Data Fusion
An Intuitive Look

Dr Salman Aslam

Sequence

Sequence : detailed

Kalman Filter/Observer/Estimator

Introduction

- Named after Rudolf E. Kalman

Kalman Filter/Observer/Estimator

Introduction cont.

Kalman Filter/Observer/Estimator

Introduction cont.

- **Bigger picture**
  - The Kalman filter is a type of Observer
  - Another name for Observer is Estimator
- **Goal**
  - The goal of the Kalman filter is to estimate the states of a system
  - The states are represented as $\mathbf{x}$ while the estimated states are represented as $\hat{\mathbf{x}}$
- **Methodology**
  - Sensors are used to measure the input u and output y of a system
  - The Kalman filter takes an I/O approach, using the input u and output y to estimate what is going on inside the system, i.e., its states $\mathbf{x}$
  - If multiple sensors are used to measure y, then the Kalman filter can intelligently combine that information to give a 'good' estimate of the states $\mathbf{x}$

Kalman Filter/Observer/Estimator

Introduction cont.

- **Typical uses**
  - Smoothing of noisy data
  - Providing estimates of parameters of interest
- **Some applications**
  - GPS receivers
  - PLLs in radio equipment
  - Smoothing the output from laptop trackpads
  - Tracking objects (eg missiles, faces, heads, hands)
  - Economics
  - Navigation
  - Fusing data from radar, laser scanners and stereo-cameras
  - Smart phones
  - Computer games
  - Weather analysis
- The most famous early use of the Kalman filter was in the Apollo navigation computer that took Neil Armstrong to the moon in 1969 and brought him back

[0] www.cs.cornell.edu/Courses/cs4758/2013sp/materials/MI63slides.pdf

Kalman Filter/Observer/Estimator

Introduction cont.

- From a theoretical standpoint, the Kalman filter is an algorithm permitting exact inference in a linear dynamical system, which is a Bayesian model similar to a Hidden Markov Model (HMM) but where the state space of the latent variables is continuous and where all latent and observed variables have a Gaussian distribution (often a multivariate Gaussian distribution)
- The Kalman filter is based on 5 steps with one equation

  per step:

  1. Prediction
  2. Prediction uncertainty
  3. Computation of Kalman gain
  4. Update (using measurement)
  5. Update uncertainty

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

Derivation

- The Kalman filter is typically derived using vector algebra as a MMSE (minimum mean squared estimator), an approach suitable for students confident in mathematics but not one that is easy to grasp for students in disciplines that do not require strong mathematics

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

Derivation cont.

- From here on, we focus our attention to a very well written paper on the Kalman filter, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*
- This paper provides a simple and intuitive derivation of the Kalman filter with the aim of teaching this useful tool to students from disciplines that do not require a strong mathematical background
- The most complicated level of mathematics required to understand this derivation is the ability to multiply two Gaussian functions together and reduce the result to a compact form
- In this paper, the Kalman filter is derived from first principles considering a simple physical system exploiting a key property of the Gaussian distribution - specifically the property that the product of two Gaussian distributions is another Gaussian distribution [function]

Kalman Filter/Observer/Estimator

1D derivation (same units)

- In the following slides, we make two assumptions for the derivation:
  1. All variables are scalars, i.e., in 1D (an example application is motion along x-axis only)
  2. The units for prediction, measurement and fused output are all the same
- An example could be that IMU is used for prediction and GPS is used for measurement
- The system equation can be modeled as:

$$\dot{x} = Ax + Bu + w \quad (w \text{ is called process noise})$$

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D derivation (same units)

We have the following variables in the Kalman filter:

No	CATEGORY	NAME	SYMBOL
1	INPUT	Input signal	u
2	PLANT	System Transition Matrix	A
	"	Control Input Matrix	B
	"	Transformation Matrix	C
3	SENSOR	Measurement mean	μ_2
	"	Measurement variance	σ_2^2
4	OBSERVER	Prediction mean	μ_1
	"	Prediction variance	σ_1^2
	"	Process noise variance	Q
	"	Kalman gain	K
	"	Updated mean	μ_{fused}
	"	Updated variance	σ_{fused}^2

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D derivation (same units) cont.

Step 1: prediction

$$\dot{\hat{x}} = A\hat{x} + Bu$$

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D derivation (same units) cont.

Step 2: uncertainty of prediction

$$
\begin{aligned}
e = \dot{x} - \dot{\hat{x}} &= (Ax + Bu + w) - (A\hat{x} + Bu) \quad \text{(prediction error)} \\
&= A(x - \hat{x}) + w \\
\mathrm{COV}[e] &= E[(e - \mu_e)(e - \mu_e)^T] \quad \text{(covariance of prediction error)} \\
&= E[ee^T] \\
&= E\left[\left(\dot{x} - \dot{\hat{x}}\right)\left(\dot{x} - \dot{\hat{x}}\right)^T\right] \\
&= E\left[\left(A(x - \hat{x}) + w\right)\left(A(x - \hat{x}) + w\right)^T\right] \\
&= E\left[\left(A(x - \hat{x}) + w\right)\left((A(x - \hat{x}))^T + w^T\right)\right] \\
&= E\left[\left(A(x - \hat{x}) + w\right)\left((x - \hat{x})^T A^T + w^T\right)\right] \\
&= E\Big[A(x - \hat{x})(x - \hat{x})^T A^T + A(x - \hat{x})w^T \\
&\quad + w(x - \hat{x})^T A^T + ww^T\Big] \\
&= AE[(x - \hat{x})(x - \hat{x})^T]A^T + E[ww^T]
\end{aligned}
$$

Kalman Filter/Observer/Estimator

1D derivation (same units) cont.

Steps 3 and 4: Kalman gain and update

The mean μ_{fused} of the Gaussian function obtained when two Gaussian distributions with mean and variance μ_1, σ_1^2 and μ_2, σ_2^2 respectively are multiplied is given by,

$$\begin{aligned} \mu_{fused} &= \frac{\mu_1\sigma_2^2+\mu_2\sigma_1^2}{\sigma_1^2+\sigma_2^2} \\ &= \mu_1 + \frac{\sigma_1^2(\mu_2-\mu_1)}{\sigma_1^2+\sigma_2^2} &= \mu_1 + k_1(\mu_2 - \mu_1) \\ &= \mu_2 + \frac{\sigma_2^2(\mu_1-\mu_2)}{\sigma_1^2+\sigma_2^2} &= \mu_2 + k_2(\mu_1 - \mu_2) \end{aligned}$$

where $k_1 = \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}$ and $k_2 = \frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}$

- Notice that k_1 and k_2 are always positive fractions
- The mean μ_{fused} is obtained by:
  - offsetting μ_1 with the weighted difference of $\mu_2 - \mu_1$, or
  - offsetting μ_2 with the weighted difference of $\mu_1 - \mu_2$
- It is more common to see $\mu_{fused} = \mu_1 + k_1(\mu_2 - \mu_1)$ than to see $\mu_{fused} = \mu_2 + k_2(\mu_1 - \mu_2)$ although of course, both are correct

Kalman Filter/Observer/Estimator

1D derivation (same units) cont.

Step 5: uncertainty of update

The variance σ^2_{fused} of the Gaussian function obtained when two Gaussian distributions with mean and variance μ_1, σ_1^2 and μ_2, σ_2^2 respectively are multiplied is given by,

$$\begin{aligned}
\sigma^2_{fused} &= \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2} \\
&= \sigma_1^2 - \frac{\sigma_1^4}{\sigma_1^2+\sigma_2^2} = \sigma_1^2 - \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}\sigma_1^2 = (1-k_1)\sigma_1^2 \\
&= \sigma_2^2 - \frac{\sigma_2^4}{\sigma_1^2+\sigma_2^2} = \sigma_2^2 - \frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}\sigma_2^2 = (1-k_2)\sigma_2^2
\end{aligned}$$

where k_1 and k_2 are the same as defined in the previous slide

- Since k_1 and k_2 are always positive fractions, the variance σ^2_{fused} of the output Gaussian function is always less than σ_1^2 and σ_2^2
- In other words, multiplying two Gaussians decreases the variance of the resulting Gaussian to less than the variances of the input Gaussians
- It is more common to see $\sigma^2_{fused} = (1-k_1)\sigma_1^2$ than to see $\sigma^2_{fused} = (1-k_2)\sigma_2^2$ although of course, both are correct

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE

Kalman Filter/Observer/Estimator

1D derivation (different units)

- Now, we assume that the units of the prediction
  and fused output are the same while the units of the measurement are different
- However, we want all units to be the same as for the measurement
- Assume that the conversion can be obtained by multiplying with the factor C
- We now repeat the previous derivation using this unit conversion

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D derivation (different units) cont.

Step 1: prediction
Same as before

---

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D derivation (different units) cont.

Step 2: uncertainty of prediction
Same as before

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D derivation (different units) cont.

Steps 3 and 4: Kalman gain and update
We convert the prediction and fused output units to measurement units as follows,

$$
\begin{aligned}
C\mu_{fused} &= \frac{C\mu_1\sigma_2^2+\mu_2C^2\sigma_1^2}{C^2\sigma_1^2+\sigma_2^2} \\
\Rightarrow \mu_{fused} &= \frac{\mu_1\sigma_2^2+\mu_2C\sigma_1^2}{C^2\sigma_1^2+\sigma_2^2} \\
&= \mu_1+\frac{\sigma_1^2(C\mu_2-C^2\mu_1)}{C^2\sigma_1^2+\sigma_2^2} = \mu_1+\frac{C\sigma_1^2(\mu_2-C\mu_1)}{C^2\sigma_1^2+\sigma_2^2} = \mu_1+K(\mu_2-C\mu_1)
\end{aligned}
$$

where $K=\frac{C\sigma_1^2}{C^2\sigma_1^2+\sigma_2^2}$

- The mean μ_{fused} is obtained by offsetting μ_1 with the weighted difference of $\mu_2-C\mu_1$
- This weighting K is called the Kalman gain

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D derivation (different units) cont.

Step 5: uncertainty of update

We convert the prediction and fused output units to measurement units as follows,

$$
\begin{aligned}
C^2\sigma^2_{fused} &= \frac{C^2\sigma_1^2\sigma_2^2}{C^2\sigma_1^2+\sigma_2^2} \\
\Rightarrow \sigma^2_{fused} &= \frac{\sigma_1^2\sigma_2^2}{C^2\sigma_1^2+\sigma_2^2} \\
&= \sigma_1^2 - \frac{C^2\sigma_1^4}{C^2\sigma_1^2+\sigma_2^2} \quad = \sigma_1^2 - \frac{C\sigma_1^2}{C^2\sigma_1^2+\sigma_2^2}C\sigma_1^2 \quad = (1-KC)\sigma_1^2
\end{aligned}
$$

where K is the Kalman gain as defined in the previous slide

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

So, in summary, the equations are:

single state

from previous time

I. PREDICTION

1. state

$$\mu_1 = A\mu_{fused} + Bu$$

2. state variance/covariance

$$\sigma_1^2 = A\sigma_{fused}^2 A + Q = A^2\sigma_{fused}^2 + Q$$

II. UPDATE

3. state

$$\mu_{fused} = \mu_1 + K(\mu_2 - C\mu_1)$$

$$K = \sigma_1^2 C(C\sigma_1^2 C + \sigma_2^2)^{-1} = \frac{C\sigma_1^2}{C^2\sigma_1^2+\sigma_2^2}$$

4. state variance/covariance

$$\sigma_{fused}^2 = (1 - KC)\sigma_1^2$$

prediction

measurement

update

- The predicted state is μ_1 and its variance is σ_1^2
- The measurement (observation) is μ_2 and its variance is σ_2^2
- The prediction and measurement are combined using the Kalman gain K

Kalman Filter/Observer/Estimator

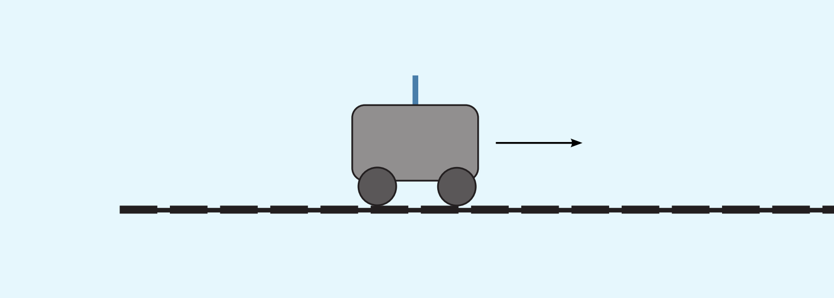
1D derivation (different units) cont.

- Step 1: The future state is predicted according to the system model (or through a sensor)
- Step 2: If a random variable is multiplied with a constant A, then the resulting variance is multiplied with constant A^2. In addition, the resulting variance is increased by Q, i.e., we have added more uncertainty.
- Step 3: The Kalman gain is computed. This is the gain of the observer and although we use K, this is actually L. Also notice that,

$$\begin{aligned} K &= \frac{C\sigma_1^2}{C^2\sigma_1^2+\sigma_2^2} \\ \Rightarrow \frac{KC^2}{C} &= \frac{\sigma_1^2}{\sigma_1^2+\frac{1}{C^2}\sigma_2^2} \\ \Rightarrow KC &= \frac{\sigma_1^2}{\sigma_1^2+\frac{1}{C^2}\sigma_2^2} < 1 \\ \Rightarrow \sigma_{fused}^2 &< \sigma_1^2 \end{aligned}$$

- Step 4: The difference between actual and expected observation is weighted by K and added to our predicted estimate
- Step 5: The variance of the update decreases from σ_1^2 to $(1-KC)\sigma_1^2$, a key strength of the Kalman filter ! Also, if σ^2 goes up, so does K. But, if σ^2 goes

- **Introduction**
  - We start by considering a simple 1-D problem, i.e., we have only one state
  - We have a train moving along a railway line, as shown below
- **Goal**
  - The goal is for an onboard computer to correctly estimate the position of the train

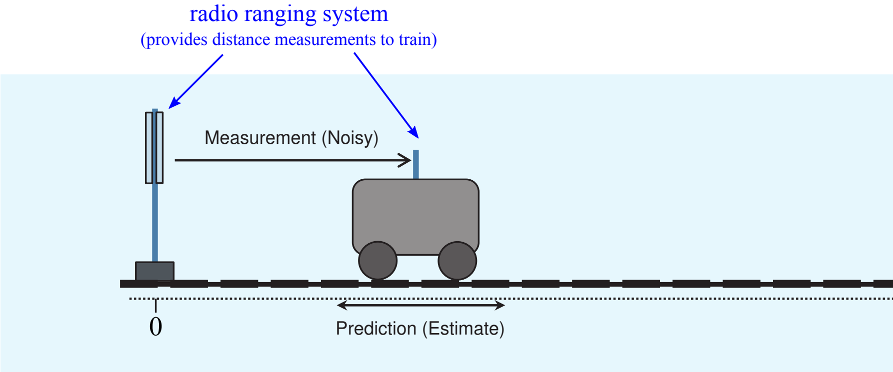


[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D example cont.

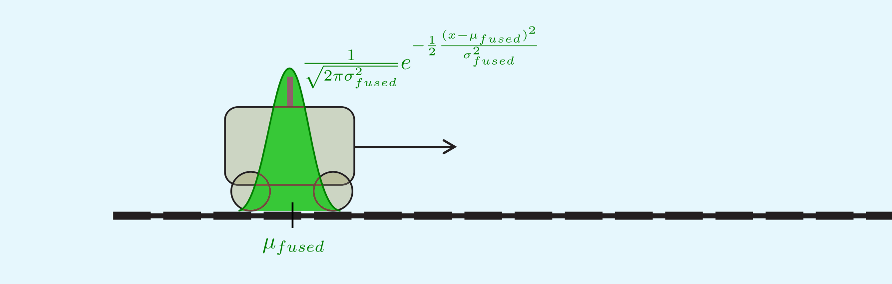
1. **Input**: The train driver may apply a *braking* or *accelerating* input to the train
2. **Plant (System)**: Since we have only one state, position of the train along the x axis, we use a single state model, $x[k] = Ax[k-1] + Bu[k-1]$
3. **Sensor**: The train can get distance measurements from a radio ranging system (similar to Distance Measuring Equipment, DME, in aircraft)
4. **Compensator**:
   - **Controller**: None
   - **Observer**: Kalman filter



[FC1] This figure shows the one dimensional system under consideration

Initialization

- At some initial time, we have some idea of where the train is
- This idea is represented by a Gaussian distribution (green curve below)
- At places where the Gaussian distribution has high amplitude, we have more confidence
- Therefore, most likely, the train is at μ_{fused}
- The bigger that σ^2_{fused} is, the more spread the Gaussian distribution, and the less our confidence of where the train really is

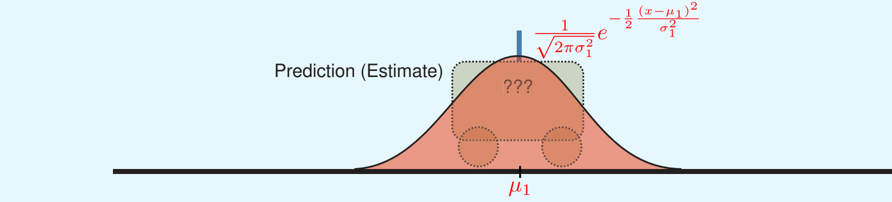


[FIG2] The initial knowledge of the system at time $t = 0$. The green Gaussian distribution represents the pdf confidence in the estimate of the position of the train. The arrow pointing to the right represents the kn the train.

0 Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Step 1 of the Kalman Filter: prediction

- Now, we will predict the new position of the train
- The Gaussian is now more spread
- Our maximum belief is in the train being at μ_1

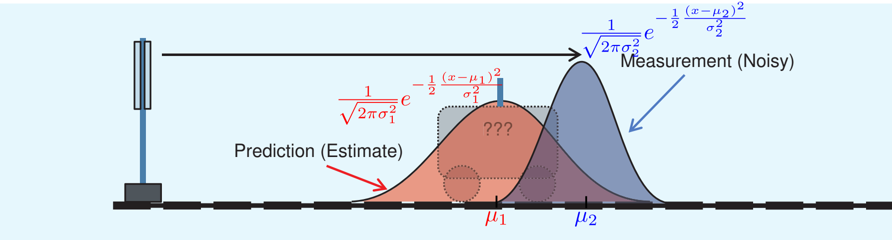


[FIG3] Here, the prediction of the location of the train at time $t = 1$ and the level of uncertainty in that pr confidence in the knowledge of the position of the train has decreased, as we are not certain if the train accelerations or decelerations in the intervening period from $t = 0$ to $t = 1$.

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Intermediate step: measurement arrives

- Now, we get a measurement from the radio ranging system, which is also noisy
- According to the radio ranging system, the train is most likely at μ_2
- Now, we have to combine the prediction (red Gaussian) with the measurement (blue Gaussian) to come to a more intelligent decision
- This is known as *sensor fusion*



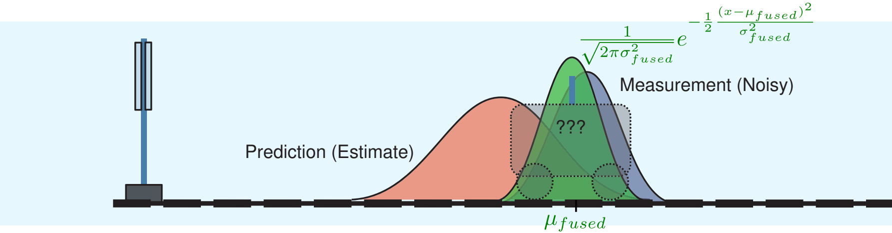
[FIG4] Shows the measurement of the location of the train at time $t = 1$ and the level of uncertainty in th represented by the blue Gaussian pdf. The combined knowledge of this system is provided by multiplyir together.

---

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Step 2 of the Kalman Filter: update

- We combine the prediction Gaussian (red) with the measurement Gaussian (blue) by multiplying them
- This gives us the green Gaussian
- We call the peak of the green Gaussian μ_{fused}
- So, we have decided that the train is most likely at μ_{fused}, but of course we're not sure, and that unsurety is shown by the green Gaussian
- Notice that the green Gaussian is less spread than the red Gaussian or the blue Gaussian showing that we are more sure now than before of the position of the train



[FIG5] Shows the new pdf (green) generated by multiplying the pdfs associated with the prediction and train's location at time $t = 1$. This new pdf provides the best estimate of the location of the train, by fusi prediction and the measurement.

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012

Kalman Filter/Observer/Estimator

1D example cont.

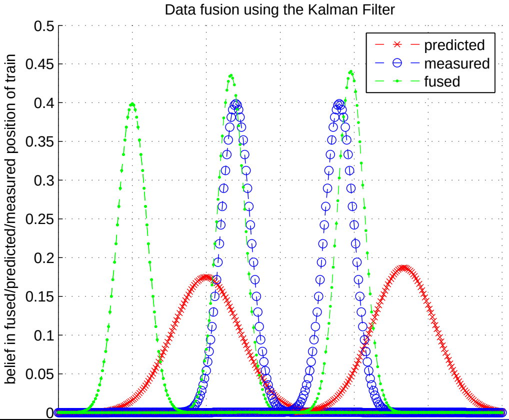
The code below implements 2 time steps for the train:

```
clear;clc;clf;
%initialization
x         =   -5:0.1:25;  %x axis
A         =   2;          %state transition matrix
B         =   5;          %control input matrix
C         =   1;          %transformation matrix
u         =   1;          %control input
Q         =   1.2;        %process noise variance, adds uncertainty to prediction
mu_2      =   [7 14];     %the output of the radio ranging system at next two times
var_2     =   1;          %the variance of the radio ranging system given by manufacturer
mu_f      =   0;          %the estimated location of the train at current time, k=0
var_f     =   1;          %the estimated variance of our estimate at current time, k=0
pdf_0     =   (1/sqrt(2*pi*var_f))*exp(-(0.5/var_f)*(x-mu_f).^2);
%=====================================
%run Kalman Filter
%=====================================
for k=1:2
    %predict
    mu_1      =   A*mu_f     +   B*u;
    var_1     =   A^2*var_f  +   Q;
    %update
    K         =   (C*var_1)/(C^2*var_1 + var_2);
    mu_f      =   mu_1 + K*(mu_2(k)-C*mu_1);
    var_f     =   (1-K*C)*var_1;
    pred      =   (1/sqrt(2*pi*var_1))*exp(-(0.5/var_1)*(x-mu_1).^2);
    meas      =   (1/sqrt(2*pi*var_2))*exp(-(0.5/var_2)*(x-mu_2(k)).^2);
    fused     =   (1/sqrt(2*pi*var_f))*exp(-(0.5/var_f)*(x-mu_f).^2);
                  axis([-5 25 0 0.5]);
```

Kalman Filter/Observer/Estimator

1D example cont.

The code on the previous slide produces this output:



Kalman Filter/Observer/Estimator

Multiple-D derivation

- Now, let's take a look at the Kalman Filter when more than one state is involved
- Notice the one-to-one correspondence between equations in 1D (i.e. one state) and multiple states

	single state	multiple
	from previous time	from previous
I. PREDICTION		
1. state	$\mu_1 = A\mu_{fused} + Bu$	$\hat{\mathbf{x}}_{k\|k-1} = \mathbf{A}\hat{\mathbf{x}}_{k-1\|k-1} + \mathbf{Bu}$
2. state variance/covariance	$\sigma_1^2 = A\sigma_{fused}^2 A + Q = A^2\sigma_{fused}^2 + Q$	$\mathbf{P}_{k\|k-1} = \mathbf{A}\mathbf{P}_{k-1\|k-1}\mathbf{A}^T +$
II. UPDATE		
3. state	$\mu_{fused} = \mu_1 + K(\mu_2 - C\mu_1)$	$\hat{\mathbf{x}}_{k\|k} = \hat{\mathbf{x}}_{k\|k-1} + \mathbf{K}_k(\mathbf{y}_k -$
	$K = \sigma_1^2 C(C\sigma_1^2 C + \sigma_2^2)^{-1} = \frac{C\sigma_1^2}{C^2\sigma_1^2 + \sigma_2^2}$	innovation
		$\mathbf{K}_k = \mathbf{P}_{k\|k-}$
4. state variance/covariance	$\sigma_{fused}^2 = (1 - KC)\sigma_1^2$	$\mathbf{P}_{k\|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{C})\mathbf{P}_{k\|k-1}$

prediction
measurement
update

[0] http://en.wikipedia.org/wiki/Kalman_filter

Kalman Filter/Observer/Estimator

2D example

- In the single state example, we used the Kalman filter to fuse, i.e., combine two pieces of information: first, our prediction of the position of a train, and second, the measurement of the train's position obtained from a radio ranging system. The result of our fusion was an estimate of the train's position.
- Now, we will extend the Kalman Filter to 2 states (also called 2 dimensions, or 2D), i.e., we will focus on a car that moves in the xy-plane and not just in a straight line along the x-axis like a train
- Here is an example in 2D followed by its code on the next 3 slides

		k=0	k=1
Reality $\mathbf{A}_{real} = \begin{bmatrix} 0.5 & 0.1 \\ 0.3 & 0.2 \end{bmatrix}$ $\mathbf{B}_{real} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	$\mathbf{x}_{real} \longrightarrow$		$\begin{bmatrix} 1.0000 \\ 1.0000 \end{bmatrix}$
Model $\mathbf{A} = \begin{bmatrix} 0.4 & 0.2 \\ 0.3 & 0.35 \end{bmatrix}$ $\mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ $\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ $\mathbf{D} = [0]$ $\mathbf{Q} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$ $\mathbf{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$u \longrightarrow$		1.0000
Initial values $\hat{\mathbf{x}}_{0\|0} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ $\mathbf{P}_{0\|0} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$	$\mathbf{y} \longrightarrow$		$\begin{bmatrix} 1.2000 \\ 1.1000 \end{bmatrix}$
1. predicted estimate: state	$\hat{\mathbf{x}}_{k\|k-1} = \mathbf{A}\hat{\mathbf{x}}_{k-1\|k-1} + \mathbf{B}\mathbf{u}_{k-1}$		$\begin{bmatrix} 1.0000 \\ 1.0000 \end{bmatrix}$
2. predicted estimate: state-covariance	$\mathbf{P}_{k\|k-1} = \mathbf{A}\mathbf{P}_{k-1\|k-1}\mathbf{A}^T + \mathbf{Q}$		$\begin{bmatrix} 2.4000 & 0.3800 \\ 0.3800 & 2.4250 \end{bmatrix}$
3. gain	$\mathbf{K}_k = \mathbf{P}_{k\|k-1}\mathbf{C}^T \Big(\underbrace{\mathbf{C}\mathbf{P}_{k\|k-1}\mathbf{C}^T + \mathbf{R}}_{\text{innovation: covariance}} \Big)^{-1}$		$\begin{bmatrix} 0.7022 & 0.0330 \\ 0.0330 & 0.7044 \end{bmatrix}$
4. updated estimate: state	$\hat{\mathbf{x}}_{k\|k} = \hat{\mathbf{x}}_{k\|k-1} + \mathbf{K}_k \left(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k\|k-1} \right)$	$\begin{bmatrix} 0.0000 \\ 0.0000 \end{bmatrix}$	$\begin{bmatrix} 1.1437 \\ 1.0770 \end{bmatrix}$

Kalman Filter/Observer/Estimator

2D example cont.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%INITIALIZATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear; clc; clf; format compact;
%total time
end_time        =   3;                  %number of iterations
N               =   2;                  %number of states
%-----------------------------------------
%real system
%-----------------------------------------
A_real          =   [0.5    0.1;
                     0.3    0.2];       %Transition Matrix      (real)
B_real          =   [1;                 %Control Input Matrix   (real)
                     1];
%estimated system
A               =   [0.4    0.2;
                     0.3    0.35];      %Transition Matrix      (estimated)
B               =   [1;                 %Control Input Matrix   (estimated)
                     1];
C               =   eye(N);             %Measurement Matrix     (estimated)
%input
u               =   ones(1,end_time);   %Step input
%states (initial values)
xhat            =   [0;
                     0];
%prediction errors (initial values)
Q               =   [2    0;            %add uncertainty while predicting
                     0    2];
P               =   Q;                  %total uncertainty of predicted state
%measurements (all values)
```

Kalman Filter/Observer/Estimator

2D example cont.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%create real states
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
x_real          =   xhat;
for k           =   1:end_time
    x_real      =   A_real * x_real + B_real*u(k);
    iter_real   =   [iter_real x_real];
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%estimate states using Kalman Filter
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for k           =   1:end_time
    %prediction
    xhat        =   A * xhat + B*u(k);                 %step 1
    iter_predx  =   [iter_predx xhat];
    P           =   A * P * A' + Q;                    %step 2
    iter_predP  =   [iter_predP P];
    %gain
    K           =   P * C' * inv(C * P * C' + R);
    iter_K      =   [iter_K K];
    %update
    xhat        =   xhat + K * (y(:,k) - C * xhat); %step 3
    iter_fusedx  =   [iter_fusedx xhat];
    P           =   (eye(N)- K * C) * P;               %step 4
    iter_fusedP  =   [iter_fusedP P];
end
sprintf('The MSE for prediction      : %.2f',norm((iter_real-iter_predx), 2))
sprintf('The MSE for measurements   : %.2f',norm((iter_real-y), 2))
sprintf('The MSE for fused estimate : %.2f',norm((iter_real-iter_fusedx), 2))
```

Kalman Filter/Observer/Estimator

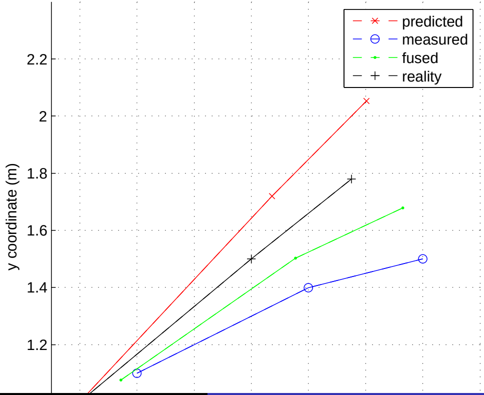
2D example cont.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%RESULTS
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
hold on;
grid on;
%plot points
plot(iter_predx(1,:), iter_predx(2,:), 'r--x');          %predicted
plot(y(1,:), y(2,:), 'b--o');                             %measured
plot(iter_fusedx(1,:), iter_fusedx(2,:), 'g-.');         %fused
plot(iter_real(1,:), iter_real(2,:), 'k--+');            %reality
%plot lines
line(iter_predx(1,:), iter_predx(2,:), 'Color', 'r');    %predicted
line(y(1,:), y(2,:), 'Color', 'b');                       %measured
line(iter_fusedx(1,:), iter_fusedx(2,:), 'Color', 'g'); %fused
line(iter_real(1,:), iter_real(2,:), 'Color', 'k');      %reality
xlabel('x coordinate (m)');
ylabel('y coordinate (m)');
axis equal
axis([0.9 2.4 0.9 2.4]);
legend('predicted', 'measured', 'fused', 'reality');
```

Kalman Filter/Observer/Estimator

2D example cont.

The code on the previous slide produces this output:



Kalman Filter/Observer/Estimator

Product of Gaussian distributions is a Gaussian function

Let p_1 and p_2 be two Gaussian pdfs with the same support:

$$p_1(x;\mu_1,\sigma_1) = \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}}$$

$$p_2(x;\mu_2,\sigma_2) = \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{(x-\mu_2)^2}{2\sigma_2^2}}$$

Their product is,

$$p_1 p_2 = \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}} \frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{(x-\mu_2)^2}{2\sigma_2^2}}$$

0 http://blog.jafma.net/2010/11/09/the-product-of-two-Gaussian-pdfs-is-not-a-pdf-but-is-Gaussian-a-k-a-loving-algebra/

Kalman Filter/Observer/Estimator

Product of Gaussian distributions is a Gaussian function

combining exponents,

$$= \frac{1}{\sqrt{2\pi\sigma_1^2\sigma_2^2 2\pi}} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2} - \frac{(x-\mu_2)^2}{2\sigma_2^2}}$$

taking LCM,

$$= \frac{1}{\sqrt{2\pi\sigma_1^2\sigma_2^2 2\pi}} e^{\frac{-\sigma_2^2(x-\mu_1)^2 - \sigma_1^2(x-\mu_2)^2}{2\sigma_1^2\sigma_2^2}}$$

[0] http://blog.jafma.net/2010/11/09/the-product-of-two-Gaussian-pdfs-is-not-a-pdf-but-is-Gaussian-a-k-a-loving-algebra/

Kalman Filter/Observer/Estimator

Product of Gaussian distributions is a Gaussian function

Multiply and divide by $\sigma_1^2 + \sigma_2^2$. This is possible because $\sigma_1^2 + \sigma_2^2 > 0$ (otherwise we would be talking about the product of two Dirac delta functions, which would be 0 unless $\mu_1 = \mu_2$)

$$p_1 p_2 = \frac{1}{\sqrt{2\pi \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}(\sigma_1^2+\sigma_2^2)2\pi}} e^{\frac{-\frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}(x-\mu_1)^2-\frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}(x-\mu_2)^2}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}}$$

Now we can separate $\frac{1}{\sqrt{2\pi(\sigma_1^2+\sigma_2^2)}}$ from the non-exponent part which resembles the scale factor of a Gaussian pdf, and then raise it to $e^{\ln}$,

$$p_1 p_2 = \frac{1}{\sqrt{2\pi \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}} \frac{1}{\sqrt{2\pi(\sigma_1^2+\sigma_2^2)}} e^{\frac{-\frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}(x-\mu_1)^2-\frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}(x-\mu_2)^2}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}}$$

Kalman Filter/Observer/Estimator

Product of Gaussian distributions is a Gaussian function

Combine exponents,

$$= \frac{1}{\sqrt{2\pi \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2+\sigma_2^2}}} e^{\ln \frac{1}{\sqrt{2\pi(\sigma_1^2+\sigma_2^2)}} + \frac{-\frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}(x-\mu_1)^2 - \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}(x-\mu_2)^2}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}}$$

replace square root in denominator of ln with -1/2,

$$= \frac{1}{\sqrt{2\pi \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2+\sigma_2^2}}} e^{-\frac{1}{2}\ln(2\pi(\sigma_1^2+\sigma_2^2)) + \frac{-\frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}(x-\mu_1)^2 - \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}(x-\mu_2)^2}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}}$$

Ok, it is time to focus only on the exponent of e.
Take LCM,

$$-\left(\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}\right)\ln(2\pi(\sigma_1^2+\sigma_2^2)) - \frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}(x-\mu_1)^2 - \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}(x-\mu_2)^2$$

Then we expand the square terms that include x and let $\alpha = \ln(2\pi(\sigma_1^2 + \sigma_2^2))$,

$$\frac{-(\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2})\alpha - \frac{\sigma_2^2}{\sigma_1^2+\sigma_2^2}(x^2+{\mu_1}^2-2x\mu_1) - \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}(x^2+{\mu_2}^2-2x\mu_2)}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}$$

And then we collect the numerator as a polynomial in x:

$$= \frac{x^2(\frac{-\sigma_1^2-\sigma_2^2}{\sigma_1^2+\sigma_2^2}) - 2(\frac{-\mu_1\sigma_2^2-\mu_2\sigma_1^2}{\sigma_1^2+\sigma_2^2})x + (\frac{-{\mu_1}^2\sigma_2^2-{\mu_2}^2\sigma_1^2-\sigma_1^2\sigma_2^2\alpha}{\sigma_1^2+\sigma_2^2})}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}$$

$$= \frac{-x^2+2(\frac{\mu_1\sigma_2^2+\mu_2\sigma_1^2}{\sigma_1^2+\sigma_2^2})x - (\frac{{\mu_1}^2\sigma_2^2+{\mu_2}^2\sigma_1^2+\sigma_1^2\sigma_2^2\alpha}{\sigma_1^2+\sigma_2^2})}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}$$

$$= \frac{-x^2+2Ax-(A^2+C)}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}$$

$$= \frac{-(x-A)^2-C}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}$$

[0] http://blog.jafma.net/2010/11/09/the-product-of-two-Gaussian-pdfs-is-not-a-pdf-but-is-Gaussian-a-k-a-loving-algebra/

where,

$$A = \frac{\mu_1\sigma_2^2+\mu_2\sigma_1^2}{\sigma_1^2+\sigma_2^2} \quad \text{and} \quad A^2 + C = \frac{{\mu_1}^2\sigma_2^2+{\mu_2}^2\sigma_1^2+\sigma_1^2\sigma_2^2\alpha}{\sigma_1^2+\sigma_2^2}$$

Then,

$$\begin{aligned}
C &= \frac{{\mu_1}^2\sigma_2^2+{\mu_2}^2\sigma_1^2+\sigma_1^2\sigma_2^2\alpha}{\sigma_1^2+\sigma_2^2} - A^2 \\
&= \frac{{\mu_1}^2\sigma_2^2+{\mu_2}^2\sigma_1^2+\sigma_1^2\sigma_2^2\alpha}{\sigma_1^2+\sigma_2^2} - \left(\frac{\mu_1\sigma_2^2+\mu_2\sigma_1^2}{\sigma_1^2+\sigma_2^2}\right)^2 \\
&= \frac{({\mu_1}^2\sigma_2^2+{\mu_2}^2\sigma_1^2+\sigma_1^2\sigma_2^2\alpha)(\sigma_1^2+\sigma_2^2)-(\mu_1\sigma_2^2+\mu_2\sigma_1^2)^2}{(\sigma_1^2+\sigma_2^2)^2} \\
&= \frac{{\mu_1}^2{\sigma_2}^4+{\mu_2}^2\sigma_1^2\sigma_2^2+\sigma_1^2{\sigma_2}^4\alpha+{\mu_1}^2\sigma_2^2\sigma_1^2+{\mu_2}^2{\sigma_1}^4+{\sigma_1}^4\sigma_2^2\alpha-{\mu_1}^2{\sigma_2}^4-{\mu_2}^2{\sigma_1}^4-2\mu_1\mu_2\sigma_1^2\sigma_2^2}{(\sigma_1^2+\sigma_2^2)^2} \\
&= \frac{{\mu_2}^2\sigma_1^2\sigma_2^2+\sigma_1^2{\sigma_2}^4\alpha+{\mu_1}^2\sigma_2^2\sigma_1^2+{\sigma_1}^4\sigma_2^2\alpha-2\mu_1\mu_2\sigma_1^2\sigma_2^2}{(\sigma_1^2+\sigma_2^2)^2} \\
&= \frac{\sigma_1^2\sigma_2^2({\mu_2}^2+\sigma_2^2\alpha+{\mu_1}^2+\sigma_1^2\alpha-2\mu_1\mu_2)}{(\sigma_1^2+\sigma_2^2)^2} \\
&= \frac{\sigma_1^2\sigma_2^2((\mu_1-\mu_2)^2+(\sigma_1^2+\sigma_2^2)\alpha)}{(\sigma_1^2+\sigma_2^2)^2} \\
&= \frac{\sigma_1^2\sigma_2^2}{(\sigma_1^2+\sigma_2^2)^2}(\mu_1-\mu_2)^2 + \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}\ln(2\pi(\sigma_1^2+\sigma_2^2)))
\end{aligned}$$

---

[0] http://blog.jafma.net/2010/11/09/the-product-of-two-Gaussian-pdfs-is-not-a-pdf-but-is-Gaussian-a-k-a-loving-algebra/

Ook. We have already values for A and C. Using those letters, the resulting product of the two Gaussian pdfs has become:

$$
\begin{aligned}
p_1 p_2 &= \frac{1}{\sqrt{2\pi \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2+\sigma_2^2}}} e^{\frac{-(x-A)^2 - C}{2\frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2+\sigma_2^2}}} \\
&= \frac{1}{\sqrt{2\pi \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2+\sigma_2^2}}} e^{\frac{-(x-A)^2}{2\frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2+\sigma_2^2}}} e^{-\frac{C}{2\frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2+\sigma_2^2}}}
\end{aligned}
$$

[0] http://blog.jafma.net/2010/11/09/the-product-of-two-Gaussian-pdfs-is-not-a-pdf-but-is-Gaussian-a-k-a-loving-algebra/

Notice that the first factor and the first exponential are a Gaussian pdf. The whole result, however, is just a Gaussian function, unless the second exponential equals 1
Let us take a closer look to that second exponential, the reason why our result is not a Gaussian pdf but a Gaussian function:

$$e^{-\frac{C}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}} = e^{-\frac{\frac{\sigma_1^2\sigma_2^2}{(\sigma_1^2+\sigma_2^2)^2}(\mu_1-\mu_2)^2+\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}\ln(2\pi(\sigma_1^2+\sigma_2^2))}{2\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}}}$$

$$= e^{-\frac{(\mu_1-\mu_2)^2}{2(\sigma_1^2+\sigma_2^2)}-\frac{1}{2}\ln(2\pi(\sigma_1^2+\sigma_2^2))}$$

$$= e^{-\frac{(\mu_1-\mu_2)^2}{2(\sigma_1^2+\sigma_2^2)}} e^{-\frac{1}{2}\ln(2\pi(\sigma_1^2+\sigma_2^2))}$$

$$= \frac{1}{\sqrt{2\pi(\sigma_1^2+\sigma_2^2)}} e^{-\frac{(\mu_1-\mu_2)^2}{2(\sigma_1^2+\sigma_2^2)}}$$

---

[0] http://blog.jafma.net/2010/11/09/the-product-of-two-Gaussian-pdfs-is-not-a-pdf-but-is-Gaussian-a-k-a-loving-algebra/

Surprisingly, this last expression is a Gaussian pdf if we consider μ_1 a variable, i.e., $p(\mu_1; \mu_2, \sqrt{(\sigma_1^2 + \sigma_2^2)})$ (we can also consider μ_2 the variable). But what we are interested in is knowing under which conditions this expression equals 1, and, thus, the product of our original pdfs p_1 and p_2 is actually a Gaussian pdf:

$$\frac{1}{\sqrt{2\pi(\sigma_1^2+\sigma_2^2)}} e^{-\frac{(\mu_1-\mu_2)^2}{2(\sigma_1^2+\sigma_2^2)}} = 1 \Leftrightarrow -\frac{(\mu_1-\mu_2)^2}{2(\sigma_1^2+\sigma_2^2)} = \ln(\sqrt{2\pi(\sigma_1^2 + \sigma_2^2)}) \Leftrightarrow$$

$$-\frac{(\mu_1-\mu_2)^2}{2(\sigma_1^2+\sigma_2^2)} = \frac{1}{2}\ln(2\pi(\sigma_1^2 + \sigma_2^2)) \Leftrightarrow$$

$$-(\mu_1 - \mu_2)^2 = (\sigma_1^2 + \sigma_2^2)\ln(2\pi(\sigma_1^2 + \sigma_2^2))$$

[0] http://blog.jafma.net/2010/11/09/the-product-of-two-Gaussian-pdfs-is-not-a-pdf-but-is-Gaussian-a-k-a-loving-algebra/

Since ${\sigma_1}^2 + {\sigma_2}^2$ is greater than zero, and the expectations are independent from the variances, we will always find cases (infinite cases, actually) in which the equality does not hold. Thus, we will find infinite cases where the product of two Gaussian pdfs is a Gaussian function but not a Gaussian pdf.
Nevertheless: the scaled Gaussian function obtained as the result of the product has the following pdf-like parameters:

$$
\begin{aligned}
A &= \frac{\mu_1\sigma_2^2+\mu_2\sigma_1^2}{\sigma_1^2+\sigma_2^2} \\
{\sigma_{1x2}}^2 &= \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}
\end{aligned}
$$

[0] http://blog.jafma.net/2010/11/09/the-product-of-two-Gaussian-pdfs-is-not-a-pdf-but-is-Gaussian-a-k-a-loving-algebra/

Kalman Filter/Observer/Estimator

Introduction cont.

- The Kalman filter is over 50 years old but is still one of the most important and common data fusion algorithms in use today
- The great success of the Kalman filter is due to
    - its small computational requirement
    - elegant recursive properties, and
    - its status as the optimal estimator for one-dimensional linear systems with Gaussian error statistics

[0] Faragher, Ramsey, *Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation*, IEEE Signal Processing Magazine, Sep 2012