I smoothed GPS data collected from a police vehicle using Kalman Filter. When designing the process and measurement noise covariance matrices, I implemented two methods: empirical method and Autocovariance Least-Squares (ALS) method, and compared the smoothing results using the noise covariance matrices **Q** and **R** estimated by each of the two methods. The conclusion is that the performance of empirical method is better than that of the ALS method. In addition, the process noise covariance matrix **Q** estimated directly from empirical method has heavy effect on smoothing GPS data. In order to smooth GPS data reasonably, matrix **Q** should be scaled by a power of ten. It also indicates that the smaller the matrix **Q** is, the smoother the denoised data is than the original GPS data.

1. Introduction

The Global Positioning System (GPS) is a worldwide radio system and has application in aviation, aircraft automatic approach and landing, and vehicle navigation and tracking, *etc*. A GPS receiver is a low frequency response navigation sensor and can provide instantaneous position accuracy in the order of 15 to 100 m normally. Therefore, data collected from GPS has lots of noise, and thus filtering noise becomes very necessary.

There are many methods to denoise GPS data, such as Kalman filter method, Particle filter method, least square fit approach *etc*. While a least square fit approach will just use positional information to smooth data, both Kalman filter and Particle filter will smooth the data taking velocities into account. In this project, I implement Kalman filter approach to smooth GPS data collected from a police vehicle. Kalman filter is an optimal estimation method that has been widely used in smoothing noisy signals, generating non-observable states, and predicting future states. It estimates the position in two stages: prediction and update. In the prediction stage, it produces a new position estimate based on the previous position and the dynamic model. And then in the update stage, after a new measurement of the vehicle's position is taken from GPS, the estimate is updated using a weighted average with more weight being given to estimates with higher certainty. In Kalman filter, one of the most difficult part is to design process and measurement noise covariance matrices **Q** and **R**. Two popular methods are implemented to estimate these two matrices: empirical approach and Autocovariance Least-Squared (ALS) approach. A comparison between these two approaches is also shown in section 4.

The paper is organized as follows. A Kalman filter design is given in Section 2. Methods for determining covariance matrices of process and measurement noises are shown in Section 3. Analysis results are presented in Section 4. Finally, conclusions are made in Section 5.

2. Kalman Filter Design

Central for all navigation and tracking applications is the motion model to which various kind of model based filters can be applied. Model that are linear in both the state dynamics and the measurements are considered:

$$\boldsymbol{x}_t = \boldsymbol{F}\boldsymbol{x}_{t-1} + \boldsymbol{\omega}_t \tag{1}$$

$$\boldsymbol{Z}_t = \boldsymbol{H}\boldsymbol{x}_t + \boldsymbol{v}_t \tag{2}$$

Here $\boldsymbol{x}_t$ is state vector containing the terms of interest (i.e. location, velocity) at time *t*; $\boldsymbol{F}$ is the state transition matrix applying the effect of each state parameter at time *t-1* on the state at time *t*; $\boldsymbol{\omega}_t$ is the vector containing process noise for each variable in the state vector, and the noises are assumed to follow Gaussian distributions with known mean 0 and covariance matrices $\boldsymbol{Q}_t$; z_t is the vector of measurements; $\boldsymbol{H}$ is the transformation matrix that maps the state vector variables into the measurements; and $\boldsymbol{v}_t$is the vector including the measurement noises for each observation, and it is also assumed that the noises are Gaussian with zero mean and covariance matrix $\boldsymbol{R}_t$.

The motion model. The signals of primary interest in navigation and tracking are related to position, velocity, and acceleration. In this project, I only have position data available, so I assume that velocity is constant, and position and velocity at time *t* are

$$p_t = p_{t-1} + v_{t-1}\Delta t$$

$$v_t = v_{t-1}$$

Since GPS position information contains longitude and latitude, I consider velocity on each direction separately. Therefore, the state vector $\boldsymbol{x_t}$ is

$$\boldsymbol{x_t} = [p_{lon}^t \;\; p_{lat}^t \;\; v_{lon}^t \;\; v_{lat}^t]^T$$

The above linear equations can be written in matrix form as

$$[p_{lon}^t \;\; p_{lat}^t \;\; v_{lon}^t \;\; v_{lat}^t]^T = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} [p_{lon}^t \;\; p_{lat}^t \;\; v_{lon}^t \;\; v_{lat}^t]^T$$

And so by comparison with (1),

$$\boldsymbol{F} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The measurement equation. The measurement equation includes a transformation matrix which is necessary if the unit of state vector $\boldsymbol{x_t}$ is different from measured location vector $\boldsymbol{z_t}$. In this project, the police vehicle was cruising only around UT campus, which is a very small region relatively to the earth. It is reasonable to use longitude and latitude to calculate distance directly. Therefore, I used same units for both $\boldsymbol{x_t}$ and $\boldsymbol{z_t}$. Based on equation (2),

$$\boldsymbol{z_t} = [z_{lon}^t \;\; z_{lat}^t]^T \text{ and } \boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Process and measurement noises. The design of the covariance matrices of process and measurement noises are among the most difficult aspects of Kalman filter design. Details about estimating these two covariance matrices are discussed in Section 3.

Prediction and update. Kalman filter involves two steps: prediction step and update step. The prediction is based on the previous states and dynamic model. The equations are

$$\widehat{\boldsymbol{x}}_{t|t-1} = \boldsymbol{F}\widehat{\boldsymbol{x}}_{t-1|t-1} \tag{3}$$

$$\boldsymbol{P}_{t|t-1} = \boldsymbol{F}\boldsymbol{P}_{t-1|t-1}\boldsymbol{F}_t^T + \boldsymbol{Q}_t \tag{4}$$

where $\hat{x}_{t|t-1}$, $P_{t|t-1}$ are priori state and covariance estimates, respectively. The update starts after a new measurement is available. The equations are given by

$$\boldsymbol{K}_t = \boldsymbol{P}_{t|t-1}\boldsymbol{H}^T(\boldsymbol{H}^T\boldsymbol{P}_{t|t-1}\boldsymbol{H}^T + \boldsymbol{R}_t)^{-1} \tag{5}$$

$$\widehat{\boldsymbol{x}}_{t|t} = \widehat{\boldsymbol{x}}_{t|t-1} + \boldsymbol{K}_t(\boldsymbol{z}_t - \boldsymbol{H}\widehat{\boldsymbol{x}}_{t|t-1}) \tag{6}$$

$$\boldsymbol{P}_{t|t} = (\boldsymbol{I} - \boldsymbol{K}_t\boldsymbol{H})\boldsymbol{P}_{t|t-1} \tag{7}$$

where $\hat{x}_{t|t}$, $P_{t|t}$ are posteriori state and covariance estimates, respectively. These two steps are repeated continuously to update the current state.

3. Determination of Covariance Matrices of Process and Measurement noises

In Kalman filter, knowledge about process and measurement noise statistics is required. The performance of a Kalman filter relies on properly defined noise statistics. Failure to do so in the design of a Kalman filter could result in large estimation errors. However, in practical applications, the noise covariances are generally not known. Recently many approaches have been developed to improve the accuracy of noise covariance estimation. In this project, I implemented two methods to estimate noise covariances.

Empirical method. In this method, I assumed that the highest order term acceleration is constant for the duration of each time period, but can differ for each time period, and each of these is uncorrelated between time periods. So this can be modeled as

$$\boldsymbol{x}_t = \boldsymbol{F}\boldsymbol{x}_{t-1} + \mathbf{G}_t\boldsymbol{\omega}'_t$$

where $\mathbf{G}_t$ is the noise gain of the system, and $\boldsymbol{\omega}_t$ is the constant piecewise acceleration. In one time period, the change in velocity is $\boldsymbol{\omega}_t\Delta t$, and the change in position is $\boldsymbol{\omega}_t\Delta t^2/2$. This gives us

$$\boldsymbol{G}_t = \begin{bmatrix} \frac{1}{2}\Delta t^2 & 0 \\ 0 & \frac{1}{2}\Delta t^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix}$$

Based on equation (1), $\boldsymbol{\omega}_t = \boldsymbol{\Gamma}_t\boldsymbol{\omega}'_t$. Therefore, the covariance matrix of the process noise is then

$$\boldsymbol{Q}_t = E[\boldsymbol{\omega}_t\boldsymbol{\omega}_t^T] = E[\boldsymbol{G}\boldsymbol{\omega}'_t\boldsymbol{\omega}_t'^T\boldsymbol{G}^T] = \boldsymbol{G}\boldsymbol{\sigma}_v^2\boldsymbol{G}^T$$

where $\boldsymbol{\sigma}_v^2$ is the covariance matrix of velocities along both longitude and latitude direction.

In addition, the covariance matrix of measurement noise is the covariance matrix of positions on longitude and latitude.

Autocovariance Least-squares method. In the approach, innovation based correlation technique is used. In equation (6), $\boldsymbol{K}$ is the estimator gain, and the residuals of the output equations $(\boldsymbol{z}_t - \boldsymbol{H}\widehat{\boldsymbol{x}}_{t|t-1})$ is the K-innovations. With the standard linear state estimator, the state estimation error, $\boldsymbol{\varepsilon}_t = \boldsymbol{x}_t - \widehat{\boldsymbol{x}}_{t|t-1}$, evolves according to

$$\boldsymbol{\varepsilon}_t = (\boldsymbol{F} - \boldsymbol{FKH})\boldsymbol{\varepsilon}_{t-1} + [\boldsymbol{I}, -\boldsymbol{FK}]\begin{bmatrix}\boldsymbol{\omega}_t \\ \boldsymbol{v}_{t-1}\end{bmatrix} \tag{8}$$

Thus, the state-space model of the K-innovation can be defined as

$$\boldsymbol{\varepsilon}_t = \overline{\boldsymbol{F}}\boldsymbol{\varepsilon}_{t-1} + \overline{\boldsymbol{G}}\overline{\boldsymbol{\omega}}_t$$

$$\boldsymbol{y}_t = \boldsymbol{H}\boldsymbol{\varepsilon}_t + \boldsymbol{v}_t$$

where $\overline{\boldsymbol{F}} = \boldsymbol{F} - \boldsymbol{FKH}$, $\overline{\boldsymbol{G}} = [\boldsymbol{I}, -\boldsymbol{FK}]$, $\overline{\boldsymbol{\omega}}_t = \begin{bmatrix}\boldsymbol{\omega}_t \\ \boldsymbol{v}_{t-1}\end{bmatrix}$, and $\boldsymbol{y}_t = \boldsymbol{z}_t - \boldsymbol{H}\widehat{\boldsymbol{x}}_{\boldsymbol{t|t-1}}$

Now consider the autocovariance, defined as the expectation of the data with some lagged version of itself

$$\boldsymbol{\ell}_j = E[\boldsymbol{y}_t\boldsymbol{y}_{t+j}^T]$$

The autocovariance of the measurement prediction or estimate error is given by

$$E[\boldsymbol{y}_t\boldsymbol{y}_t^T] = \boldsymbol{HPH}^T + \boldsymbol{R} \tag{9}$$

$$E\left[\boldsymbol{y}_t\boldsymbol{y}_{t+j}^T\right] = \boldsymbol{H}\overline{\boldsymbol{F}}^{\boldsymbol{j}}\boldsymbol{PH}^T + \boldsymbol{H}\overline{\boldsymbol{F}}^{\boldsymbol{j-1}}\boldsymbol{FKR},\ j \geq 1 \tag{10}$$

Then autocovariance matrix (ACM) is then defined as

$$\boldsymbol{\mathcal{R}}(N) = \begin{bmatrix}\boldsymbol{\ell}_0 & \cdots & \boldsymbol{\ell}_{N-1} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\ell}_{N-1}^T & \cdots & \boldsymbol{\ell}_0\end{bmatrix},\ N\ is\ the\ number\ of\ lags \tag{11}$$

Substitution of equation (9) and (10) into (11) followed by separation of the right-hand side into terms is performed. And then the *vec* operator is applied to both sides of the resulting equation. The *vec* operator is the column-wise stacking of a matrix into a vector. This allows the problem to be states as a linear least-squared problem $Ax = b$,

$$vec\left(\widehat{\boldsymbol{\mathcal{R}}}(N)\right) = \boldsymbol{A}[vec(\boldsymbol{Q})^T\ \ vec(\boldsymbol{R})^T]^{\boldsymbol{T}} \tag{12}$$

where $\boldsymbol{A}$ is a matrix composed of system matrices $\boldsymbol{F}, \boldsymbol{H}$ and Kalman filter gain $\boldsymbol{K}$. The left hand side of equation (12) can be estimated from data. Given a sequence of data $\{y_i\}_{i=1}^{N_d}$, the estimate of the autocovariance can be computed by

$$\widehat{\boldsymbol{\ell}}_j = \frac{1}{N_d - j}\sum_{i=1}^{N_d - j} y_i\, y_{i+j}^T$$

which is the unbiased autocovariance estimator. The solution for the least-squared problem is

$$[vec(\boldsymbol{Q})^T \ \ vec(\boldsymbol{R})^T]^{\boldsymbol{T}} = \boldsymbol{A}^{+} vec\left(\widehat{\boldsymbol{\mathcal{R}}}(N)\right), \quad \boldsymbol{A}^{+} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$$

4. Testing and results

Data set. The Kalman filter is applied to smooth GPS data collected from a police vehicle cruising around UT campus. The information I use in the dataset includes longitude, latitude and time stamp. The total number of samples is 814458.

I examine the time interval between each two neighbor samples. If the time interval between them is larger than 30 seconds, I assume these two samples were sampled from two different cruising events. In total, there are 4646 events. The number of samples in a single event ranges from to 1 to 11119. I only choose 1150 events each of which has more that 100 samples. There are two reasons: the first is Kalman filter needs time to converge; the second is when estimating matrix $\boldsymbol{Q}$ by ALS, data points in the beginning are usually ignored until initial condition is negligible. GPS data from different events is smoothed separately.

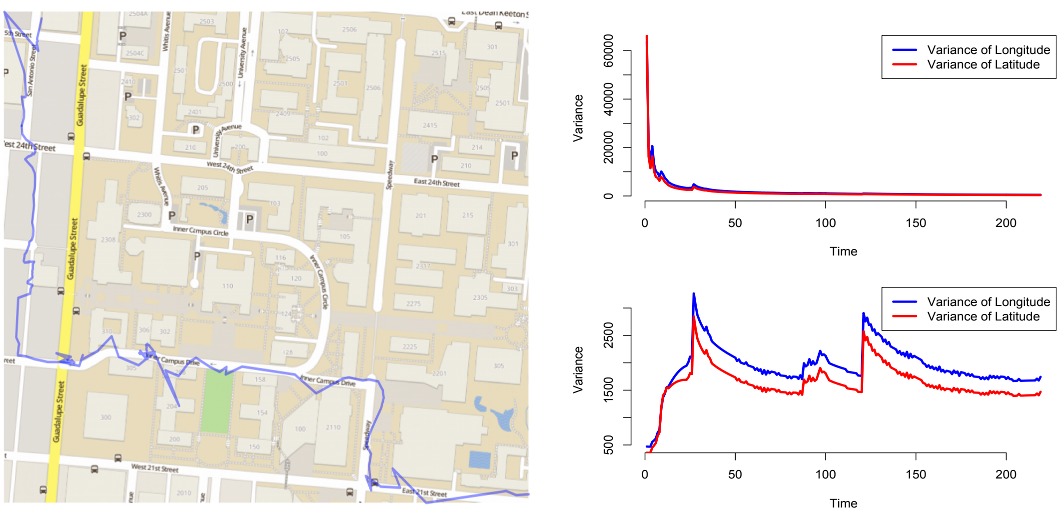
Smoothing results. In the paper, smoothing results of two events are given. Event I occurred between 2015-7-08 19:15:04 and 2015-7-08 19:27:27. The total number of samples is 219. Figure 1A shows the original GPS data. There are noises in the route record by GPS. Matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$ are estimated by both empirical method and ALS method.

	Matrix $\boldsymbol{Q}$ [Matrix $\boldsymbol{\omega}$]	Matrix $\boldsymbol{R}$
Empirical method	$\begin{bmatrix} 1.29E-09 & -4.56E-10 \\ -64.56E-10 & 1.68E-09 \end{bmatrix}$	$\begin{bmatrix} 27474.12 & -16773.61 \\ -16773.61 & 20913.06 \end{bmatrix}$
ALS method	$\begin{bmatrix} 0.026 & -0.0578 & 0.5446 & -0.5845 \\ -0.020 & 0.05712 & -0.5408 & 0.5697 \\ 0.451 & -0.5004 & 0.9841 & -1.0002 \\ -0.442 & 0.5026 & -0.9893 & 1.0004 \end{bmatrix}$	$\begin{matrix} 2.26E-07 & -2.26E-07 \\ -2.26E-07 & 2.340E-07 \end{matrix}$

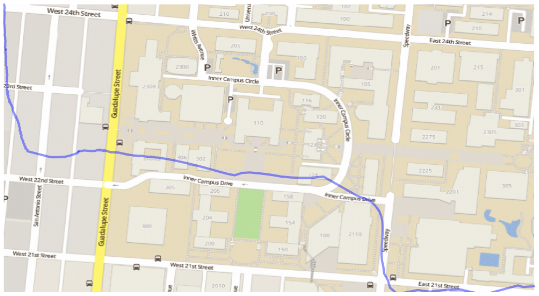
Event II occurred between 2015-7-04 05:29:19 and 2015-7-04 05:54:49. The total number of samples is 435. Figure 2A shows the original GPS data. The original data is also noisy. Matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$ are estimated using both empirical method and ALS method.

	Matrix $\boldsymbol{Q}$ [Matrix $\boldsymbol{\omega}$]	Matrix $\boldsymbol{R}$
Empirical method	$\begin{bmatrix} 4.98E-09 & -6.4E-10 \\ -6.4E-10 & 7.76E-09 \end{bmatrix}$	$\begin{bmatrix} 267439.8 & -105697.3 \\ -105697.3 & 254067.1 \end{bmatrix}$
ALS method	$\begin{bmatrix} 0.0186 & -0.0415 & 0.3906 & -0.4192 \\ -0.0144 & 0.0409 & -0.3879 & 0.4087 \\ 0.3232 & -0.3589 & 0.7059 & -0.7174 \\ -0.3170 & 0.3605 & -0.7096 & 0.7176 \end{bmatrix}$	$\begin{bmatrix} 1.436E-06 & -1.355E-06 \\ -1.35E-06 & 1.475E-06 \end{bmatrix}$

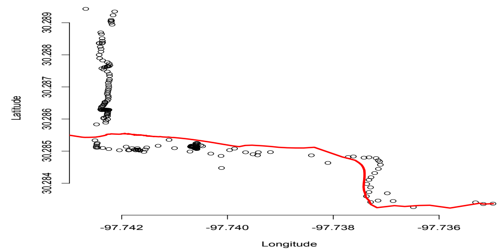
I smoothed data using matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$ estimated from both methods. However, when applying matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$ estimated by ALS approach, the smoothing result is noisier (results are not shown). It is possible that the three assumptions for time-invariant ALS method is not hold for the GPS data. In this paper, only smoothing results using matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$ estimated from empirical method are shown. The denoising results for event I and event II are given in Figure 1 and Figure 2, separately. When the original matrix $\boldsymbol{Q}$ is used (Figure1C, D, Figure2C, D), the Kalman filter converges very quickly (Figure1B upper, Figure2B upper); and the data is much smoother than original data. However, the results cannot match the actual road path. I also scaled the original matrix $\boldsymbol{Q}$ by 10^6 and 10^9 for event I and event II, respectively. The results are also smoother than the original data, and can almost match the actual road path, but the variances $\boldsymbol{P}$ of both longitude and latitude do not converge.



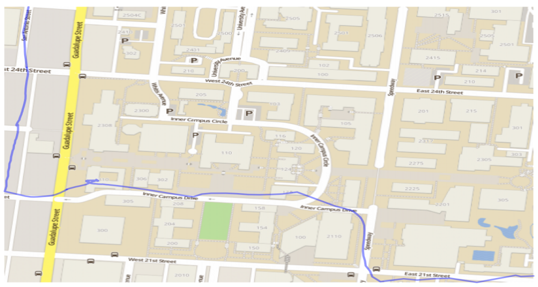
A **B**



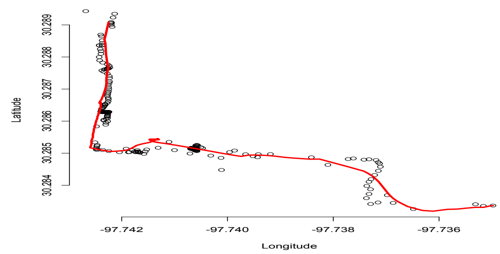
C



D

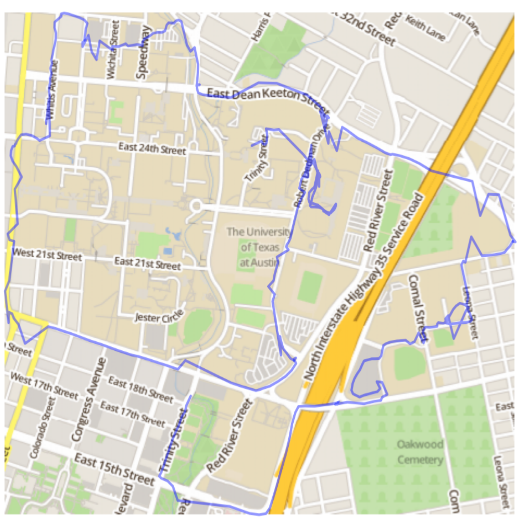


E

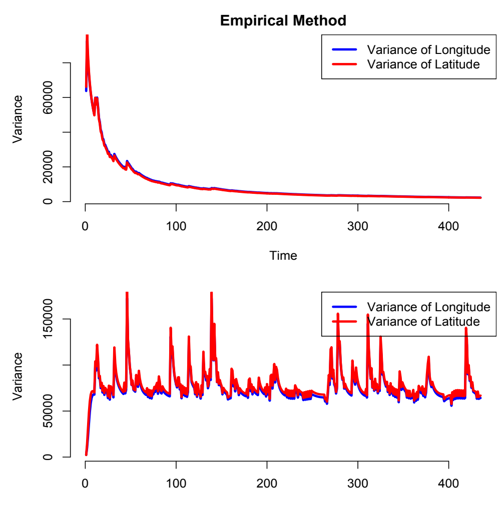


F

Figure 1 Smoothing results for event I. A) Original data; B) Convergence of variances of longitude and latitude. Upper plot is the result using original Matrix **Q**, while lower plot shows the result using scaled Matrix **Q**; C) D) Smoothing result using original Matrix **Q**; E) F) Smoothing result using original Matrix **Q**.



A



B

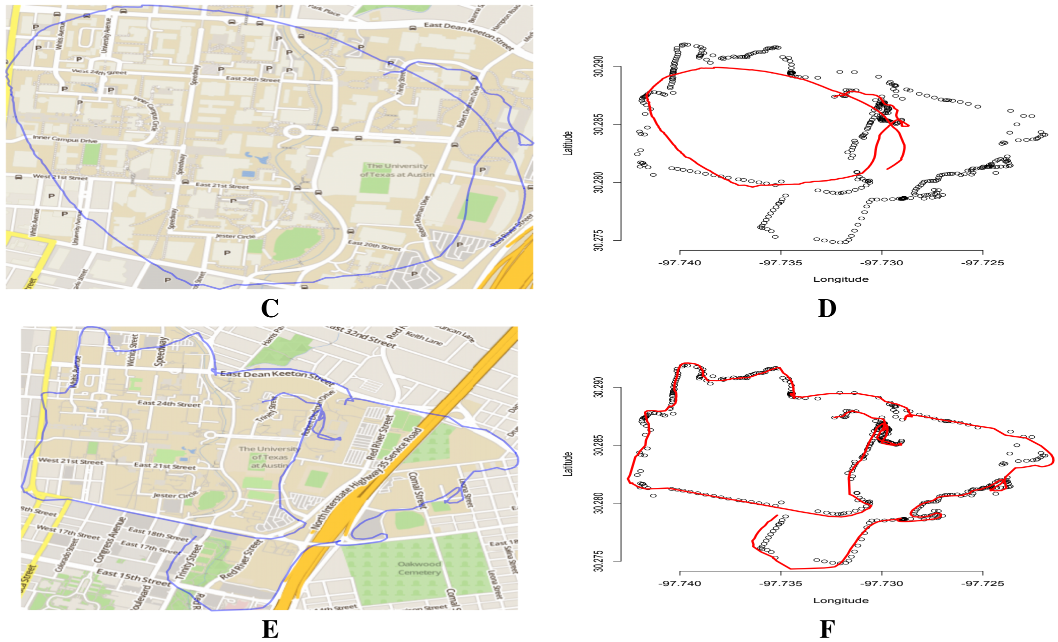


Figure 2 Smoothing results for event II. A) Original data; B) Convergence of variances of longitude and latitude. Upper plot is the result using original Matrix **Q**, while lower plot shows the result using scaled Matrix **Q**; C) D) Smoothing result using original Matrix **Q**; E) F) Smoothing result using original Matrix **Q**.

5. Conclusion and Future Work

In this project, I successfully denoise the GPS data using Kalman filter. In order to smooth GPS data reasonably, road path information is essential. Furthermore, the design of covariance matrices of process and measurement noises is critical for the performance of Kalman filter. I implemented two methods to estimate matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$. The covariance matrices estimated by empirical method perform well in smoothing GPS data, while the performance of ALS approach is not ideal. In the next step, I will investigate what assumption of ALS method is not satisfied by the GPS data, and how to improve the time-invariant ALS method.

6. Source Code

All project source code is available in the following GitHub repository

`https://github.com/KaiUT/denoising_GPS_data`

7. Reference

Faragher R. (2012) Understanding the Basis of the Kalman Filter via a Simple and Intuitive Derivation. *IEEE Signal Processing Magazine*. 128, September 2012.

Grewal MG, Andrews AP. (2008) Kalman Filtering: Theory and Practice Using MATLAB. Third Edition.

Labbe RR. (2016) Kalman and Bayesian Filters in Python. February 18, 2016.

Mehra RK. (1972) Approaches to adaptive filtering. *IEEE Transaction on Automatic Control*. 17:903-908.

Odelson BJ, Rajamani MR, Rawlings JB. (2006) A new autovariance least-squares method for estimating noise covariances. *Automatica*. 42:303-308.

Tolman BW. (2008) GPS Precise Absolute Positioning via Kalman Filtering. *ION GNSS 21st*. *International Technical Meeting of the Satellite Division*. 16-19, September 2008, Savannah, GA.