دانشکده ریاضی و علوم کامپیوتر
دانشگاه صنعتی شریف

فیلتر کالمن (Kalman Filter)

گزارش مطالعه و پیاده‌سازی فیلتر کالمن

پروژه درس بهینه‌سازی در علوم داده

دکتر مجتبی تفاق، ترم دوم سال تحصیلی ۱۴۰۱

نگارندگان: محسن قدرتی، محمد صالح بهرامی

۱ مقدمه

در این گزارش قصد داریم الگوریتم فیلتر کالمن را برای بهبود (فیلتر) برآوردهای یک بردار وضعیت، معرفی کنیم و با ارایه تعدادی مدل ساده از دینامیک تغییرات قیمت رمز ارز بیت‌کوین، قدرت فیلتر کالمن در پیش‌بینی آینده را محک بزنیم. در انتها نیز، تعمیم‌هایی از فیلتر کالمن به دینامیک‌های غیر خطی و مساله‌های با چندین مدل موازی را بیان خواهیم کرد.

مطالعات ما در این گزارش، مربوط به نحوه تغییرات میانگین قیمت معاملاتی بیت‌کوین در بازه دو ماهه از دسامبر ۲۰۱۹ الی ژوییه ۲۰۲۰ است که از صرافی بایننس تهیه شده و مربوط به کندل‌های ۵ دقیقه‌ای است. به نظر می‌رسد نتایج نهایی کمابیش در سایر بازارها و یا سایر پارامترهای بازار و سایر بازه‌های زمانی نیز قابل پیاده‌سازی باشد.

۲ تعاریف اولیه

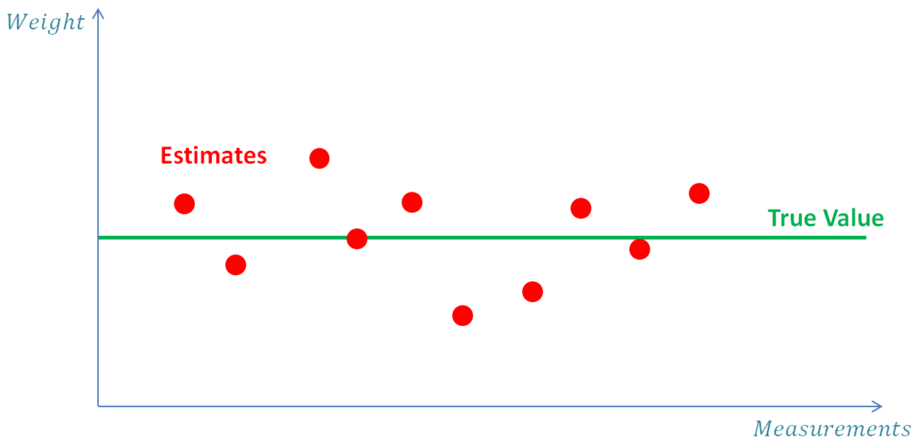
۱.۲ مدل‌های اتوریگرسیو

مدل‌های اتورگرسیو، مدل‌هایی هستند که فرض می‌کنند مقدار یک سری زمانی به تعدادی ثابت از مقادیر قبلی به شکل خطی وابسته است و تفاضل این دو مقدار از یکدیگر یک نویز غیر قابل اندازه‌گیری خواهد بود. به عبارت دیگر، اگر x_t یک سری زمانی باشد، مدل زیر یک مدل اتورگرسیو است:

$$x_t = \sum_{i=1}^{m} x_{t-i} + \epsilon_t$$

۲.۲ فیلتر

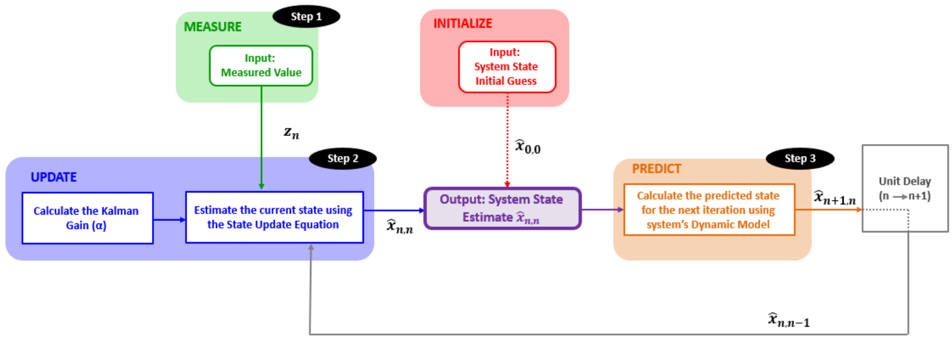
فرض کنید به دنبال اندازه‌گیری وزن یک شمش طلا هستیم و یک ترازو داریم که در اندازه‌گیری خود مقداری خطا دارد و هر بار یک وزن خاص بر می‌گرداند.



با فرض آنکه مانگین خطای ترازوی مورد نظر ما صفر است (نویز سفید داریم)، یک راه موثر برای تخمین وزن واقعی طلا، آن است که چندین بار شمش را وزن کنیم و میانگین وزن‌های یافته شده را برگردانیم. اما برای بروزرسانی میانگین مذکور، وقتی n بار آزمایش انجام شده است و مشاهداتی داشته‌ایم، نیازی به ذخیره‌سازی همگی نتایج نیست. کافی است به نکته زیر توجه کنیم:

$$\bar{x}_n = \frac{x_1 + \cdots + x_n}{n} = \frac{(n-1)\bar{x}_{n-1} + x_n}{n} = \bar{x}_{n-1} + \frac{1}{n}(x_n - \bar{x}_{n-1})$$

یعنی کافیست در گام tام، میانگین را به اندازه ضریبی ($\frac{1}{n}$) در جهت تفاضل مشاهده جدیدمان از این میانگین موجود، تغییر دهیم. به این فرآیند، فیلتر کردن می‌گویند. در واقع با داشتن یک مشاهده و یک تقریب تا لحظه nام، فیلتر کردن یعنی آن که به نحوی این دو مقدار را با یکدیگر ترکیب کنیم که بهترین تقریب جدید ساخته شود. در شکل زیر شمایی از روند تکراری فیلتر کردن یک سری زمانی را مشاده می‌کنیم:



۳ معرفی مساله داده‌کاوی

۱.۳ صورت مساله و داده‌ها

داده‌های این مساله، شامل اطلاعات کندلی ۵ دقیقه‌ای دو ماه از بازار بیت‌کوین در صرافی بایننس است. این اطلاعات شامل میانگین قیمت معاملاتی بیت‌کوین در هر کندل نیز می‌شود. قصد داریم با مدل‌های مختلف فیلتر کالمن، و اطلاعات قیمتی و حجمی تا به یک لحظه خاص، تغییرات قیمتی در ۵ دقیقه بعدی را پیش‌بینی کنیم.

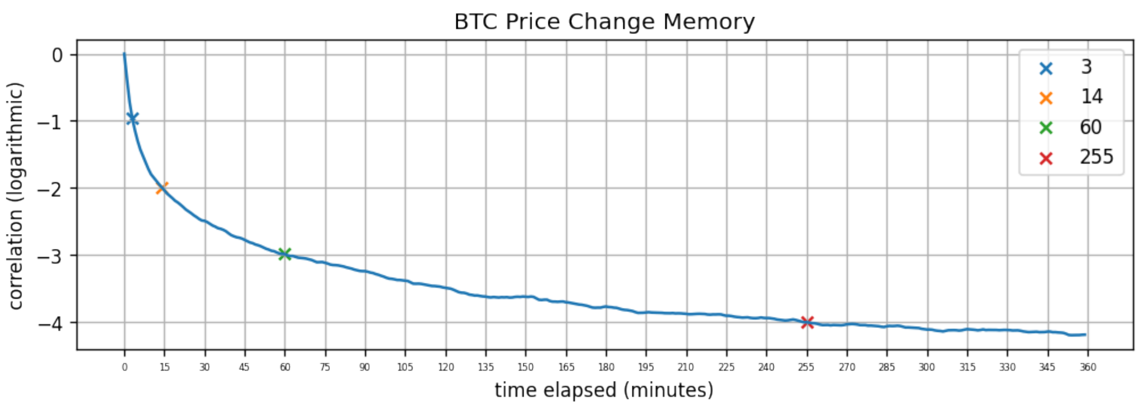
سطرهای اولیه داده‌ها به صورت زیر است:

	Open	High	Low	Close	Volume	Value	No. Trades	Taker Buy Volume	Taker Buy Value	Average Price	Average Price Change
2019-11-27 00:00:00	7154.75	7164.59	7140.54	7162.89	164.200264	1.174323e+06	1234	103.771014	742217.277131	7151.772051	NaN
2019-11-27 00:05:00	7164.4	7169.00	7152.92	7157.86	115.531698	8.275534e+05	904	73.404774	525826.737982	7162.998797	0.001570
2019-11-27 00:10:00	7157.32	7160.19	7144.93	7151.68	112.780186	8.065373e+05	914	57.210353	409185.537990	7151.409684	-0.001618
2019-11-27 00:15:00	7152.16	7152.20	7138.93	7142.66	157.903923	1.128197e+06	833	70.538540	503984.524916	7144.828939	-0.000920
2019-11-27 00:20:00	7142.63	7148.81	7138.94	7146.63	71.999481	5.143852e+05	638	42.162902	301241.148857	7144.290518	-0.000075
...	...	...	...	...	...	...	...	...	...	...	...
2020-02-03 23:35:00	9298.38	9300.35	9277.07	9294.66	148.543688	1.379950e+06	1348	82.834562	769549.767916	9289.856558	-0.000593
2020-02-03 23:40:00	9294.66	9307.69	9294.01	9304.29	90.220836	8.392079e+05	908	64.413847	599153.526012	9301.708836	0.001276
2020-02-03 23:45:00	9304	9319.99	9295.98	9316.82	139.875918	1.301847e+06	1874	68.115588	633991.892739	9307.156778	0.000586
2020-02-03 23:50:00	9316.15	9316.88	9303.73	9306.52	95.824739	8.921500e+05	1331	45.580965	424366.578549	9310.226182	0.000330
2020-02-03 23:55:00	9307.21	9307.73	9291.34	9292.24	127.278443	1.183576e+06	1141	69.671406	647856.534166	9299.110800	-0.001194

19872 rows × 11 columns

۲.۳ تخمین حافظه موثر

حافظه یک مدل اتورگرسیو عبارت است از تعداد مشاهداتی از سری زمانی که برای مدل‌سازی دینامیک مشاهده جدید لازم دارد. در شکل زیر، ضریب همبستگی تغییرات قیمتی بیت‌کوین در یک دقیقه آینده با تغییرات قیمتی از t دقیقه قبل تا این لحظه رسم شده است. همانطور که از شکل مشخص است، برای آنکه حداقل همبستگی $\frac{۱}{۴}$ داشته باشیم، لازم است حافظه را به $m = ۱۴$ محدود کنیم. یعنی رابطه جدی بین زمان‌های دورتر از ۱۴ دقیقه قبل با لحظه فعلی وجود ندارد.



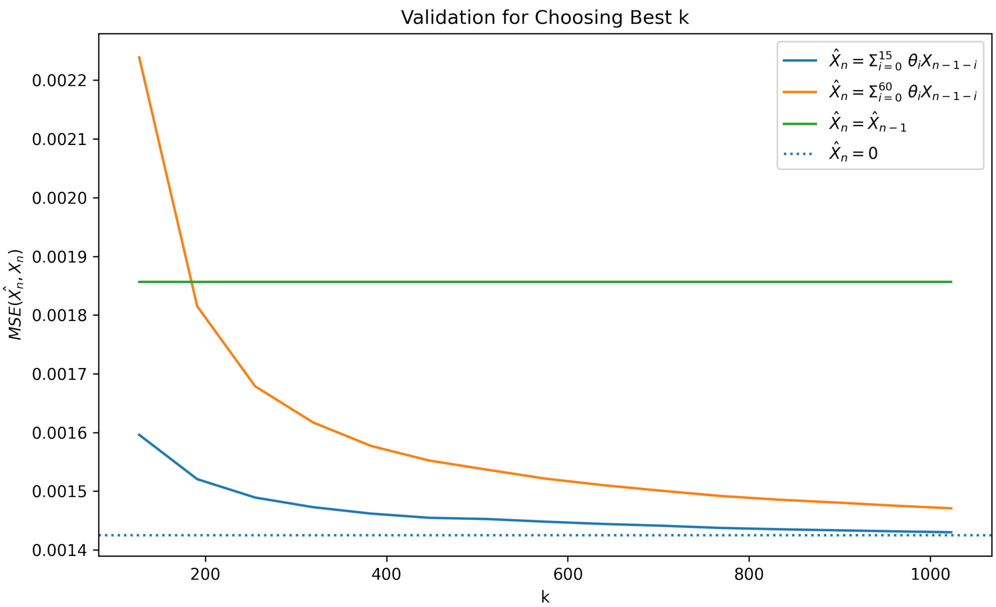
correlation diagram (in binary logarithmic scale) of BTC price return with cummulative price return up to some future point

در ادامه، پیاده‌سازی‌های خود را به دو عدد $m = ۱۵$ و $m = ۶۰$ محدود می‌کنیم.

۳.۳ تخمین سایز مناسب برای یادگیری دینامیک مساله

دینامیک مساله در مدل اتورگرسیو، شامل تعدادی ضریب اسکالر و یک عرض از مبدا ثابت است. در واقع برای یافتن این ضرایب، نیاز به حل یک معادله رگسیون خطی داریم. اما اینکه تا چند نمونه برای یافتن ضرایبی با دقت بالا (p_value ناچیز) لازم است، خود به validation اختیاج دارد. در شکل زیر، واریانس نویز مدل‌های خطی بدون کنترل، به ازای تعداد نمونه استفاده شده در یادگیری ضرایب دینامیک مدل رسم شده است. مدل آبی به ازای حافظه $m = ۱۵$ و مدل نارنجی به ازای حافظه $m = ۶۰$ تولید شده است. همانطور که از شکل پیداست، هنگامی که حافظه افزایش می‌یابد، واریانس نویز سیستم خطی نیز افزایش می‌یابد. یعنی حافظه خیلی بزرگ معادل است با عدم قطعیت بیشتر در پیش‌بینی تغییرات قیمت بیت‌کوین. هم‌چنین توجه می‌کنیم که هر دو مدل تقریبا به ازای همه انتخاب‌های k از مدلی که تغییرات قیمتی بیت‌کوین را برابر با تغییرات قیمتی در لحظه قبل می‌داند، بهتر هستند. اما از طرفی هیچ‌کدام از مدلی که تغییرات قیمتی را صفر پیش‌بینی می‌کند بهتر نیستند. دلایل اینکه از صفر پیش‌بینی کردن بهتر

نیستیم، زیاد است و به تحلیل اقتصادی در کنار تحلیل داده‌ها نیاز دارد. اما تعدادی از عمده‌ترین دلایل چنین پیش‌بینی‌ناپذیریی عبارتند از اینکه اول از همه، دینامیک تغییرات قیمت در بازارهای کارایی مانند بیت‌کوین، به شدت غیرخطی و غیریکنواخت است. دوم اینکه در بسیاری از زمان‌ها، به دلیل عدم وجود اتفاق نظر در بین اهالی بازار، رفتار قیمت تصادفی یا بسیار سبه تصادفیست. پس مدلی که بخواهد در همه زمان‌ها تغییرات قیمت را پیش‌بینی کند، ناچار است بسیاری از نمونه‌هایش را از متغیرهایی بگیرد که کاملا نرمال و تصادفی متقارن هستند. در نتیجه بهتر از توزیعی که نمونه‌هایش داراست، نمی‌تواند پیش‌بینی انجام دهد. یعنی همان پیش‌بینی $\hat{X}_n = 0$.



۴.۳ تهیه تعدادی ویژگی (کنترل دینامیک)

۴ فیلتر کالمن

۱.۴ تعریف

۲.۴ الگوریتم

۳.۴ تحلیل زمانی

۴.۴ پیاده‌سازی

۱.۴.۴ مدل ۱: دینامیک خطی بدون کنترل

فرض می‌کنیم دینامیک تغییرات قیمت به شکل زیر باشد:

$$\tilde{r}_n = f_{0,n}\tilde{r}_{n-1} + \sum_{i=1}^{m-1} f_{i,n} r_{i,n-1} + f_{m,n} + w_n, \qquad w_n \sim \mathcal{N}(0,\ \sigma_n^2). \tag{1}$$

به عبارت دیگر، فرض می‌کنیم اگر تغییرات قیمت از $i = 2, \ldots, m-1$ کندل گذشته تا کندل گذشته را بدانیم و تغییرات قیمت از کندل گذشته تا کندل فعلی را نیز بدانیم، بازدهی پیش رو ترکیبی خطی از این مقادیر به علاوه یک نویز با توزیع نرمال است. در این صورت برای بردار $\mathbf{x}^{(\mathbf{n})}$ فرم معادل زیر را خواهیم داشت:

$$\mathbf{x}^{(\mathbf{n})} = F_n \mathbf{x}^{(\mathbf{n-1})} + \mathbf{w}^{(\mathbf{n})}, \tag{2}$$

که در آن

$$\mathbf{x^{(n)}} := \begin{bmatrix} \tilde{r}_n \\ r_{1,n} \\ \vdots \\ r_{m-1,n} \\ 1 \end{bmatrix}, \quad F_n := \begin{bmatrix} f_{0,n} & f_{1,n} & \cdots & f_{m,n} \\ 1+\tilde{r}_{n-1} & 0 & \ddots & 0 \\ 0 & 1+\tilde{r}_{n-1} & 0 & \vdots \\ 0 & \ddots & & 0 \\ \vdots & 0 & 1+\tilde{r}_{n-1} & 0 \\ 0 & \cdots & 0 & 1 \end{bmatrix}, \quad \mathbf{w^{(n)}} := \begin{bmatrix} w_n \\ -\tilde{r}_{n-1}^2 \\ \tilde{r}_{n-1} \\ \vdots \\ \tilde{r}_{n-1} \\ 0 \end{bmatrix}.$$

با توجه به این که در هر لحظه n، بردار $\mathbf{x^{(n-1)}}$ معلوم است و کواریانس بردار $\mathbf{w^{(n)}}$ تنها در درایه $1,1$ ناصفر و برابر با σ_n^2 است، به دنبال مقادیری از $f_{0,n},\ \ldots,\ f_{m-1,n}$ می‌گردیم که σ_n^2 را کمینه کند. (در واقع خطای مدل را در حد امکان کاهش دهد.) هم‌چنین توجه می‌کنیم که اضافه کردن عدد ۱ به انتهای بردار $\mathbf{x^{(n)}}$ باعث می‌شود بتوان عرض از مبدا مدل ۱ را به $f_{m,n}$ منتقل کرد و لذا فرض صفر بودن میانگین نویز w_n برقرار است. اکنون سعی می‌کنیم با بازنویسی معادله بالا، فرم مجموع مربعاتی برای یافتن بهترین $f_{i,n}$ها ارایه کنیم.

اگر قرار دهیم $\theta_n = (f_{0,n},\ \ldots,\ f_{m,n})^T$، به دنبال حل مساله بهینه‌سازی زیر هستیم:

$$minimize_{(\theta_n)} \quad |\tilde{r}_n - \langle \theta_n,\ \mathbf{x^{(n-1)}} \rangle|^2$$

اما با توجه به اینکه مساله بالا احتمالا بی‌نهایت جواب دارد و این موضوع باعث تغییرات سریع دینامیک مدل در هر لحظه می‌شود که خلاف طبیعت مدل است، و مهم‌تر آنکه انتظار داریم دینامیک مدل در طول یک پنجره کوتاه مدت (kتایی) پایدار باشد (یعنی $\theta_n \simeq \theta_{n-1} \simeq \cdots \simeq \theta_{n-(k-1)}$) و بتوانیم از اطلاعات این پنجره برای تقریب یک دینامیک کم‌نوسان بهره ببریم، مساله بهینه‌سازی فوق را به مساله زیر تعمیم می‌دهیم:

$$minimize_{(\theta_n)} \quad |\tilde{r}_{n-(k-1)} - \langle \theta_n,\ \mathbf{x^{(n-1-(k-1))}} \rangle|^2 + \cdots + |\tilde{r}_n - \langle \theta_n,\ \mathbf{x^{(n-1)}} \rangle|^2$$

که با در نظر گرفتن ماتریس $X^{(n)}$ به صورت

$$X^{(n)} = \begin{bmatrix} \mathbf{x^{(n-(k-1))}}^T \\ \vdots \\ \mathbf{x^{(n)}}^T \end{bmatrix} \quad \rightarrow \quad X^{(n)} := \begin{bmatrix} \mathbf{b_0^{(n)}} & \ldots & \mathbf{b_m^{(n)}} \end{bmatrix}$$

به فرم زیر قابل بیان است:

$$for\ given\ n: \quad minimize_{(\theta_n)}\ ||X^{(n-1)}\theta_n - \mathbf{b_0^{(n)}}||_2^2 \tag{۳}$$

توجه به این نکته ضروری است که از بردار $X_0^{(n)}$، درایه اول مربوط به بازدهی آینده است و در درسترس نیست. لذا به جای حل مساله فوق برای n آن را برای $n-1$ حل می‌کنیم.

در ادامه به بررسی نتایج حاصل از حل مساله بهینه‌سازی ۳ به کمک روش کمترین مجموع مربعات و پیاده‌سازی فیلتر کالمن بر روی مدل ۱ می‌پردازیم.

مشاهده ۱. مطابق با نمودار زیر می‌توان فرض نرمال بودن نویز در اندازه‌گیری و مدل را تا حدودی تایید کرد. هم‌چنین توجه می‌کنیم که میانگین توزیع نویز، با دقت ۷ رقم اعشار صفر خواهد بود.

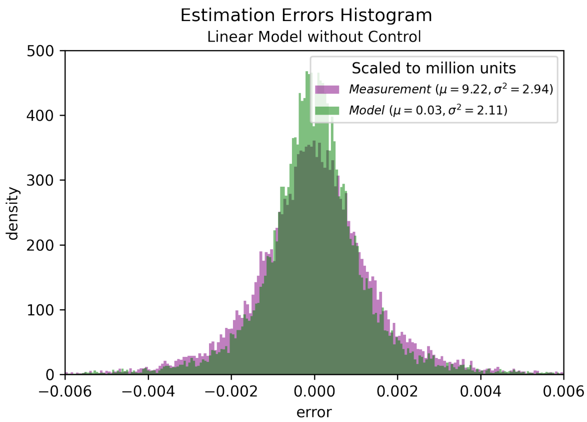
مشاهده ۲. همان‌طور که اشاره کردیم، فرض می‌کنیم دینامیک مدل دچار تغییرات سریع نمی‌شود. این نکته در نمودار زیر به صورت طیف‌های پیوسته قابل مشاهده است. هم‌چنین در نمودار زیر این نکته قابل توجه وجود دارد که ضرایب بزرگ‌تر و تاثیرگذارتر در مدل خطی پیشنهاد شده، به لحظات نزدیک‌تر به لحظه حال مرتبط می‌شوند. این نیز مشاهده‌ای است که انتظارش را داشتیم.

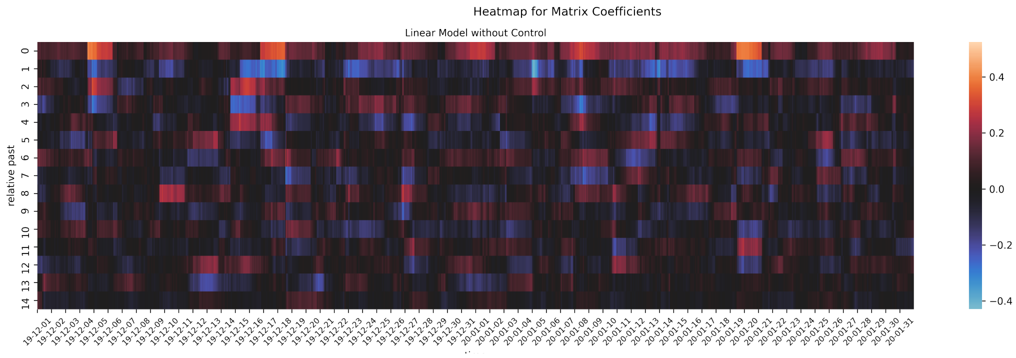
نتیجه ۳. در شکل می‌توان خروجی فیلتر کالمن را بر روی مدل خطی در هر لحظه به ازای یک پنجره خاص زمانی مشاهده کرد.

۲.۴.۴ مدل ۲: دینامیک خطی همراه با کنترل

در مدل زیر، فرض می‌کنیم تعدادی ویژگی (که در بخش قبل تهیه شده‌اند و در همان بخش، همبستگیشان با تابع هدف ما بررسی و تایید شده است) نیز علاوه بر بردار $\mathbf{x^{(n-1)}}$، در دینامیک تغییرات قیمت به شکل خطی موثر باشند. به عبارت دیگر دینامیک مساله را با مدل زیر تقریب می‌زنیم:

$$\tilde{r}_n = f_{0,n}\tilde{r}_{n-1} + \sum_{i=1}^{m-1} f_{i,n} r_{i,n-1} + f_{m,n} + \sum_{i=0}^{s-1} g_{i,n} u_{i,n} + w_n, \qquad w_n \sim \mathcal{N}(0,\ \sigma_n^2). \tag{۴}$$





که با فرم زیر معادل است:

$$\mathbf{x^{(n)}} = F_n \mathbf{x^{(n-1)}} + G_n \mathbf{u^{(n)}} + \mathbf{w^{(n)}}, \tag{۵}$$

و در آن $\mathbf{u^{(n)}}$ بردار ویژگی‌ها در لحظه n است. در واقع:

$$G_n := \begin{bmatrix} g_{0,n} & \cdots & g_{s-1,n} \\ 0 & \cdots & 0 \\ & \ddots & \\ 0 & \cdots & 0 \end{bmatrix}, \quad \mathbf{u^{(n)}} := \begin{bmatrix} u_{0,n} \\ \vdots \\ u_{s-1,n} \end{bmatrix}.$$

با در نظر گرفتن ماتریس $X^{(n)}$ و بردار $\mathbf{b_0^{(n)}}$ مدل قبلی، و تعریف ماتریس U_n و بردار θ'_n به شکل زیر،

$$U_n := \begin{bmatrix} \mathbf{u^{(n-(k-1))}}^T \\ \vdots \\ \mathbf{u^{(n)}}^T \end{bmatrix}, \qquad \theta'_n := \begin{bmatrix} g_{0,n} \\ \vdots \\ g_{s-1,n} \end{bmatrix}$$

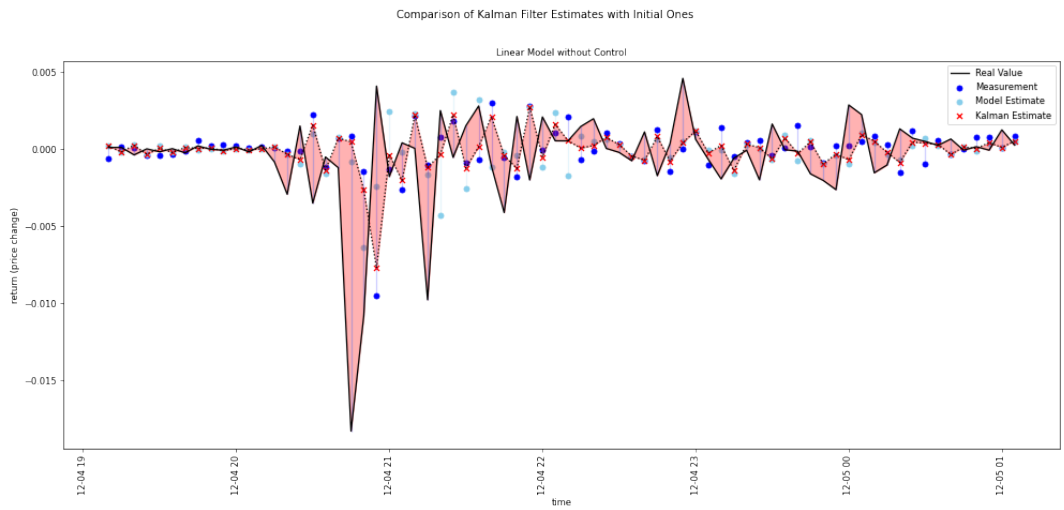
می‌توان به سادگی و مشابه با روند مدل ۱، به مساله کمترین مجموع مربعات زیر دست یافت:

$$for\ given\ n: \quad minimize_{(\theta_n, \theta'_n)}\ ||X^{(n-1)}\theta_n + U_n\theta'_n - \mathbf{b_0^{(n)}}||_2^2 \tag{۶}$$

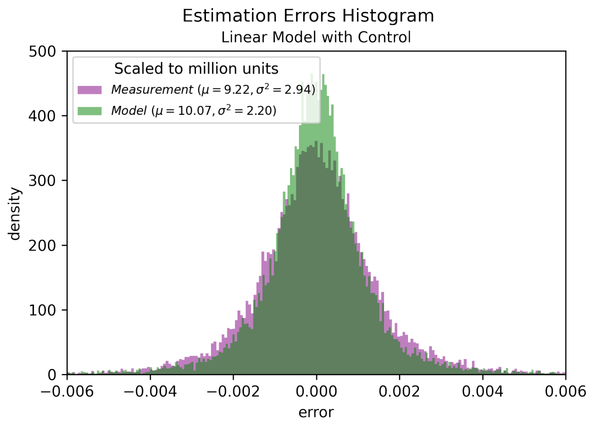
اکنون با ادغام دو ماتریس $X^{(n-1)}$ و U_n و ادغام دو بردار θ_n و θ'_n مساله بهینه‌سازی معادل زیر را داریم:

$$A_n := \begin{bmatrix} X^{(n-1)} & U_n \end{bmatrix}, \quad \theta^*_n := \begin{bmatrix} \theta_n \\ \theta'_n \end{bmatrix} \qquad \longrightarrow \qquad for\ given\ n: \quad minimize_{(\theta^*_n)}\ ||A_n\theta^*_n - \mathbf{b_0^{(n)}}||_2^2$$

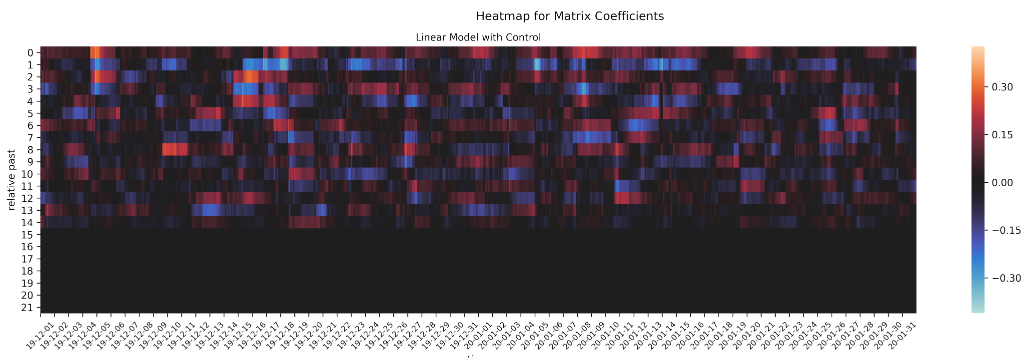
در ادامه به بررسی نتایج حاصل از حل مساله بهینه‌سازی ۶ به کمک روش کمترین مجموع مربعات و پیاده‌سازی فیلتر کالمن بر روی مدل ۴ می‌پردازیم.



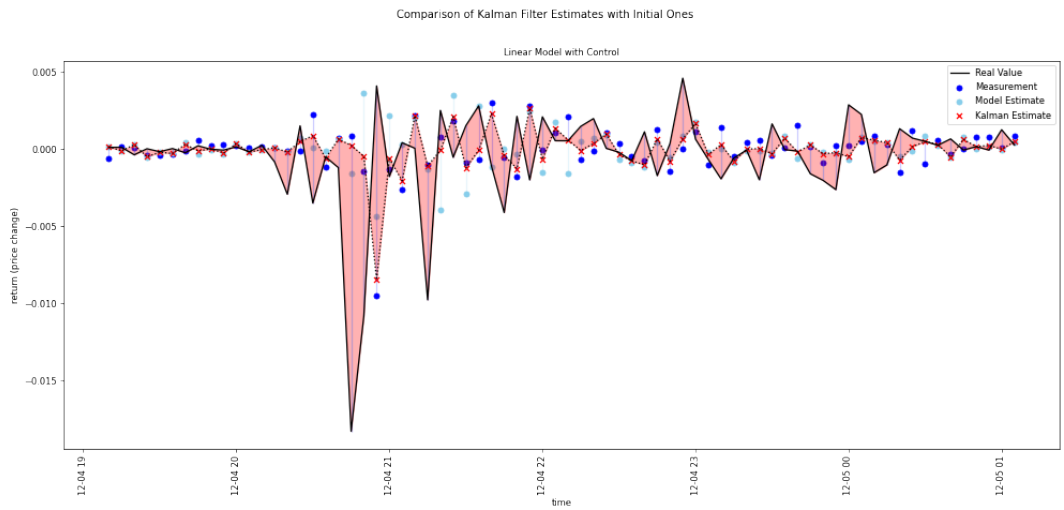
مشاهده ۴. مطابق با نمودار زیر می‌توان فرض نرمال بودن نویز در اندازه‌گیری و مدل را مجددا تا حدودی تایید کرد. هرچند همانطور که از نمودار پیداست، واریانس مدل کمی افزایش (جزیی) داشته و کم نشده است! در بخش‌های بعدی، زمانی که همه مدل‌های خطی را معرفی کردیم و به مقایسه عملکرد آن‌ها پرداختیم، دلیل عملیاتی این موضوع را توضیح خواهیم داد.



مشاهده ۵. نمودار زیر مشابه با نمودار مربوط به مدل ۱ قابل قبول است. نکته حایز اهمیت در این نمودار، تباهیده شدن ضرایب مربوط به ویژگی‌هاست که در واقع به دلیل انحراف معیار بالاتر ستون‌های مربوط به ویژگی‌ها نسبت به ستون‌های مربوط به بازدهی است و اطلاعاتی از دست نرفته است.



نتیجه ۶. در شکل زیر می‌توان خروجی فیلتر کالمن را بر روی مدل خطی در هر لحظه به ازای یک پنجره خاص زمانی مشاهده کرد.



۳.۴.۴ مدل ۳: دینامیک خطی حساس به زمان

منظور از مدل حساس به زمان، مدلی است که به نوعی به اطلاعات اخیرتر بیشتر توجه دارد (مثلا توسط یک ماتریس وزن‌دهی). همانطور که می‌توان حدس زد، چنین مدلی اساسا واریانس نویز بیشتری دارد. اما در عوض ممکن است به کمک فیلتر کالمن بتوان نویز آن را بهتر کنترل کرد، چرا که چنین مدلی در صورت خطی بودن واقعیت دینامیک مساله، سریع‌تر بروزرسانی می‌شود.

به دو روش می‌توان حساسیت به زمان را در مدل منعکس کرد. یکی حساسیت در یادگیری پارامترها، و دیگری حساسیت در توجه به اطلاعات سری زمانی.

روش اول معادل است با مساله بهینه‌سازی زیر:

$$for\ given\ n: \quad minimize_{(\theta_n^*)}\ ||D_1 A_n \theta_n^* - D_1 \mathbf{b}_0^{(\mathbf{n})}||_2^2, \quad D_1 = \frac{1}{\lambda} diag(\alpha^0,\ \dots,\ \alpha^{k-1}),\ \lambda = \sum_{i=0}^{k-1} \alpha^i \qquad (7)$$

که در آن α پارامتر تاثیرگذاری است و مقداری حداقل برابر با ۱ دارد. پارامتر α در واقع بیان می‌کند کوچک کردن هر مقدار باقیمانده w_j در مساله مجموع کمترین مربعات، چه اندازه از کوچک کردن مقدار باقیمانده w_{j-1} مهم‌تر است.

روش دوم معادل است با وزن‌دهی به درایه‌های بردار θ_n به گونه‌ای که هرچه اندیس i در $f_{i,n}$ افزایش می‌یابد، $|f_{i,n}|$ کاهش یابد. پس باید وزن‌های بزرگ‌تری به $f_{i,n}$های با اندیس بزرگ‌تر در مساله بهینه‌سازی ۶ بدهیم. یعنی مساله بهینه‌سازی زیر:

$$for\ given\ n: \quad minimize_{(\theta_n^*)}\ |\begin{bmatrix} X^{(n-1)} D_2 & U_n \end{bmatrix} \theta_n^* - \mathbf{b}_0^{(\mathbf{n})}||_2^2, \quad D_2 = \frac{1}{\lambda} diag(\alpha^0,\ \dots,\ \alpha^m),\ \lambda = \sum_{i=0}^{m} \alpha^i \qquad (8)$$

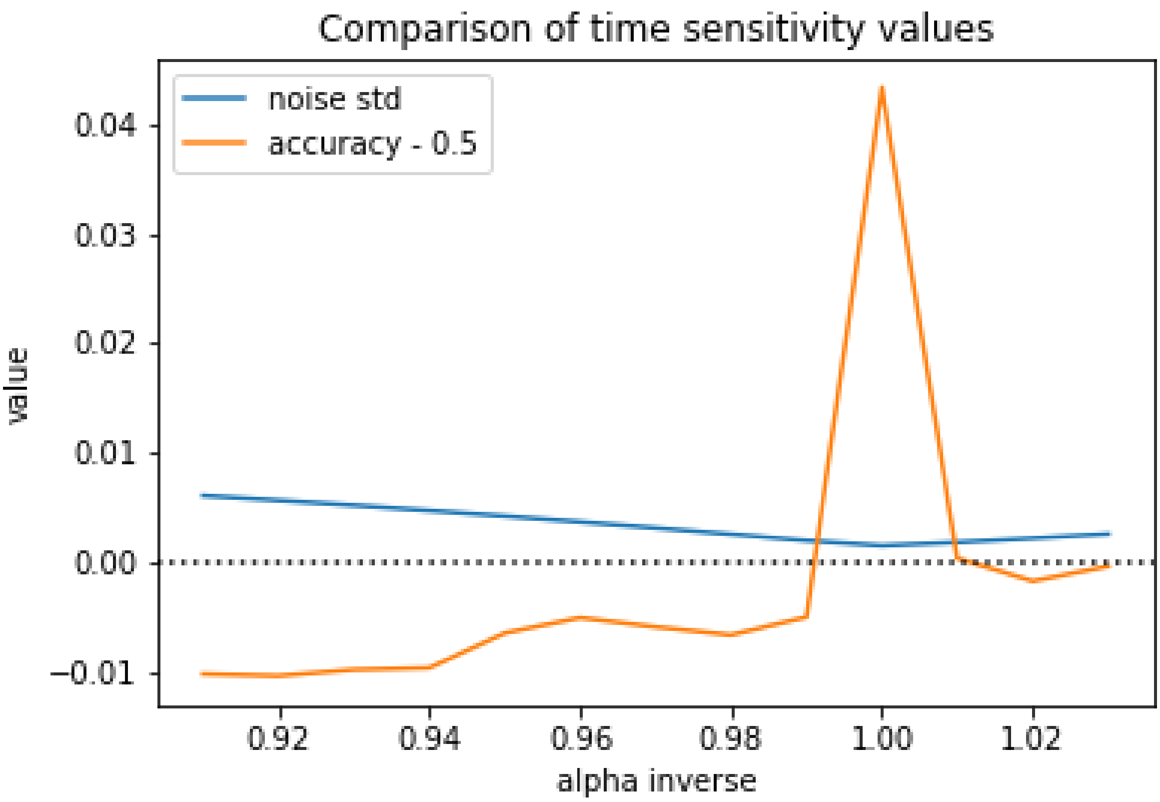
در ادامه تنها نتیجه حل مساله بهینه‌سازی ۷ آمده است:

مشاهده ۷. ابتدا به ازای مقادیر مختلف α با بررسی دقت پیش‌بینی (صرفا درستی علامت $\tilde{r}_n$) و انحراف معیار نویز مدل، بهترین α را انتخاب می‌کنیم:

بر اساس شکل در می‌یابیم نقاطی که در پیش‌بینی جهت $\tilde{r}_n$ بیشترین دقت را دارند (از تشخیص تصادفی فاصله دارند و در عین حال انحراف معیار نویزشان کم است) عبارت‌اند از ۰٫۹۴ و ۱. از طرفی کمترین انحراف معیار نویز، همانطور که پیش بینی شده بود، در حوالی ۱ رخ می‌دهد. پس اگر به دنبال مقداری غیر بدیهی (مخالف ۱) هستیم، $\alpha = ۰٫۹۴$ گزینه مناسبی است. (توجه می‌کنیم که نیمه عمر چنین انتخابی حدود ۱۲ است، یعنی حساسیت مدل به داده هر لحظه حدود دو برابر حساسیت آن به داده ۱۲ کندل قبل است!)

هم‌چنین در ادامه تصویری از هیستوگرام نویز تعدادی از انتخاب‌های α قابل مشاهده است.

نتیجه ۸. در شکل زیر می‌توان خروجی فیلتر کالمن را بر روی مدل خطی حساس به زمان (با $\alpha^{-1} = ۰٫۹۴$) در هر لحظه به ازای یک پنجره خاص زمانی مشاهده کرد.



۴.۴.۴ مدل ۴: دینامیک خطی با ماتریس چگال

در مدل‌های قبل، برای سهولت در محاسبه و سادگی مدل، فرض کردیم ماتریس‌های F_n و G_n تنک هستند. در این مدل، بدون تغییر فرض خطی بودن دینامیک مساله، فرض تنک بودن F_n و G_n را حذف می‌کنیم. در این صورت هرچند واریانس نویز تقریب آخرین بازدهی افزایش می‌یابد، اما با توجه به اینکه حالت تنک معادله خطی زیر، یک حالت خاص از معادله کلی است، انتظار داریم مساله بهینه‌سازی بتواند جواب‌های پایدارتری تولید کند. فایده دستگاه‌های خطی با واریانس نویز w_n بالاتر ولی پایداری بیشتر ان است که در بازارهای مالی به طور طبیعی مقدار بسیار زیادی نویز غیر قابل کنترل وجود دارد، و با پایدارتر کردن مدل، به ذات دینامیک مساله بهتر دسترسی داریم. (با فرض شبیه به خطی بودن این دینامیک)

$$\mathbf{x^{(n)}} = F_n\mathbf{x^{(n-1)}} + G_n\mathbf{u^{(n)}} + \mathbf{w^{(n)}}$$

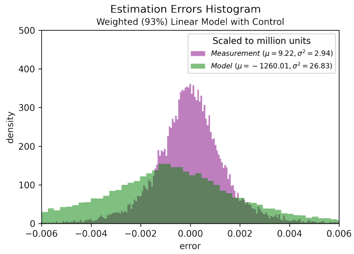
هم‌چنین در مدل بالا، فرض کنیم یک بردار آفست $\mathbf{h^{(n)}}$ نیز در هر لحظه به سمت راست معادله اضافه می‌شود و در عوض $\mathbf{w^{(n)}}$ توزیع نرمال با میانگین ۰ دارد. در این صورت مساله بهینه‌سازی یافتن بهترین پارامترها برای تقریب خطی دینامیک مساله، در فرم کلی خود به صورت زیر است:

$$for\ given\ n: \qquad minimize_{(F_n,\, G_n,\, \mathbf{h^{(n)}})} \quad \left|\left| \begin{bmatrix} F_n & G_n & diag(\mathbf{h^{(n)}}) \end{bmatrix} \begin{bmatrix} \mathbf{x^{(n-1)}} \\ \mathbf{u^{(n)}} \\ \mathbf{1_{m\times 1}} \end{bmatrix} - \mathbf{x^{(n)}} \right|\right|_2^2 \tag{9}$$

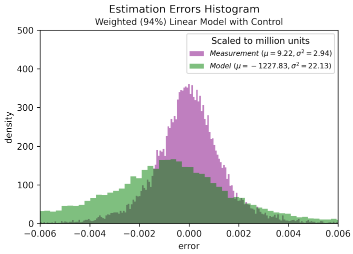
که با ترانهاده گرفتن از دو عبارت ماتریسی، و تکرار آنچه در مورد مدل ۱ بیان شد، به فرم زیر رسید:

$$for\ given\ n: \qquad minimize_{(F_n,\, G_n,\, \mathbf{h^{(n)}})} \quad \left|\left| \begin{bmatrix} X^{(n-1)} & U_n & \mathbf{1_{k\times m}} \end{bmatrix} \begin{bmatrix} F_n^T \\ G_n^T \\ diag(\mathbf{h^{(n)}}) \end{bmatrix} - X^{(n)} \right|\right|_2^2 \tag{10}$$

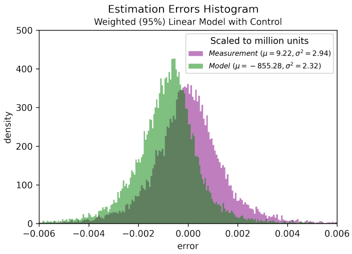
مساله بهینه‌سازی فوق که تعمیم کمترین مجموع مربعات عادی به ماتریس مجهول است، در تمرین ۱۳.۱۵ کتاب vmls بررسی شده است. در واقع هر ستون ماتریس مجهول به شکل جداگانه با روش کمترین مجموع مربعات یافته می‌شود و سپس کل ماتریس را از کنار هم قرار دادن این ستون‌ها معرفی می‌کنیم. توجه می‌کنیم که پس از یافتن ماتریس مجهول، در واقع ستون اول به تنهایی برای تشکیل دینامیک خطی همراه با کنترل لازم است. اما کل ماتریس را برای محاسبه ماتریس‌های کوواریانس در هر مرحله از الگوریتم کالمن ذخیره می‌کنیم.



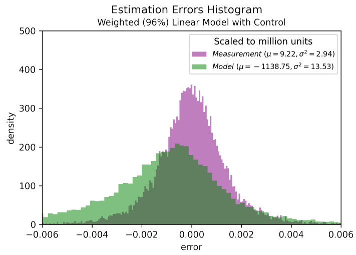
$\alpha^{-1} =$ ۰٫۹۳ (آ)



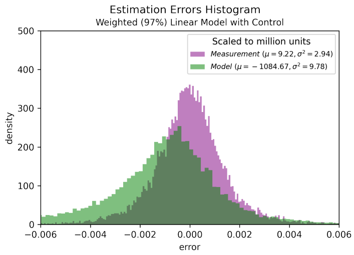
$\alpha^{-1} =$ ۰٫۹۴ (ب)



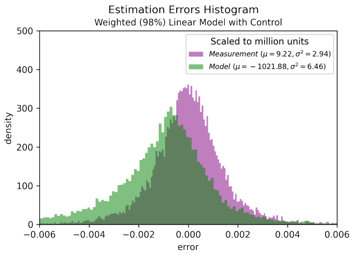
$\alpha^{-1} =$ ۰٫۹۵ (ج)



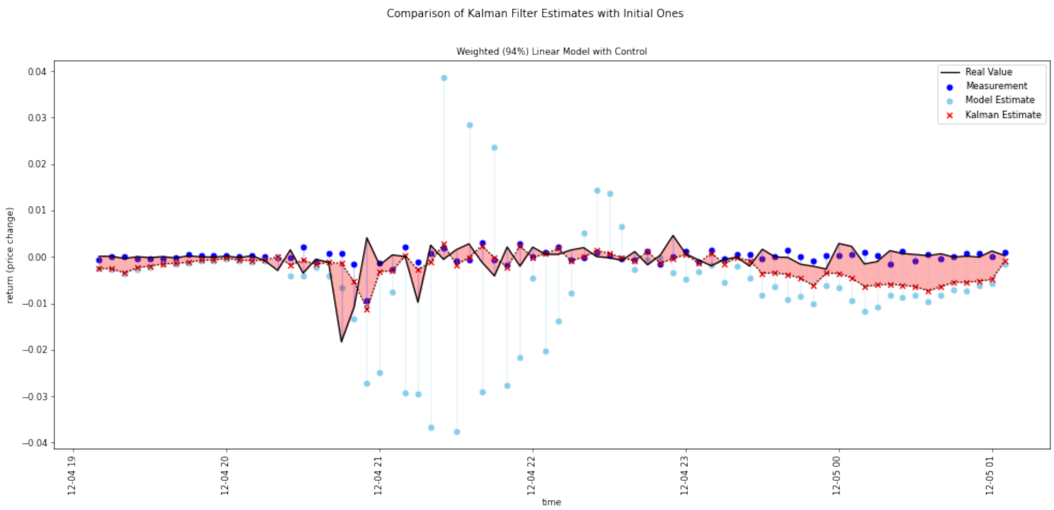
$\alpha^{-1} =$ ۰٫۹۶ (د)



$\alpha^{-1} =$ ۰٫۹۷ (ه)



$\alpha^{-1} =$ ۰٫۹۸ (و)



۵ فیلتر کالمن تعمیم‌یافته

۱.۵ تعریف

۲.۵ پیاده‌سازی

۱.۲.۵ مدل ۵: دینامیک درجه ۲ بدون کنترل

۲.۲.۵ مدل ۶: دینامیک درجه ۲ همراه با کنترل خطی

۳.۲.۵ مدل ۷: دینامیک شبکه عصبی

۴.۲.۵ مدل ۸: دینامیک مشتق‌ناپذیر

۶ فیلتر کالمن سریع

۷ فیلتر کالمن با چند مدل

۱.۷ جمع‌بندی مدل‌های ارایه شده

۲.۷ تعمیم فیلتر کالمن به چند مدل موازی

۱.۲.۷ تعمیم ضرایب کالمن

۲.۲.۷ ترکیب چند فیلتر

۸ ارجاع و منابع

[?]