

# 《机器学习》课程系列

卡尔曼滤波基础\*

Preliminaries on Kalman Filtering

武汉纺织大学数学与计算机学院

杜小勤

2020/09/15

## Contents

1	概述	2
2	卡尔曼滤波的算法推导	4
3	滤波算法的通用形式	12
3.1	标量情形 . . . . .	13
3.1.1	2 个标量情形 . . . . .	13
3.1.2	多个标量情形 . . . . .	15
3.1.3	增量融合 . . . . .	17
3.2	向量情形 . . . . .	18
4	练习	21
5	参考文献	21

---

\*本系列文档属于讲义性质，仅用于学习目的。Last updated on: November 6, 2020。

## 1 概述

现代滤波器理论始于 N. Wiener 在 1940 年代的工作，其理论基础是均方误差 (Mean Squared Error) 的最小化，因而基于这种理论的滤波器也被称为最小二乘方 (Least-Square) 滤波器。实际上，这种滤波器的全称为线性时域最小均方误差滤波器 (Linear Time-Domain Minimum Mean-Square Error Filter)。不论其名称如何，滤波器的任务是，从带有加性噪声的信号中恢复出原始信号。然而，Wiener 滤波器因其存在的固有缺点，在实际中鲜有应用。

卡尔曼滤波器 (Kalman Filter) 于 1960 年由 Rudolf E. Kalman 提出，是一种高效率的递归滤波器 (自回归滤波器)，它能够从一系列的不完全及包含噪声的测量中，估计出动态系统的状态。

由于卡尔曼滤波器能够从带噪声的数据中抽取出有用的信息，且具有较小的计算与空间需求，因而被广泛应用于飞机与太空船的导航 (Navigation)、无人机导航与控制系统、全球定位系统接收器 (Global Positioning System Receiver)、无线电设备的锁相环 (Phase-Locked Loop)、机器人运动规划、计算机的控制系统、计算机游戏、信号处理及无线传感器网络等应用中。早期最著名的卡尔曼滤波器是 Apollo 登月飞船的导航系统，成功地帮助宇航员 Neil Armstrong 登上月球并返回。

实际上，卡尔曼滤波器由一套数学方程组成——它以最小化均方误差为优化目标，以递推或递归 (Recursion) 的形式，提供了一套估算动态系统状态的计算公式。该滤波器功能非常强大，能够实现过去、现在与将来的状态实时估算。

总体上，卡尔曼滤波算法分为 2 步，它们分别是预测步 (Prediction Step) 与更新步 (Update Step)。在预测阶段，算法估算动态系统的当前状态以及相应的不确定性 (Uncertainty)<sup>1</sup>。在观测到新数据后，算法将对预测阶段得到的状态估算进行带权值的更新——具有较高确定性的量，其权值也较高，状态更新时发挥的作用也越大。卡尔曼滤波算法能够以实时迭代的方式运行，估算时仅需前一阶段的状态数据。算法的 2 阶段运行示意图如图 1-1 所示。

在图 1-1 中，卡尔曼滤波器负责跟踪并更新系统的状态  $\hat{\mathbf{x}}_{k|k}$  与状态误差协方

---

<sup>1</sup>粗略地说，随机变量的不确定性程度可以使用方差 (Variance) 来表示：方差越小，不确定性程度越小，表明智能体对自己的判断越自信 (Confidence)；反之，不确定性程度越大，表明智能体对自己的判断越不自信。对于标量形式，方差的倒数被称为精度 (Precision)；对于矩阵形式，方差的逆被称为精度矩阵 (Precision Matrix) 或信息矩阵 (Information Matrix)。

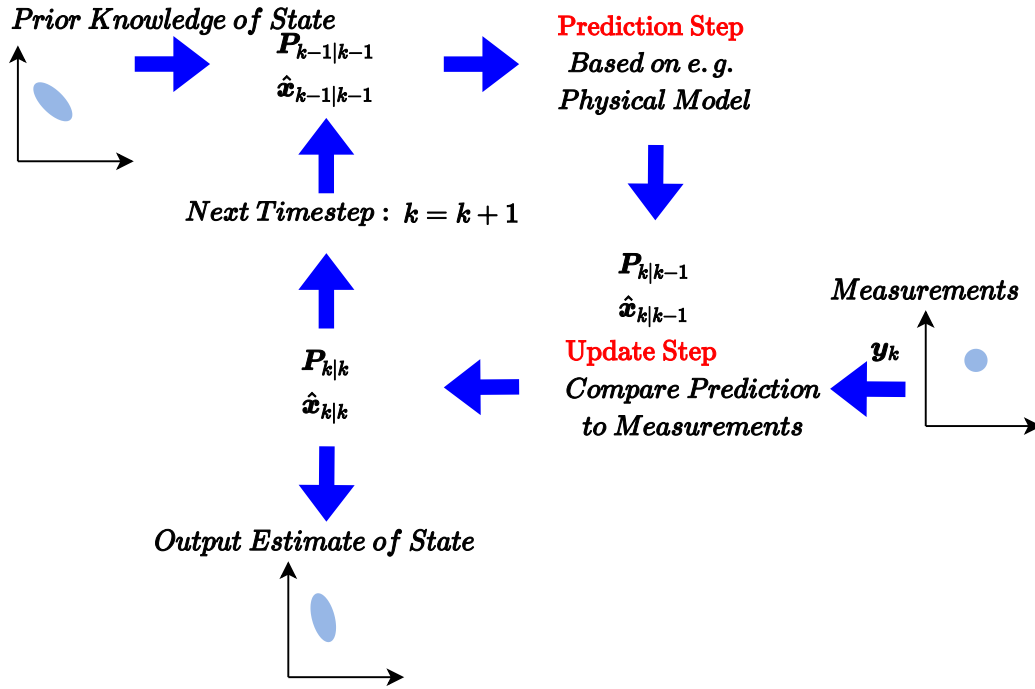


图 1-1: 卡尔曼滤波算法的 2 阶段示意图

差矩阵  $P_{k|k}$ <sup>2</sup>: 在预测阶段, 利用上一阶段的  $\hat{x}_{k-1|k-1}$  和  $P_{k-1|k-1}$ , 估算出本阶段的先验状态  $\hat{x}_{k|k-1}$  和先验协方差矩阵  $P_{k|k-1}$ ; 然后, 在更新阶段, 利用观测 (Observation) 或测量 (Measurement) 数据  $y_k$ <sup>3</sup>, 对先验状态  $\hat{x}_{k|k-1}$  进行更新, 同时也更新先验协方差矩阵  $P_{k|k-1}$ , 从而得到本阶段的后验状态  $\hat{x}_{k|k}$  与后验协方差矩阵  $P_{k|k}$ 。于是, 本轮迭代结束, 算法进入新一轮迭代。上述过程, 循环往复, 每次都对动态系统的状态及协方差矩阵进行有效的更新。从过程的角度而言, 卡尔曼滤波迭代算法, 可以分为状态过程 (State Process) 与观测过程 (Observation Process) 或测量过程 (Measurement Process), 两者交替作用, 并进行有效的信息融合。信息融合所得到的状态预测效果, 要优于任一单方过程所能达到的状态预测效果。

尽管卡尔曼滤波器无需假设误差服从高斯分布, 但是在状态过程与观测过程的协方差矩阵为已知且误差服从高斯分布的情况下, 从均方误差的最优化角度而言, 卡尔曼滤波器是最佳的线性估算器。

<sup>2</sup>状态误差协方差矩阵  $P_{k|k}$  可以反映出智能体对状态估算的不确定性程度, 并据此调整估算状态与观测数据之间的更新权重。

<sup>3</sup>无论是状态数据, 还是观测数据, 均包含一定程度的随机噪声。

对于非线性动态系统，已经提出了扩展卡尔曼滤波器 (Extended Kalman Filter, EKF) 和无迹卡尔曼滤波器 (Unscented Kalman Filter, UKF)。

EKF 的基本思想是，将非线性系统线性化，然后进行常规的卡尔曼滤波。具体地，EKF 对非线性函数进行 Taylor 展开，并进行一阶线性化截断，即忽略高阶项，从而将非线性问题转化为线性问题。然而，EKF 具有如下的缺点：

- 对于强非线性系统，Taylor 展开式中被忽略的高阶项将会带来较大的误差，可能会导致滤波发散；
- EKF 在线性化时需要使用雅可比矩阵 (Jacobian Matrix)，其繁琐的计算过程将会导致算法实现上的相对困难；

在满足线性系统、高斯白噪声、随机变量服从高斯分布这 3 个假设条件时，EKF 是最小方差准则下的次优滤波器，其性能依赖于局部非线性度。

UKF 是一种新型的滤波估计算法，它以无迹 (UT) 变换为基础，摒弃了 EKF 中非线性函数线性化的传统做法——采用卡尔曼线性滤波框架，对于一步预测方程，使用无迹变换来处理均值和协方差的非线性传递。UKF 对非线性函数的概率密度分布进行近似，用一系列确定样本来逼近状态的后验概率密度。UKF 没有对非线性函数进行线性化近似，因而不需要求导计算雅可比矩阵，也不存在线性化过程中忽略高阶项的问题，因而计算精度较高。基于上述优点，UKF 已经被广泛应用于导航、目标跟踪、信号处理以及神经网络等多个领域中。

## 2 卡尔曼滤波的算法推导

为便于理解卡尔曼滤波算法的基本原理，我们对图1-1进行简化，得到其简化模型，如图2-2所示。可以看出，卡尔曼滤波算法首先根据系统自身的动力特征对系统状态进行预测，然后再利用观测或测量到的新数据对预测状态进行更新或校正，使状态预测更为准确，这便完成了算法的一轮迭代。

实际上，卡尔曼滤波算法的上述工作方式，使之适用于估计由随机变量组成的动态系统的最优状态——即便观测数据中包含有噪声，观测值不准确，卡尔曼滤波算法也能够完成对状态真实值的最优估计。

下面，我们从一个应用实例开始，逐步推导出卡尔曼滤波算法。

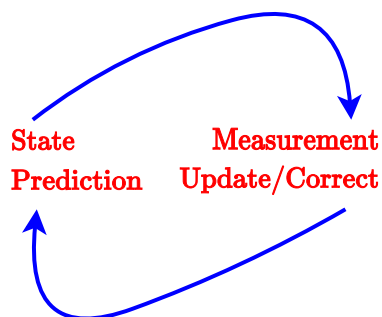


图 2-2: 卡尔曼滤波算法的简化模型

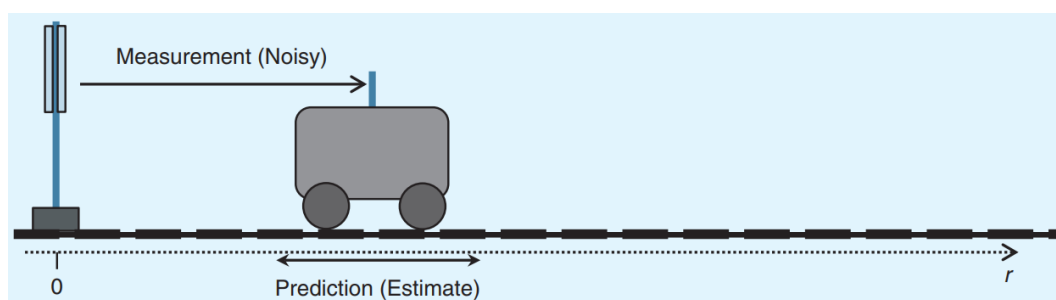


图 2-3: 列车位置与速度的估算问题

如图2-3所示，假设有一辆列车行进在一个平直的轨道上，且在轨道的一端配备了雷达测速与测距仪，设列车的质量为  $m$ ，列车司机通过踩刹车或踩油门达到减速或加速的目的，其施加给列车的力表示为  $f_t$ 。试结合雷达测量数据以及列车自身的运动规律，对列车的位置与速度做出“准确的”估计。

首先，应用运动方程对列车的状态过程进行建模。设  $x_t$  表示  $t$  时刻的位置，可表示为：

$$x_t = x_{t-1} + \dot{x}_{t-1}\Delta t + \frac{f_t(\Delta t)^2}{2m} \quad (1)$$

设  $\dot{x}_t$  表示  $t$  时刻的速度，可表示为：

$$\dot{x}_t = \dot{x}_{t-1} + \frac{f_t\Delta t}{m} \quad (2)$$

其中， $\Delta t$  表示时刻  $t-1$  与时刻  $t$  之间的时间间隔。令  $u_t = \frac{f_t}{m}$ ，它表示加速度。实际上，从卡尔曼滤波算法的角度而言，它是列车（动态系统）状态的控制量<sup>4</sup>。于

<sup>4</sup>一般情况下，控制量表现为向量形式  $\mathbf{u}_t$ （默认为列向量）。对于本例，控制量为标量形式  $u_t$ 。依据问题的性质，控制向量  $\mathbf{u}_t$  的分量可能包括施加给系统的力，例如牵引力与制动力等，这些力用于改变系统的运动状态；分量也可能为力施加之后被改变的量，例如加速度等。

是，方程 (1) 与方程 (2) 可表示为：

$$\begin{cases} x_t = x_{t-1} + \dot{x}_{t-1}\Delta t + \frac{(\Delta t)^2}{2}u_t \\ \dot{x}_t = \dot{x}_{t-1} + \Delta t u_t \end{cases} \quad (3)$$

借助于矩阵形式，上式可表示为：

$$\begin{bmatrix} x_t \\ \dot{x}_t \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ \dot{x}_{t-1} \end{bmatrix} + \begin{bmatrix} \frac{(\Delta t)^2}{2} \\ \Delta t \end{bmatrix} u_t \quad (4)$$

进一步地，设状态向量 (State Vector) 为  $\mathbf{x}_t$ ，其定义为<sup>5</sup>：

$$\mathbf{x}_t = \begin{bmatrix} x_t \\ \dot{x}_t \end{bmatrix} \quad (5)$$

设状态转移矩阵为  $\mathbf{F}$ ，其定义为：

$$\mathbf{F} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \quad (6)$$

该矩阵的名称来源于其实现的功能——对时刻  $t-1$  的状态向量  $\mathbf{x}_{t-1}$  执行矩阵变换  $\mathbf{F}$ ，得到的变换结果就是时刻  $t$  的状态向量  $\mathbf{x}_t$  的一个加法项。设控制输入矩阵为  $\mathbf{B}$ <sup>6</sup>，其定义为：

$$\mathbf{B} = \begin{bmatrix} \frac{(\Delta t)^2}{2} \\ \Delta t \end{bmatrix} \quad (7)$$

于是，可以将公式 (4) 表示为<sup>7</sup>：

$$\mathbf{x}_t = \mathbf{F}\mathbf{x}_{t-1} + \mathbf{B}u_t \quad (8)$$

上式建立了  $t-1$  时刻的状态  $\mathbf{x}_{t-1}$  与  $t$  时刻的状态  $\mathbf{x}_t$  之间的递推或演化关系。其意义在于，时刻  $t$  的状态  $\mathbf{x}_t$  由时刻  $t-1$  的状态  $\mathbf{x}_{t-1}$ 、状态转移矩阵  $\mathbf{F}$ 、控制输入矩阵  $\mathbf{B}$  以及系统动力控制量  $u_t$  共同 (线性地) 确定。上述动力系统被称为线性动态系统。

<sup>5</sup>依据问题的性质，状态向量  $\mathbf{x}_t$  的分量可能包括位置、速度、方向等描述系统状态的量。

<sup>6</sup>一般情况下，控制输入矩阵  $\mathbf{B}$  为矩阵形式。此处，既可以把  $\mathbf{B}$  当作  $2 \times 1$  矩阵，也可以当作列向量。

<sup>7</sup>从公式的通用性角度考虑，我们将标量作为特殊的向量，因而也使用粗体字  $u_t$  表示本例中的控制量  $u_t$ 。

然后，应用雷达测速/测距仪获得的信息对列车的测量过程进行建模。由于测量过程与状态过程所导出的量，可能存在于不同的空间中，因而需要引入变换矩阵  $H$ ，它负责将状态向量  $x_t$  转换为观测向量  $y_t$ ，即存在如下变换：

$$y_t = Hx_t \quad (9)$$

于是，方程 (8) 与方程 (9) 分别完成了对系统的状态过程与测量过程的建模。

然而，对于大多数实际的动态系统，存在着诸多不稳定的因素，这些因素将会导致状态过程与测量过程的不确定性。例如，对于一个在地面上行进的带有 GPS 定位的轮式机器人而言，如果对机器人下达诸如“前进 10 米”、“向右前进 10 米”等类似的指令，尽管下达的指令清晰无误，但是由于受到机器人自身执行机构（马达/电动机、轮子等）及外部环境的影响（风向、地面湿滑等），机器人实际行驶的距离与理想值并不完全吻合。此外，机器人的 GPS 传感器，也不可避免地存在测量误差。这意味着，不论是状态过程，还是测量过程，所获得的信息都具有某种不确定性，都是不准确的。因此，需要将上述各种不确定性因素考虑在内，显式地对它们建模。

基于上述认识，将系统的状态方程 (8) 与测量方程 (9) 重新定义为：

$$\begin{cases} x_t = Fx_{t-1} + Bu_t + \epsilon_t \\ y_t = Hx_t + \delta_t \end{cases} \quad (10)$$

其中， $\epsilon_t$  表示状态过程的噪声向量，且假设  $\epsilon_t$  的每个分量服从零均值分布，即  $\epsilon_t \sim p_1(0, Q_t)$ ，其中  $Q_t$  表示状态过程噪声的协方差矩阵。 $\delta_t$  表示测量过程的噪声向量，且假设  $\delta_t$  的每个分量服从零均值分布，即  $\delta_t \sim p_2(0, R_t)$ ，其中  $R_t$  表示测量过程噪声的协方差矩阵。 $x_t$  表示  $t$  时刻系统的真实状态， $y_t$  表示  $t$  时刻系统的实际观测。

从方程组 (10) 可以看出， $t$  时刻系统的真实状态  $x_t$  不能直接获得，对于我们而言，它是未知的。但是，我们可以有效地混合系统的估算状态  $Fx_{t-1} + Bu_t$  及实际观测  $y_t$ ，从而得到较为准确的、接近真实状态  $x_t$  的估算状态  $\hat{x}_t$ 。

为了准确地描述各个参与计算的成分，定义  $t$  时刻的先验估算状态  $\hat{x}_{t|t-1}$  为：

$$\hat{x}_{t|t-1} = F\hat{x}_{t-1|t-1} + Bu_t \quad (11)$$

它表示直接依据系统的动力方程，利用  $t-1$  时刻的后验估算状态  $\hat{x}_{t-1|t-1}$  计算而得到的估算状态。于是，可以得到先验状态  $\hat{x}_{t|t-1}$  与未知的真实状态  $x_t$  之间差的

协方差矩阵：

$$\mathbf{P}_{t|t-1} = \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T \right] \quad (12)$$

将公式 (10) 中的  $\mathbf{x}_t$  及公式 (11) 中的  $\hat{\mathbf{x}}_{t|t-1}$  代入上式，可得：

$$\begin{aligned} \mathbf{P}_{t|t-1} &= \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T \right] \\ &= \mathbb{E} \left[ (\mathbf{F} (\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1|t-1}) + \boldsymbol{\epsilon}_t) (\mathbf{F} (\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1|t-1}) + \boldsymbol{\epsilon}_t)^T \right] \\ &= \mathbf{F} \mathbb{E} \left[ (\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1|t-1}) (\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1|t-1})^T \right] \mathbf{F}^T + \\ &\quad \mathbf{F} \mathbb{E} \left[ (\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1|t-1}) \boldsymbol{\epsilon}_t^T \right] + \mathbb{E} \left[ \boldsymbol{\epsilon}_t (\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1|t-1})^T \right] \mathbf{F}^T + \mathbb{E} [\boldsymbol{\epsilon}_t \boldsymbol{\epsilon}_t^T] \\ &= \mathbf{F} \mathbf{P}_{t-1|t-1} \mathbf{F}^T + \mathbf{0} + \mathbf{0} + \mathbf{Q}_t = \mathbf{F} \mathbf{P}_{t-1|t-1} \mathbf{F}^T + \mathbf{Q}_t \end{aligned} \quad (13)$$

上式的倒数第 2 行，利用了估算状态的误差  $(\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1|t-1})$  与状态过程噪声  $\boldsymbol{\epsilon}_t$  之间的不相关性。

至此，我们得到了状态预测阶段所需的 2 个公式：

$$\begin{cases} \hat{\mathbf{x}}_{t|t-1} = \mathbf{F} \hat{\mathbf{x}}_{t-1|t-1} + \mathbf{B} \mathbf{u}_t \\ \mathbf{P}_{t|t-1} = \mathbf{F} \mathbf{P}_{t-1|t-1} \mathbf{F}^T + \mathbf{Q}_t \end{cases} \quad (14)$$

现在，进入测量更新阶段，即利用  $t$  时刻的实际测量向量  $\mathbf{y}_t$  来更新先验状态  $\hat{\mathbf{x}}_{t|t-1}$ ，从而得到较为准确的后验状态  $\hat{\mathbf{x}}_{t|t}$ 。其更新公式采用如下的线性混合公式：

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t|t-1} + \mathbf{K}_t (\mathbf{y}_t - \mathbf{H} \hat{\mathbf{x}}_{t|t-1}) \quad (15)$$

其中， $\mathbf{K}_t$  被称为混合因子 (矩阵)。实际上，它就是每次迭代需要优化计算的卡尔曼增益 (Kalman Gain)。

为了得到最优的卡尔曼增益  $\mathbf{K}_t$ ，我们需要定义一个优化准则。此处，我们仍然选取均方误差最小化作为优化准则。具体地，首先计算后验状态  $\hat{\mathbf{x}}_{t|t}$  与未知的真实状态  $\mathbf{x}_t$  之间差的协方差矩阵：

$$\mathbf{P}_{t|t} = \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t}) (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t})^T \right] \quad (16)$$

将公式 (10) 中的  $\mathbf{y}_t$  代入公式 (15)，并将公式 (15) 代入上式，可得：

$$\begin{aligned} \mathbf{P}_{t|t} &= \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t}) (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t})^T \right] \\ &= \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1} - \mathbf{K}_t (\mathbf{H} \mathbf{x}_t + \boldsymbol{\delta}_t - \mathbf{H} \hat{\mathbf{x}}_{t|t-1})) \right. \\ &\quad \left. (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1} - \mathbf{K}_t (\mathbf{H} \mathbf{x}_t + \boldsymbol{\delta}_t - \mathbf{H} \hat{\mathbf{x}}_{t|t-1}))^T \right] \\ &= \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1} - \mathbf{K}_t (\mathbf{H} (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) + \boldsymbol{\delta}_t)) \right. \\ &\quad \left. (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1} - \mathbf{K}_t (\mathbf{H} (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) + \boldsymbol{\delta}_t))^T \right] \end{aligned} \quad (17)$$



利用公式  $\mathbf{P}_{t|t-1} = \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T \right]$  进一步对上式变形：

$$\begin{aligned}
 \mathbf{P}_{t|t} &= \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T \right] - \\
 &\quad \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T \mathbf{H}^T \mathbf{K}_t^T \right] - \\
 &\quad \mathbb{E} \left[ (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) \boldsymbol{\delta}_t^T \mathbf{K}_t^T \right] - \\
 &\quad \mathbb{E} \left[ \mathbf{K}_t \mathbf{H} (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})^T \right] - \\
 &\quad \mathbb{E} \left[ \mathbf{K}_t \boldsymbol{\delta}_t (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) \right] + \\
 &\quad \mathbb{E} \left[ (\mathbf{K}_t (\mathbf{H} (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) + \boldsymbol{\delta}_t)) (\mathbf{K}_t (\mathbf{H} (\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1}) + \boldsymbol{\delta}_t))^T \right]
 \end{aligned} \tag{18}$$

利用先验估算误差  $(\mathbf{x}_t - \hat{\mathbf{x}}_{t|t-1})$  与测量误差  $\boldsymbol{\delta}_t$  之间的不相关性，继续对上式变形：

$$\begin{aligned}
 \mathbf{P}_{t|t} &= \mathbf{P}_{t|t-1} - \mathbf{P}_{t|t-1} \mathbf{H}^T \mathbf{K}_t^T - \mathbf{K}_t \mathbf{H} \mathbf{P}_{t|t-1} + \\
 &\quad (\mathbf{K}_t \mathbf{H}) \mathbf{P}_{t|t-1} (\mathbf{K}_t \mathbf{H})^T + \mathbf{K}_t \mathbb{E} [\boldsymbol{\delta}_t \boldsymbol{\delta}_t^T] \mathbf{K}_t^T \\
 &= \mathbf{P}_{t|t-1} - \mathbf{P}_{t|t-1} (\mathbf{K}_t \mathbf{H})^T - \mathbf{K}_t \mathbf{H} \mathbf{P}_{t|t-1} + \\
 &\quad (\mathbf{K}_t \mathbf{H}) \mathbf{P}_{t|t-1} (\mathbf{K}_t \mathbf{H})^T + \mathbf{K}_t \mathbf{R}_t \mathbf{K}_t^T \\
 &= (\mathbf{I} - \mathbf{K}_t \mathbf{H}) \mathbf{P}_{t|t-1} (\mathbf{I} - \mathbf{K}_t \mathbf{H})^T + \mathbf{K}_t \mathbf{R}_t \mathbf{K}_t^T
 \end{aligned} \tag{19}$$

上式便是计算后验状态误差的协方差矩阵  $\mathbf{P}_{t|t}$  的通式。为了便于求导优化，我们使用另一种形式的  $\mathbf{P}_{t|t}$ ，对上式的倒数第 2 行继续变形，可得：

$$\begin{aligned}
 \mathbf{P}_{t|t} &= \mathbf{P}_{t|t-1} - \mathbf{P}_{t|t-1} (\mathbf{K}_t \mathbf{H})^T - \mathbf{K}_t \mathbf{H} \mathbf{P}_{t|t-1} + \\
 &\quad (\mathbf{K}_t \mathbf{H}) \mathbf{P}_{t|t-1} (\mathbf{K}_t \mathbf{H})^T + \mathbf{K}_t \mathbf{R}_t \mathbf{K}_t^T \\
 &= \mathbf{P}_{t|t-1} - \mathbf{P}_{t|t-1} \mathbf{H}^T \mathbf{K}_t^T - \mathbf{K}_t \mathbf{H} \mathbf{P}_{t|t-1} + \\
 &\quad \mathbf{K}_t (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) \mathbf{K}_t^T
 \end{aligned} \tag{20}$$

对后验状态误差的协方差矩阵  $\mathbf{P}_{t|t}$  稍加分析可知，其主对角线上的元素就是状态向量的各个状态分量的后验误差的（标量）方差。因此，我们的优化目标是，针对卡尔曼增益  $\mathbf{K}_t$ ，最小化它们的和（均方误差和），即  $\min_{\mathbf{K}_t} \text{tr}(\mathbf{P}_{t|t})$ 。于是，应用标

量函数对矩阵求导的方法及迹技巧 (Trace Trick), 可得<sup>8</sup>:

$$\begin{aligned}
 d(\text{tr}(\mathbf{P}_{t|t})) &= \text{tr}(d\mathbf{P}_{t|t}) \\
 &= \text{tr}(-\mathbf{P}_{t|t-1}\mathbf{H}^T d\mathbf{K}_t^T - d\mathbf{K}_t \mathbf{H} \mathbf{P}_{t|t-1} + \\
 &\quad d\mathbf{K}_t (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) \mathbf{K}_t^T + \mathbf{K}_t (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) d\mathbf{K}_t^T)
 \end{aligned} \tag{21}$$

利用公式  $\text{tr}(\mathbf{X}) = \text{tr}(\mathbf{X}^T)$  及协方差矩阵  $\mathbf{P}_{t|t-1}$  与协方差矩阵  $\mathbf{R}_t$  的对称性, 继续变形, 可得:

$$\begin{aligned}
 d(\text{tr}(\mathbf{P}_{t|t})) &= \text{tr}(-2d\mathbf{K}_t \mathbf{H} \mathbf{P}_{t|t-1} + 2d\mathbf{K}_t (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) \mathbf{K}_t^T) \\
 &= \text{tr}(-2\mathbf{H} \mathbf{P}_{t|t-1} d\mathbf{K}_t + 2(\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) \mathbf{K}_t^T d\mathbf{K}_t)
 \end{aligned} \tag{22}$$

于是, 可以得到导数:

$$\frac{d(\text{tr}(\mathbf{P}_{t|t}))}{d\mathbf{K}_t} = -2\mathbf{P}_{t|t-1}\mathbf{H}^T + 2\mathbf{K}_t (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) \tag{23}$$

令该导数等于 0, 可以求解出最优卡尔曼增益:

$$\mathbf{K}_t^* = \mathbf{P}_{t|t-1}\mathbf{H}^T (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t)^{-1} \tag{24}$$

将公式 (24) 代入公式 (20) 中, 得到:

$$\begin{aligned}
 \mathbf{P}_{t|t}^* &= \mathbf{P}_{t|t-1} - \mathbf{P}_{t|t-1}\mathbf{H}^T (\mathbf{K}_t^*)^T - \mathbf{K}_t^* \mathbf{H} \mathbf{P}_{t|t-1} + \mathbf{K}_t^* (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) (\mathbf{K}_t^*)^T \\
 &= \mathbf{P}_{t|t-1} - \mathbf{P}_{t|t-1}\mathbf{H}^T (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t)^{-1} \mathbf{H} \mathbf{P}_{t|t-1} - \\
 &\quad \mathbf{P}_{t|t-1}\mathbf{H}^T (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t)^{-1} \mathbf{H} \mathbf{P}_{t|t-1} + \\
 &\quad \mathbf{P}_{t|t-1}\mathbf{H}^T (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t)^{-1} (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t)^{-1} \mathbf{H} \mathbf{P}_{t|t-1} \\
 &= \mathbf{P}_{t|t-1} - \mathbf{P}_{t|t-1}\mathbf{H}^T (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t)^{-1} \mathbf{H} \mathbf{P}_{t|t-1}
 \end{aligned} \tag{25}$$

上式是计算  $\mathbf{P}_{t|t}^*$  的基本通式。如果将公式 (24) 代回上式, 也可以得到如下的两个等价公式:

$$\mathbf{P}_{t|t}^* = \mathbf{P}_{t|t-1} - \mathbf{K}_t^* (\mathbf{H} \mathbf{P}_{t|t-1} \mathbf{H}^T + \mathbf{R}_t) (\mathbf{K}_t^*)^T \tag{26}$$

或者:

$$\mathbf{P}_{t|t}^* = \mathbf{P}_{t|t-1} - \mathbf{K}_t^* \mathbf{H} \mathbf{P}_{t|t-1} = (\mathbf{I} - \mathbf{K}_t^* \mathbf{H}) \mathbf{P}_{t|t-1} \tag{27}$$

<sup>8</sup>关于“标量函数对矩阵的求导”的内容, 请参见《机器学习》课程系列之“判别函数的线性分类基础”的附录部分。

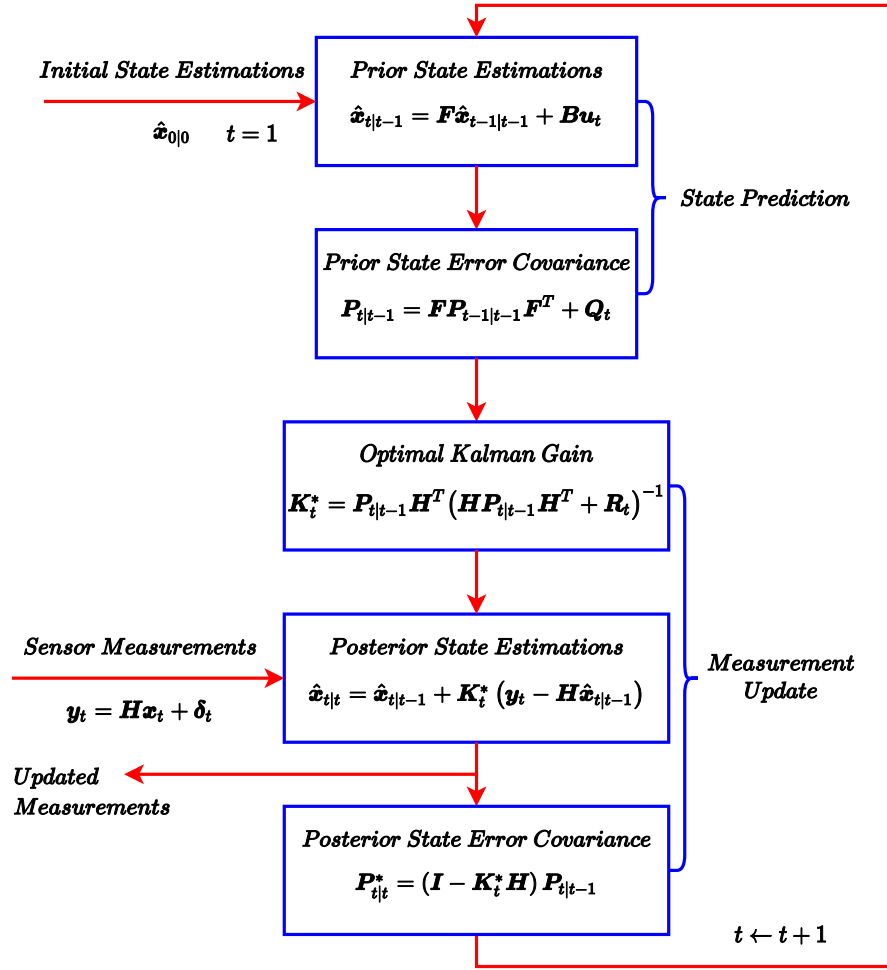


图 2-4: 卡尔曼滤波算法流程图

其中，公式 (27) 的形式是非常简洁的。

至此，我们得到了测量更新阶段所需要的 3 个公式：

$$\begin{cases} \hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t^*(y_t - H\hat{x}_{t|t-1}) \\ P_{t|t}^* = (I - K_t^*H)P_{t|t-1} \\ K_t^* = P_{t|t-1}H^T(HP_{t|t-1}H^T + R_t)^{-1} \end{cases} \quad (28)$$

公式 (14) 和公式 (28) 便组成了卡尔曼滤波算法的 5 个核心递推公式。

虽然我们从一个应用实例出发，推导出了卡尔曼滤波算法，但是鉴于公式 (10) 具备一般化形式，上述 5 个核心递推公式，也适用于一般情形。

图2-4展示了卡尔曼滤波算法的流程图。

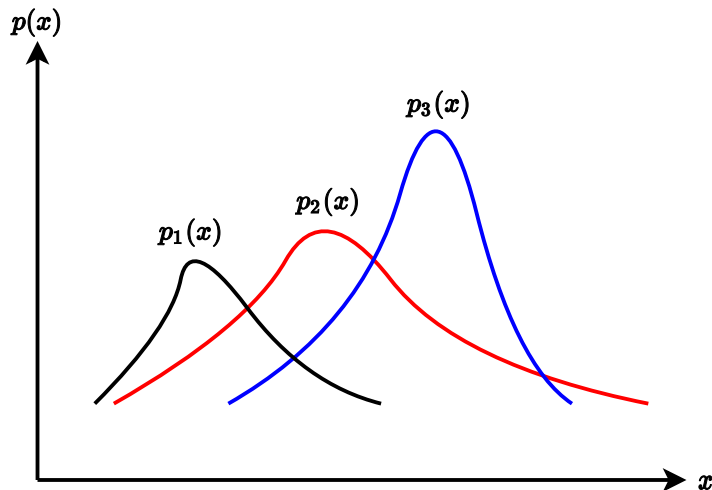


图 3-5: 3 个估算量服从不同的概率分布

### 3 滤波算法的通用形式

下面，从更一般性的角度来考虑滤波算法的基本组成与更新规则。

由前面的讨论可知，从后验状态估算误差的角度而言，卡尔曼滤波能够将先验状态估算与测量数据最优地融合在一起，其基本出发点为线性融合或混合公式(15)。下面，我们对其进行扩展——将多个数据或信息进行最优的融合，以得到最优的状态估算。

为讨论方便，在下面的论述中，将所有参与融合或混合的量均统称为估算(Estimation)，即估算既可以表示带噪声的测量数据，也可以表示依据系统动力性计算而得到的状态数据。此外，“估算”一词也能够反映出这些数据自身具有的近似性与不准确性。

为了建模每个估算量  $x_i$ ，我们令  $x_i \sim p_i(\mu_i, \Sigma_i)$ ，它表示估算量  $x_i$  服从均值为  $\mu_i$ 、协方差矩阵为  $\Sigma_i$  的概率分布。如果估算量为标量  $x_i$ ，则  $x_i \sim p_i(\mu_i, \sigma_i^2)$ ，它表示估算量  $x_i$  服从均值为  $\mu_i$ 、方差为  $\sigma_i^2$  的概率分布。

例如，图3-5展示了服从不同概率分布的 3 个估算量。对于此例，融合量  $y$  可表示为：

$$y = \sum_{i=1}^3 \alpha_i x_i \quad (29)$$

即  $y$  表示  $x_i (i = 1, 2, 3)$  的加权混合或线性组合，其中  $\alpha_i$  表示混合权值<sup>9</sup>。

<sup>9</sup>一般情况下， $\alpha_i$  为归一化权值，即  $\sum_{i=1}^n \alpha_i = 1$ 。如果权值  $\alpha_i$  不满足此条件，可以使用公式

显然，在融合量  $y$  的计算公式中，一个关键的地方是，如何最优地确定每个估算量  $x_i$  的权值系数。对于通用的滤波器而言，算法的核心思想是，将每个估算量  $x_i$  的方差  $\sigma_i^2$  与它的权值系数  $\alpha_i$  进行关联——粗略地说，方差  $\sigma_i^2$  越大，表明估算量  $x_i$  越不可靠，其权值  $\alpha_i$  就应该越小；反之，权值  $\alpha_i$  就应该越大<sup>10</sup>。这是一种非常自然的想法！下文推导出的最优权值计算公式，确实反映了通用滤波算法的这一特点。在图3-5中，3个估算量的方差关系为  $\sigma_1^2 < \sigma_3^2 < \sigma_2^2$ ，则它们的权值关系为  $\alpha_1 > \alpha_3 > \alpha_2$ 。

在下面的推导过程中，我们假设估算量之间是不相关的，即估算量两两满足关系  $\mathbb{E}[x_i|x_j] = \mathbb{E}[x_i](i \neq j)$ <sup>11</sup>。这一假设是较为合理的——如果  $x_i$  与  $x_j$  不相关，则在已知  $x_i$  的情况下，该假设既不会妨碍我们对  $x_j$  取值的大致估算（即其值应该与  $x_i$  接近），但也不会对  $x_j$  按自身分布取值造成影响（即  $x_j \sim p_j(\mu_j, \sigma_j^2)$ ）；相反，如果假设  $x_i$  与  $x_j$  相互独立，即假设  $x_i$  与  $x_j$  完全无关，则该约束条件过于苛刻，这相当于直接否定了  $x_j$  的取值与  $x_i$  相近的事实——不符合  $x_i$  与  $x_j$  均为对真实值的估算这一事实。

此外，我们也假设估算器 (Estimator) 为无偏估算器 (Unbiased Estimator)，即假设其均值与真实值 (Ground Truth) 一致。

### 3.1 标量情形

#### 3.1.1 2个标量情形

**引理 3.1** 设  $x_1 \sim p_1(\mu_1, \sigma_1^2), \dots, x_n \sim p_n(\mu_n, \sigma_n^2)$ ，其中  $x_i$  与  $x_j (i \neq j)$  不相关，设  $y = \sum_{i=1}^n \alpha_i x_i$ ，且  $\sum_{i=1}^n \alpha_i = 1$ ，则<sup>12</sup>：

---

$\alpha_i / \sum_{i=1}^n \alpha_i$  进行归一化。

<sup>10</sup>对于前述的卡尔曼滤波算法，如果假设估算量服从高斯分布，则稍加分析就可以表明这一点。高斯分布下的卡尔曼滤波算法的推导过程，详见文献 [5]。

<sup>11</sup>此处，仅列出标量估算量之间的关系。对于向量估算量形式，关系类似。关系  $\mathbb{E}[x_i|x_j] = \mathbb{E}[x_i]$  等价于关系  $\mathbb{E}[(x_i - \mu_i)(x_j - \mu_j)] = 0$ 。这可以依据方差的定义公式，直接得到。需要注意的是，2个随机变量之间的不相关性条件要弱于独立性条件。如果  $x_i$  与  $x_j$  相互独立，则  $p(x_2|x_1) = p(x_2)$ 。此外，由独立性条件，可以得出不相关性结论。但是，反过来不一定成立。不相关性，只能表明2个随机变量之间不存在线性关系，但非线性关系仍可能成立，即它们不一定相互独立。

<sup>12</sup>相关的推导，请参考文献 [6]。

1.

$$\begin{cases} \mu_y = \sum_{i=1}^n \alpha_i \mu_i \\ \sigma_y^2 = \sum_{i=1}^n \alpha_i^2 \sigma_i^2 \end{cases} \quad (30)$$

2. 如果随机变量  $x_{n+1}$  与  $x_1, \dots, x_n$  两两不相关, 则  $x_{n+1}$  与  $y$  也是不相关的。

首先, 讨论 2 个估算量的融合。设  $x_1$  的权值为  $\alpha$ , 则  $x_2$  的权值可定义为  $1 - \alpha$ 。于是, 得到:

$$y = \alpha x_1 + (1 - \alpha) x_2 \quad (31)$$

其方差为:

$$\sigma_y^2 = \alpha^2 \sigma_1^2 + (1 - \alpha)^2 \sigma_2^2 \quad (32)$$

为了得到最优权值  $\alpha$ , 需要关于  $\alpha$  最小化  $\sigma_y^2$ 。求其导数, 并令其等于 0, 得到:

$$\frac{d\sigma_y^2}{d\alpha} = 2\alpha\sigma_1^2 - 2(1 - \alpha)\sigma_2^2 = 0 \quad (33)$$

求解  $\alpha$ , 得到:

$$\alpha^* = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \quad (34)$$

上式意义非常明确: 如果  $\sigma_1^2 < \sigma_2^2$ , 表明  $x_1 \sim p_1(\mu_1, \sigma_1^2)$  的结果更可信, 则  $\alpha_1 = \alpha^* > 1 - \alpha^* = \alpha_2$ ,  $x_1$  与  $x_2$  融合时,  $x_1$  的权值更大些, 即  $x_1$  占比更多些。实际上, 权值  $\alpha^*$  充当了卡尔曼滤波中的最优卡尔曼增益  $k^*$  的角色。

为了便于观察  $\alpha^*$  公式中的规律及扩展为  $n$  个估算量情形, 可以将公式 (34) 变形为:

$$\alpha^* = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} = \frac{\frac{1}{\sigma_1^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \quad (35)$$

于是, 最优线性融合量  $y^*$  可以表示为:

$$y^* = \frac{\frac{1}{\sigma_1^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} x_1 + \frac{\frac{1}{\sigma_2^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} x_2 \quad (36)$$

最优线性融合量  $y^*$  的方差为:

$$\begin{aligned} \sigma_{y^*}^2 &= \alpha_*^2 \sigma_1^2 + (1 - \alpha_*)^2 \sigma_2^2 \\ &= \frac{1}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \end{aligned} \quad (37)$$

或者，可表示为：

$$\sigma_{y^*}^2 = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2} \quad (38)$$

为了进一步简化公式，可以将 2 个分布的方差分别替换为其精度形式。具体地，令：

$$\begin{cases} v_1 = \frac{1}{\sigma_1^2} \\ v_2 = \frac{1}{\sigma_2^2} \end{cases} \quad (39)$$

于是，最优线性融合量  $y^*$  及其精度分别为：

$$\begin{cases} y^* = \frac{v_1}{v_1 + v_2} x_1 + \frac{v_2}{v_1 + v_2} x_2 \\ v_y^* = v_1 + v_2 \end{cases} \quad (40)$$

此处，我们可以将精度理解为对估算量的可信度：

- 估算量所占的权重，取决于估算量的可信度，且与可信度成正比；
- 融合量的可信度，等于估算量的可信度之和，即它要高于任一单个估算量的可信度。这表明融合后的结果更加可靠；

在实际应用中，很难确定各个估算量的概率分布的具体形式。但是，上述公式表明，我们只需要确定这些概率分布的方差，而无需确定其均值以及具体形式。

### 3.1.2 多个标量情形

对于  $n$  个估算量的融合，可以得出类似的结论：

$$\begin{cases} \alpha_i^* = \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^n \frac{1}{\sigma_j^2}} \\ y^* = \sum_{i=1}^n \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^n \frac{1}{\sigma_j^2}} x_i \\ \sigma_{y^*}^2 = \frac{1}{\sum_{j=1}^n \frac{1}{\sigma_j^2}} \end{cases} \quad (41)$$

令  $v_i = \frac{1}{\sigma_i^2}$ ，则可以得到：

$$\begin{cases} \alpha_i^* = \frac{v_i}{\sum_{j=1}^n v_j} \\ y^* = \sum_{i=1}^n \frac{v_i}{\sum_{j=1}^n v_j} x_i \\ v_y^* = \sum_{j=1}^n v_j \end{cases} \quad (42)$$

证明：由引理 (3.1) 可知，从  $n$  个估算量得到融合量  $y$ ，其方差为：

$$\sigma_y^2 = \sum_{i=1}^n \alpha_i^2 \sigma_i^2 \quad (43)$$

为得到最优权值系数  $\alpha_i$ ，可以关于  $\alpha_i$  最优化求解上面的方差函数  $\sigma_y^2$ ，这是一个带约束条件  $\sum_{i=1}^n \alpha_i = 1$  的优化问题。为此，可以应用拉格朗日乘数法，列出其拉格朗日优化函数为<sup>13</sup>：

$$\mathcal{L}(\alpha_1, \dots, \alpha_n) = \sum_{i=1}^n \alpha_i^2 \sigma_i^2 + \lambda \left( \sum_{i=1}^n \alpha_i - 1 \right) \quad (44)$$

其中  $\lambda$  为拉格朗日乘数。求解如下偏导数，并令其等于 0，得到：

$$\begin{cases} \frac{\partial \mathcal{L}(\alpha_1, \dots, \alpha_n)}{\partial \alpha_1} = 2\alpha_1 \sigma_1^2 + \lambda = 0 \\ \frac{\partial \mathcal{L}(\alpha_1, \dots, \alpha_n)}{\partial \alpha_2} = 2\alpha_2 \sigma_2^2 + \lambda = 0 \\ \vdots \\ \frac{\partial \mathcal{L}(\alpha_1, \dots, \alpha_n)}{\partial \alpha_n} = 2\alpha_n \sigma_n^2 + \lambda = 0 \end{cases} \Rightarrow \begin{cases} \alpha_1 = -\frac{\lambda}{2\sigma_1^2} \\ \alpha_2 = -\frac{\lambda}{2\sigma_2^2} \\ \vdots \\ \alpha_n = -\frac{\lambda}{2\sigma_n^2} \end{cases} \quad (45)$$

进一步地，利用约束条件  $\sum_{i=1}^n \alpha_i = 1$ ，得到：

$$-\frac{\lambda}{2\sigma_1^2} - \frac{\lambda}{2\sigma_2^2} - \dots - \frac{\lambda}{2\sigma_n^2} = 1 \quad (46)$$

求解  $\lambda$  得到：

$$\lambda = \frac{-2}{\sum_{j=1}^n \frac{1}{\sigma_j^2}} \quad (47)$$

<sup>13</sup>关于拉格朗日乘数法，请阅读《机器学习》课程系列之“基础知识”。



将上式代入公式 (45)，可以得到：

$$\alpha_i^* = \frac{\frac{1}{\sigma_i^2}}{\sum_{j=1}^n \frac{1}{\sigma_j^2}} \quad (48)$$

最后将  $\alpha_i^*$  分别代入方差  $\sigma_y^2$  及融合量  $y$  的计算公式，即可得证。  $\square$

### 3.1.3 增量融合

在实际应用中，多数情况下，估算量可能以增量的方式获得，例如每次只获得 1 个估算量。在这种方式下，如何利用目前所得到的估算量计算出融合量呢？一种直接的方式是，执行所谓的批处理——只要得到 1 个估算量，就按照公式 (42) 计算出新的融合量  $y^*$ 。然而，这种批计算方式，存在 2 个问题：

- 计算效率低

没有有效地利用已有的融合量进行增量计算；

- 空间浪费大

内存中需要保存所有的估算量；

为了解决上述问题，可以利用所谓的增量计算技术。为此，需要分别建立  $n$  个估算量与  $n-1$  个估算量时  $y^*$  与  $v_y^*$  的递推关系。为讨论方便，将递推步  $n$  与  $n-1$  显式地表示于公式符号中。对于  $v_y^*$ ，得到：

$$v_{y_n}^* = \sum_{j=1}^n v_j \quad \Rightarrow \quad v_{y_n}^* = v_{y_{n-1}}^* + v_n \quad (49)$$

对于  $y^*$ ，得到：

$$\begin{aligned} y_n^* &= \sum_{i=1}^n \frac{v_i}{\sum_{j=1}^n v_j} x_i \quad \Rightarrow \quad y_n^* = \sum_{i=1}^{n-1} \frac{v_i}{\sum_{j=1}^{n-1} v_j} x_i + \frac{v_n}{\sum_{j=1}^n v_j} x_n \quad \Rightarrow \\ y_n^* &= \frac{\sum_{i=1}^{n-1} v_i x_i}{\sum_{j=1}^{n-1} v_j} + \frac{v_n}{\sum_{j=1}^n v_j} x_n \quad \Rightarrow \quad y_n^* = \frac{y_{n-1}^* \sum_{j=1}^{n-1} v_j}{\sum_{j=1}^n v_j} + \frac{v_n}{\sum_{j=1}^n v_j} x_n \quad \Rightarrow \\ y_n^* &= \frac{v_{y_{n-1}}^*}{v_{y_{n-1}}^* + v_n} y_{n-1}^* + \frac{v_n}{v_{y_{n-1}}^* + v_n} x_n \end{aligned} \quad (50)$$

上式意义也非常明确：时刻  $n$  的最优融合量  $y_n^*$  等于时刻  $n-1$  的最优融合量  $y_{n-1}^*$  与时刻  $n$  的估算量  $x_n$  的加权平均，权值由它们的精度占比确定。

于是，利用时间步  $n$  与  $n-1$  之间的递推公式 (49) 和递推公式 (50) 即可进行增量融合。

从参与融合的数据个数的角度而言，上述增量融合公式，相当于将  $n$  个估算量的融合转换为 2 个估算量的增量融合，使得滤波算法可以应用于实时系统中。

### 3.2 向量情形

引理 3.2 设  $\mathbf{x}_1 \sim p_1(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), \dots, \mathbf{x}_n \sim p_n(\boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n)$ ，其中  $\mathbf{x}_i$  与  $\mathbf{x}_j (i \neq j)$  不相关，设  $\mathbf{y} = \sum_{i=1}^n \mathbf{A}_i \mathbf{x}_i$ ，且  $\sum_{i=1}^n \mathbf{A}_i = \mathbf{I}$ ，则<sup>14</sup>：

$$1. \quad \begin{cases} \boldsymbol{\mu}_y = \sum_{i=1}^n \mathbf{A}_i \boldsymbol{\mu}_i \\ \boldsymbol{\Sigma}_{yy} = \sum_{i=1}^n \mathbf{A}_i \boldsymbol{\Sigma}_i \mathbf{A}_i^T \end{cases} \quad (51)$$

2. 如果随机变量  $\mathbf{x}_{n+1}$  与  $\mathbf{x}_1, \dots, \mathbf{x}_n$  两两不相关，则  $\mathbf{x}_{n+1}$  与  $\mathbf{y}$  也是不相关的。

在估算量为向量的情形下，滤波算法的最优化目标是，最小化均方误差和，即：

$$\min_{\mathbf{A}_1, \dots, \mathbf{A}_n} \text{tr} \left( \mathbb{E} \left[ (\mathbf{y} - \boldsymbol{\mu}_y) (\mathbf{y} - \boldsymbol{\mu}_y)^T \right] \right) = \min_{\mathbf{A}_1, \dots, \mathbf{A}_n} \text{tr} (\boldsymbol{\Sigma}_{yy}) \quad (52)$$

或者，可表示为：

$$\min_{\mathbf{A}_1, \dots, \mathbf{A}_n} \mathbb{E} \left[ (\mathbf{y} - \boldsymbol{\mu}_y)^T (\mathbf{y} - \boldsymbol{\mu}_y) \right] \quad (53)$$

首先，需要列出拉格朗日优化函数。优化目标已在公式 (52) 或公式 (53) 中列出。现在，考虑约束条件  $\sum_{i=1}^n \mathbf{A}_i = \mathbf{I}$ ，其为矩阵形式，需要将其转换为标量形式。为此，将约束条件转换为  $\sum_{i=1}^n \mathbf{A}_i - \mathbf{I} = \mathbf{0}$ ，且为该约束矩阵的每一个元素引入一个拉格朗日乘数。为方便起见，将这些拉格朗日乘数置于一个矩阵中，其维度与  $\mathbf{A}_i$  一样，命名为  $\boldsymbol{\Lambda}$ 。于是，带拉格朗日乘数的约束条件的标量形式即为矩阵的标准内积<sup>15</sup>：

$$\langle \boldsymbol{\Lambda}, \sum_{i=1}^n \mathbf{A}_i - \mathbf{I} \rangle = \text{tr} \left( \boldsymbol{\Lambda}^T \left( \sum_{i=1}^n \mathbf{A}_i - \mathbf{I} \right) \right) \quad (54)$$

<sup>14</sup>相关的推导，请参考文献 [6]。

<sup>15</sup>关于矩阵的标准内积定义，请阅读《机器学习》课程系列之“基础知识”的附录部分。

于是，得到如下的拉格朗日优化函数：

$$\begin{aligned}\mathcal{L}(\mathbf{A}_1, \dots, \mathbf{A}_n) &= \text{tr}(\Sigma_{\mathbf{y}\mathbf{y}}) + \text{tr}\left(\Lambda^T \left(\sum_{i=1}^n \mathbf{A}_i - \mathbf{I}\right)\right) \\ &= \text{tr}\left(\sum_{i=1}^n \mathbf{A}_i \Sigma_i \mathbf{A}_i^T\right) + \text{tr}\left(\Lambda^T \left(\sum_{i=1}^n \mathbf{A}_i - \mathbf{I}\right)\right)\end{aligned}\quad (55)$$

应用协方差矩阵  $\Sigma_i$  的对称性、标量函数对矩阵的求导方法以及迹技巧，可得<sup>16</sup>：

$$\begin{aligned}d\mathcal{L}(\mathbf{A}_1, \dots, \mathbf{A}_n) &= d\left(\text{tr}\left(\sum_{i=1}^n \mathbf{A}_i \Sigma_i \mathbf{A}_i^T\right) + \text{tr}\left(\Lambda^T \left(\sum_{i=1}^n \mathbf{A}_i - \mathbf{I}\right)\right)\right) \\ &= \text{tr}(d\mathbf{A}_i \Sigma_i \mathbf{A}_i^T) + \text{tr}(\mathbf{A}_i \Sigma_i d\mathbf{A}_i^T) + \text{tr}(\Lambda^T d\mathbf{A}_i) \\ &= \text{tr}(2d\mathbf{A}_i \Sigma_i \mathbf{A}_i^T) + \text{tr}(\Lambda^T d\mathbf{A}_i) \\ &= \text{tr}(2\Sigma_i \mathbf{A}_i^T d\mathbf{A}_i) + \text{tr}(\Lambda^T d\mathbf{A}_i)\end{aligned}\quad (56)$$

于是，可以得到：

$$\frac{\partial \mathcal{L}(\mathbf{A}_1, \dots, \mathbf{A}_n)}{\partial \mathbf{A}_i} = 2\mathbf{A}_i \Sigma_i + \Lambda = \mathbf{0}\quad (57)$$

进一步地，得到：

$$\mathbf{A}_1 \Sigma_1 = \mathbf{A}_2 \Sigma_2 = \dots = \mathbf{A}_n \Sigma_n = -\frac{\Lambda}{2}\quad (58)$$

则：

$$\begin{cases} \mathbf{A}_1 = -\frac{\Lambda}{2} \Sigma_1^{-1} \\ \mathbf{A}_2 = -\frac{\Lambda}{2} \Sigma_2^{-1} \\ \vdots \\ \mathbf{A}_n = -\frac{\Lambda}{2} \Sigma_n^{-1} \end{cases}\quad (59)$$

利用约束条件  $\sum_{i=1}^n \mathbf{A}_i = \mathbf{I}$ ，可得到：

$$\Lambda = -2\mathbf{I} (\Sigma_1^{-1} + \Sigma_2^{-1} + \dots + \Sigma_n^{-1})^{-1} = -2 \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1}\quad (60)$$

将上式代入公式 (58)，得到：

$$\mathbf{A}_i^* = \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \Sigma_i^{-1}\quad (61)$$

<sup>16</sup>关于“标量函数对矩阵的求导”的内容，请阅读《机器学习》课程系列之“判别函数的线性分类基础”的附录部分。

于是，最优融合量为：

$$\mathbf{y}^* = \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \sum_{i=1}^n \Sigma_i^{-1} \mathbf{x}_i \quad (62)$$

利用  $\Sigma_i$  为对称矩阵的事实，则  $\mathbf{y}^*$  的最优协方差矩阵为：

$$\begin{aligned} \Sigma_{yy}^* &= \sum_{i=1}^n \mathbf{A}_i^* \Sigma_i (\mathbf{A}_i^*)^T \\ &= \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \sum_{i=1}^n \Sigma_i^{-1} \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \\ &= \left( \sum_{j=1}^n \Sigma_j^{-1} \right)^{-1} \end{aligned} \quad (63)$$

类似地，引入精度矩阵  $\mathbf{V}_i = \Sigma_i^{-1}$ ，则最优精度矩阵及最优融合量分别为：

$$\begin{aligned} \mathbf{V}_{y^*} &= (\Sigma_{yy}^*)^{-1} = \sum_{j=1}^n \mathbf{V}_j \\ \mathbf{y}^* &= \mathbf{V}_{y^*}^{-1} \sum_{i=1}^n \mathbf{V}_i \mathbf{x}_i \end{aligned} \quad (64)$$

同样，最优精度矩阵的增量形式为：

$$\mathbf{V}_{y_n^*} = \sum_{j=1}^n \mathbf{V}_j = \mathbf{V}_{y_{n-1}^*} + \mathbf{V}_n \quad (65)$$

最优融合量的增量形式为：

$$\begin{aligned} \mathbf{y}_n^* &= \mathbf{V}_{y_n^*}^{-1} \sum_{i=1}^n \mathbf{V}_i \mathbf{x}_i \\ &= \left( \mathbf{V}_{y_{n-1}^*} + \mathbf{V}_n \right)^{-1} \left( \sum_{i=1}^{n-1} \mathbf{V}_i \mathbf{x}_i + \mathbf{V}_n \mathbf{x}_n \right) \\ &= \left( \mathbf{V}_{y_{n-1}^*} + \mathbf{V}_n \right)^{-1} \mathbf{V}_{y_{n-1}^*} \mathbf{V}_{y_{n-1}^*}^{-1} \sum_{i=1}^{n-1} \mathbf{V}_i \mathbf{x}_i + \left( \mathbf{V}_{y_{n-1}^*} + \mathbf{V}_n \right)^{-1} \mathbf{V}_n \mathbf{x}_n \\ &= \left( \mathbf{V}_{y_{n-1}^*} + \mathbf{V}_n \right)^{-1} \mathbf{V}_{y_{n-1}^*} \mathbf{y}_{n-1}^* + \left( \mathbf{V}_{y_{n-1}^*} + \mathbf{V}_n \right)^{-1} \mathbf{V}_n \mathbf{x}_n \end{aligned} \quad (66)$$

显然，如果估算量  $\mathbf{x}_i$  退化为标量，则上述推导出的所有公式均分别等价于其对应的标量公式。

## 4 练习

1. 以第2节中描述的示例为基础，试设计一个卡尔曼滤波应用场景，对某个运动的物体，使用卡尔曼滤波算法，预测其位置与速度，并绘制真实数据、观测数据及预测数据的曲线。

## 5 参考文献

1. Robert Grover Brown, Patrick Y. C. Hwang. Introduction to Random Signals and Applied Kalman Filtering with MATLAB Exercises(4Ed.). John Wiley & Sons, Inc., 2012.
2. Kalman, R. E. 1960. “A New Approach to Linear Filtering and Prediction Problems,” Transaction of the ASME—Journal of Basic Engineering, pp. 35-45 (March 1960).
3. Stuart J. Russell, Peter Norvig, 殷建平等译。《人工智能：一种现代的方法》，第3版，清华大学出版社。
4. Yan Pei, Swarnendu Biswas, Donald S. Fussell, Keshav Pingali. An Elementary Introduction to Kalman Filtering. <https://arxiv.org/abs/1710.04055>. [Submitted on 9 Oct 2017 (v1), last revised 27 Jun 2019 (this version, v5)].
5. Ramsey Faragher. Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation [Lecture Notes]. IEEE Signal Processing Magazine [128-132] September 2012, Digital Object Identifier 10.1109/M SP.2012.2203621 Date of publication: 20 August 2012.
6. Peter S Maybeck. 1982. Stochastic models, estimation, and control. Vol. 3. Academic press.
7. Greg Welch and Gary Bishop. An Introduction to the Kalman Filter. TR 95-041 Department of Computer Science University of North Carolina at Chapel Hill Chapel Hill, NC 27599-3175, Monday, July 24, 2006.

8. Kalman filter. [https://en.wikipedia.org/wiki/Kalman\\_filter#Kalman\\_gain\\_derivation](https://en.wikipedia.org/wiki/Kalman_filter#Kalman_gain_derivation).
9. How a Kalman filter works, in pictures. <http://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/>.
10. Unscented Kalman Filter. <https://baike.baidu.com/item/UKF/8576597>.