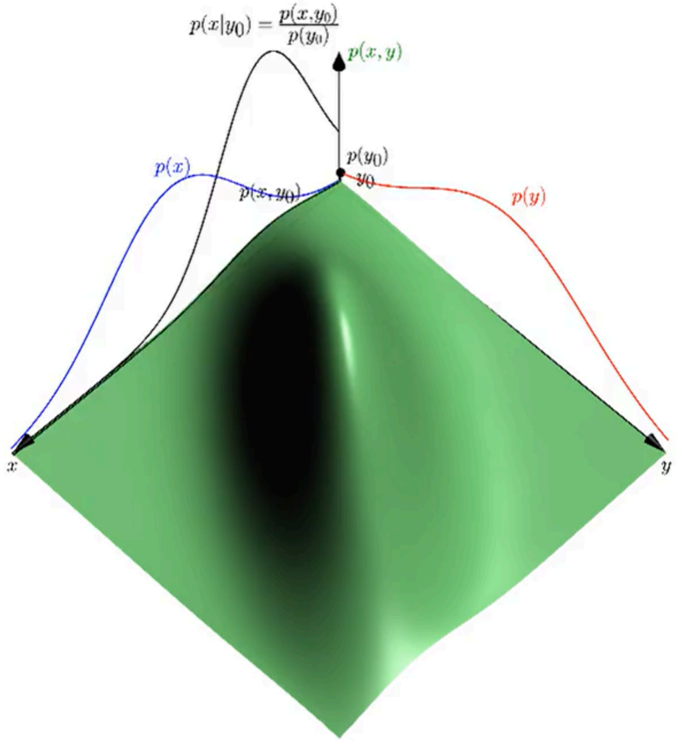
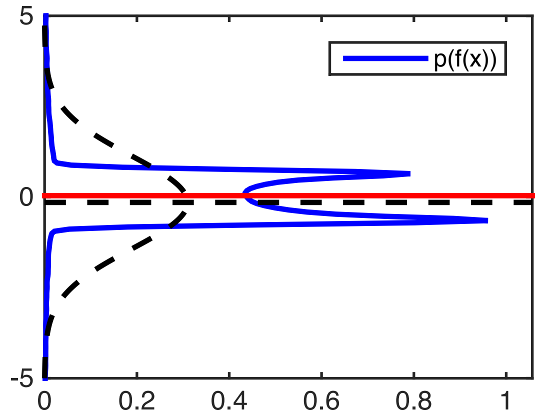
MCHA6100
Advanced Estimation

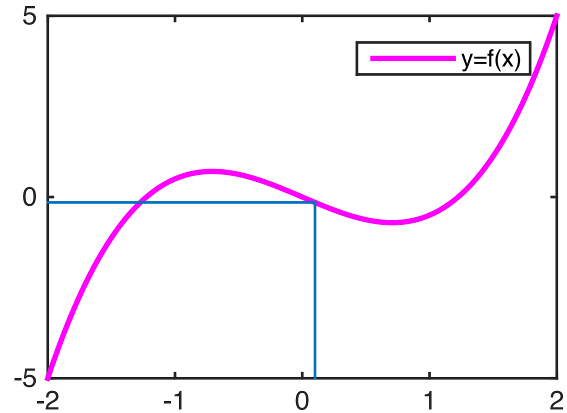


A/Prof Adrian Wills
School of Engineering

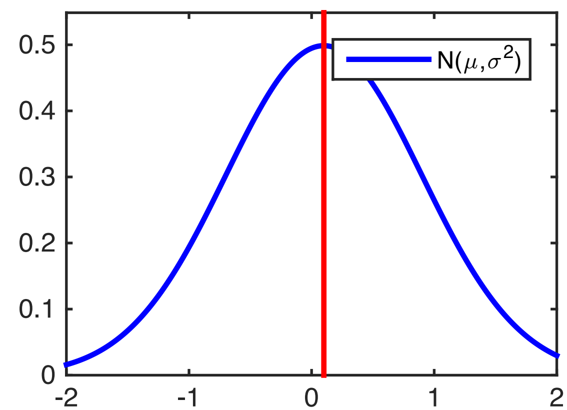
THE UNIVERSITY OF NEWCASTLE AUSTRALIA
FACULTY OF ENGINEERING AND BUILT ENVIRONMENT
www.newcastle.edu.au

Nonlinear mapping

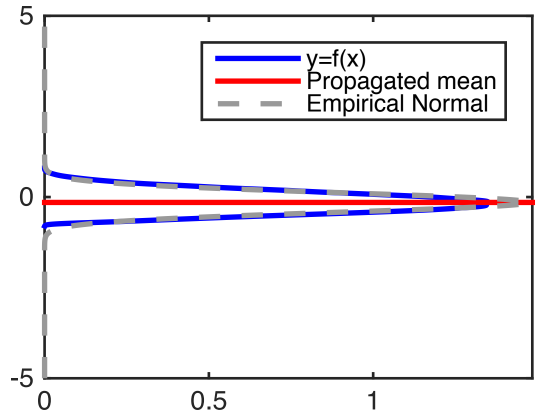


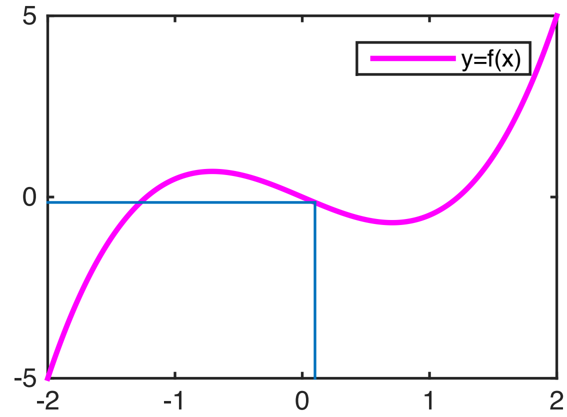


$$y = x^3 - \frac{3}{2}x$$

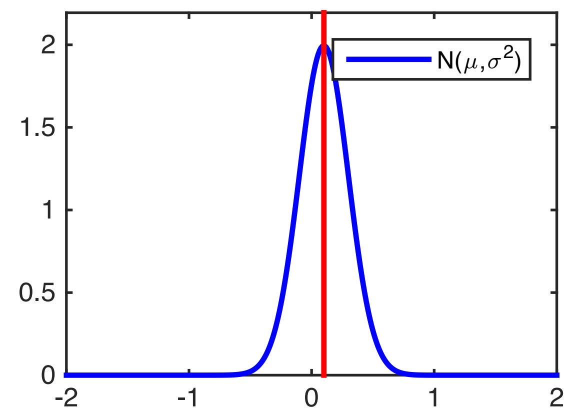


Nonlinear mapping





$$y = x^3 - \frac{3}{2}x$$



Extended Kalman Filter

Taylor Expansion: A local model of a differentiable function $h : S \to \mathbb{R}^m$ on the open set $S \subseteq \mathbb{R}^n$ is provided by the Taylor series expansion (first order Taylor series expansion)

$$h(x) = h(\bar{x}) + J_h(\bar{x})(x - \bar{x}) + e(x - \bar{x}),$$

where $e(\cdot)$ describes all the higher order terms and $J_h(\bar{x})$ is the Jacobian matrix of h evaluated at $\bar{x}$ and is defined as

$$J_h(\bar{x}) \triangleq \begin{bmatrix} \frac{\partial h_1(\bar{x})}{\partial x_1} & \cdots & \frac{\partial h_1(\bar{x})}{\partial x_n} \\ \frac{\partial h_2(\bar{x})}{\partial x_1} & \cdots & \frac{\partial h_2(\bar{x})}{\partial x_n} \\ \vdots & \vdots & \vdots \\ \frac{\partial h_m(\bar{x})}{\partial x_1} & \cdots & \frac{\partial h_m(\bar{x})}{\partial x_n} \end{bmatrix}$$

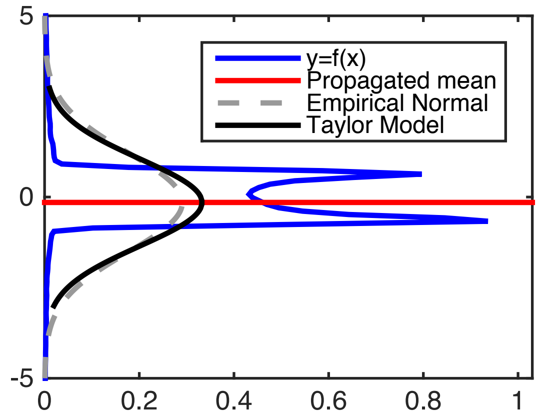
Extended Kalman Filter Assumption: Using this expansion we can rewrite the state-space model as

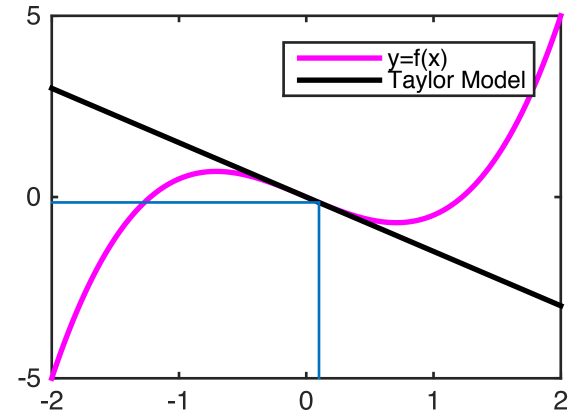
$$\begin{aligned} X_{t+1} &\approx f_t(\mu) + J_{f_t}(\mu)(X_t - \mu) + W_t \\ Y_t &\approx g_t(\mu) + J_{g_t}(\mu)(X_t - \mu) + V_t \end{aligned}$$

and in the above we have assumed the multivariate normal distribution on noise terms and initial state as

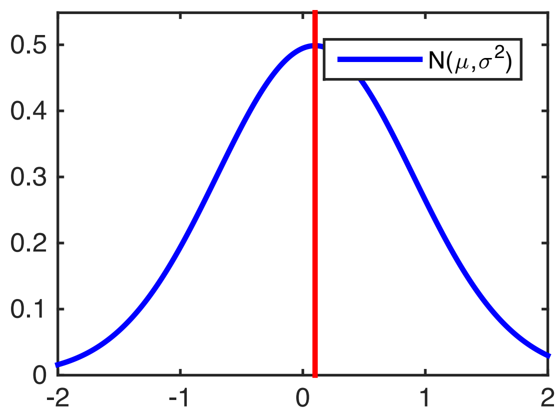
$$\begin{aligned} X_1 &\sim \mathcal{N}(\mu_{1|0},\, P_{1|0}) \\ W_t &\sim \mathcal{N}(0,\, Q_t) \\ V_t &\sim \mathcal{N}(0,\, R_t) \end{aligned}$$

Extended Kalman Filter

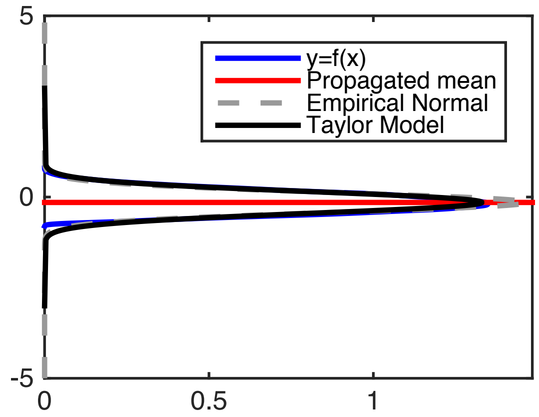


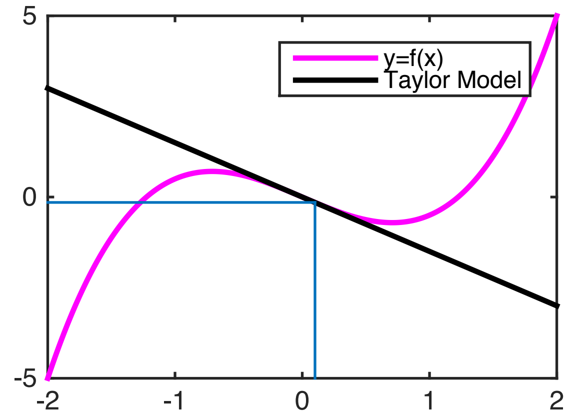


$$y = x^3 - \frac{3}{2}x$$

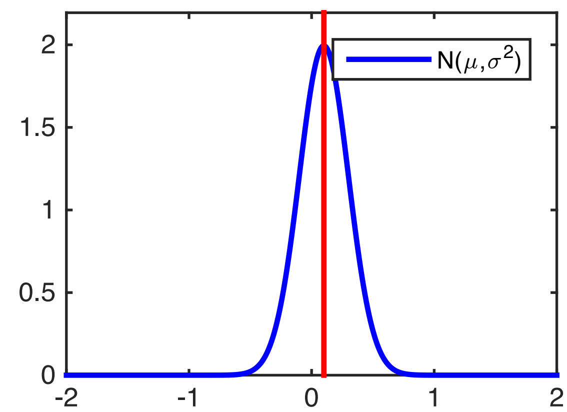


Extended Kalman Filter





$$y = x^3 - \frac{3}{2}x$$



Extended Kalman Filter – Summary

Theorem 1 (Extended Kalman Filter) *Assume that the state-space model can be described as*

$$
\begin{aligned}
X_{t+1} &= f_t(X_t) + W_t, & W_t &\sim \mathcal{N}(0,\, Q_t), & X_1 &\sim \mathcal{N}(\mu_{1|0},\, P_{1|0}) \\
Y_t &= g_t(X_t) + V_t & V_t &\sim \mathcal{N}(0,\, R_t)
\end{aligned}
$$

Further assume that the state-space model can be linearised about a given point μ *as follows*

$$
\begin{aligned}
X_{t+1} &\approx f_t(\mu) + J_{f_t}(\mu)(X_t - \mu) + W_t \\
Y_t &\approx g_t(\mu) + J_{g_t}(\mu)(X_t - \mu) + V_t
\end{aligned}
$$

Then, according to these assumptions, the Bayes' filter has the following solution to the measurement update

$$
\begin{aligned}
p(x_t \mid y_{1:t}) &= \mathcal{N}(\mu_{t|t},\, P_{t|t}) \\
\mu_{t|t} &= \mu_{t|t-1} + P_{t|t-1}C_t^T \left(C_t P_{t|t-1} C_t^T + R_t\right)^{-1} (y_t - g_t(\mu_{t|t-1})) \\
P_{t|t} &= P_{t|t-1} - P_{t|t-1}C_t^T \left(C_t P_{t|t-1} C_t^T + R_t\right)^{-1} C_t P_{t|t-1} \\
C_t &= J_{g_t}(\mu_{t|t-1})
\end{aligned}
$$

and the time update

$$
\begin{aligned}
p(x_{t+1} \mid y_{1:t}) &= \mathcal{N}(\mu_{t+1|t},\, P_{t+1|t}) \\
\mu_{t+1|t} &= f_t(\mu_{t|t}) \\
P_{t+1|t} &= A_t P_{t|t} A_t^T + Q_t \\
A_t &= J_{f_t}(\mu_{t|t})
\end{aligned}
$$

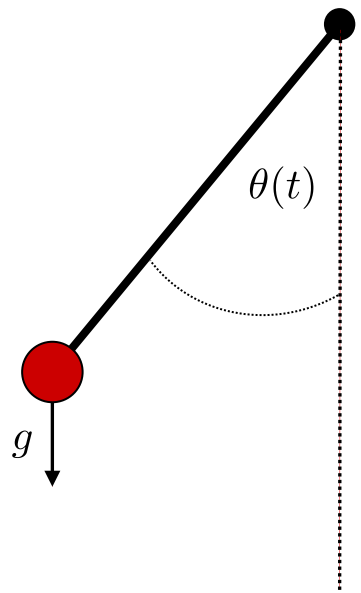
EKF Example

Unscented Kalman Filter

UKF Example

Extended Kalman Filter Example

Consider the following pendulum system with unit length and mass



$$\ddot{\theta}(t) = -g\sin(\theta(t))$$

Suppose we measure the x-position

$$y(t) = \sin(\theta(t)) + e(t)$$

A state-space model is

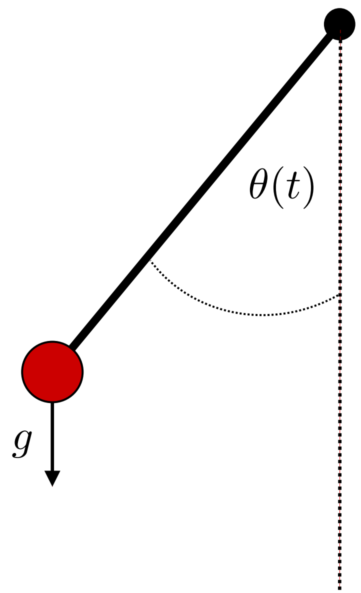
$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} x_2(t) \\ -g\sin(x_1(t)) \end{bmatrix}$$

$$y(t) = \sin(x_1(t)) + e(t)$$

We assume that there is an unknown torque acting on the mass.

How do we estimate the pendulum angle based on y(t) sampled at regular intervals?

Extended Kalman Filter Example



A discrete-time model can be estimated as

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} x_1(k) + \Delta x_2(k) \\ x_2(k) + \Delta(-g\sin(x_1(t))) \end{bmatrix} + \begin{bmatrix} q_1(k) \\ q_2(k) \end{bmatrix}$$

$$\begin{bmatrix} q_1(k) \\ q_2(k) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{\Delta^3}{3} & \frac{\Delta^2}{2} \\ \frac{\Delta^2}{2} & \Delta \end{bmatrix} \right), \qquad \Delta = 0.01$$

$$y(k) = \sin(x_1(k)) + e(k)$$

$$\begin{bmatrix} x_1(1) \\ x_2(1) \end{bmatrix} = \begin{bmatrix} 1 + \frac{\pi}{2} \\ 0 \end{bmatrix}$$

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Extended Kalman Filtering Example

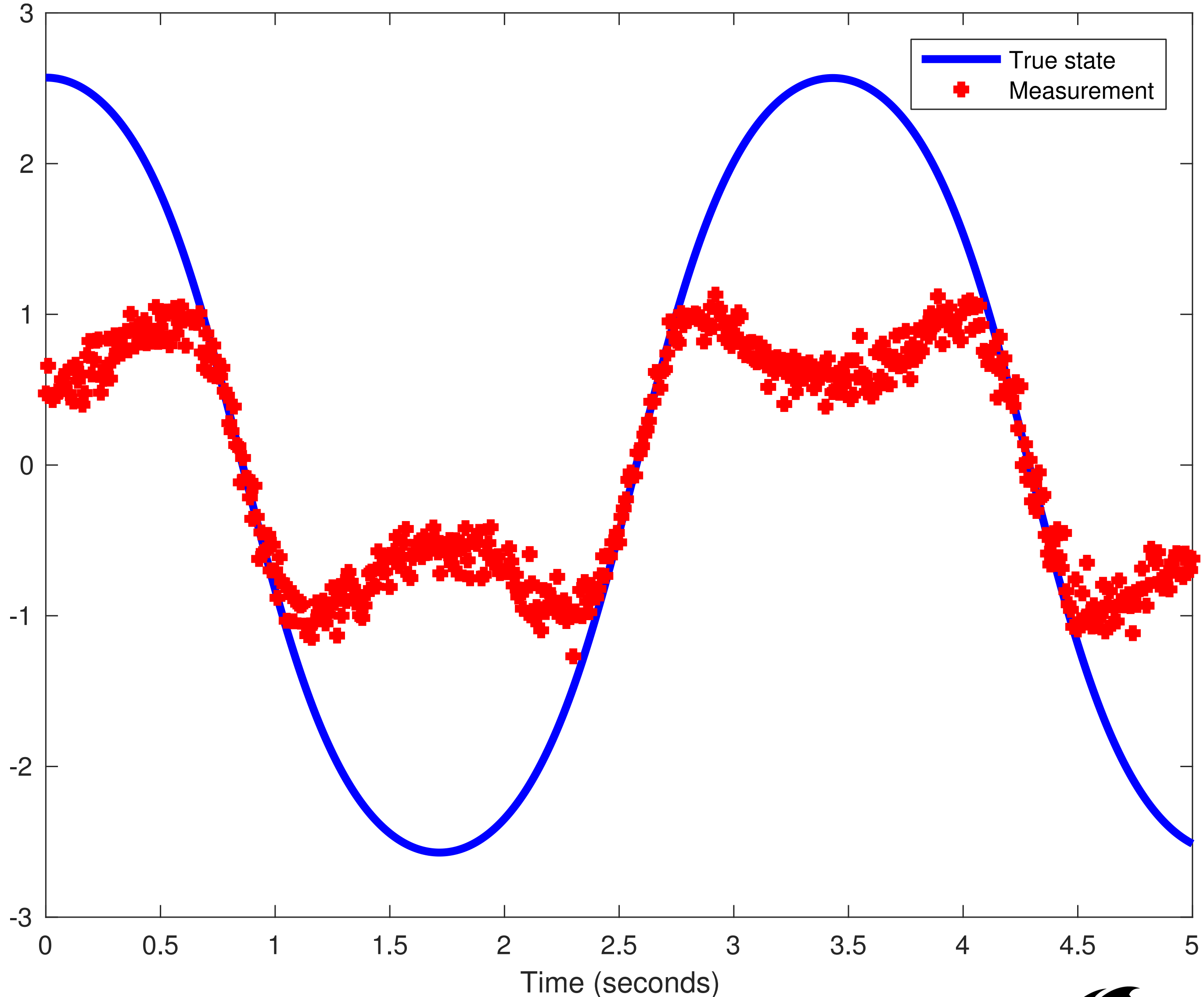

```
%Simulate a pendulum swinging with random force input
%  d^2 a(t) / dt^2 = -g sin(a(t)) + w(t)
%
% In discrete-time state-space form we have
%
% d_/dt [x_1(t)] = [x_2(t)         ]     +    [ 0   ]
%       [x_2(t)]   [-g*sin(x_1(t))]          [w(t)]
%
%           y(t) = sin(x_1(t))           +     e(t)
%
% A discrete-time model is given by (with sample interval D = constant)
%
% [x_1(k+1)] = [x_1(k) + D*x_2(k)            ]  + [q_1(k)]
% [x_2(k+1)]   [x_2(k) + D*(-g*sin(x_1(t)))]    [q_2(k)]
%
%       y(k) = sin(x_1(k))                      +  e(k)
%
% where e(k) ~ N(0,1) and
%
%  [q_1(k)] ~ N(0,Q),    where Q = [(D^3)/3  (D^2)/2]
%  [q_1(k)]                        [(D^2)/2     D   ]
%
%  Let x_1(1) = pi/2

%Generate the data
D      = 0.01;
x0     = [1+pi/2; 0];
g      = 9.81;
fcn    = @(tt,xx) [xx(2,:);-g*sin(xx(1,:))];
[tx,x] = ode45(fcn,[0:D:5],x0);
x = x.';

%Obtain the measurements
y = sin(x(1,:)) + sqrt(0.01)*randn(1,N);
```

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = \begin{bmatrix} x_2(t) \\ -g\sin(x_1(t))\end{bmatrix}$$

$$y(k) = \sin(x_1(k)) + e(k)$$

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Extended Kalman Filtering Example



THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Extended Kalman Filtering Example


```
%Now we run the EKF
n = 2;
N = length(tx);
R =  0.1;
Q =  [(D^3)/3  (D^2)/2;
      (D^2)/2     D];

%Now run the EKF
%Make some room for the filter
Pp  = zeros(n,n,N+1);
Pf  = zeros(n,n,N);
mup = zeros(n,N+1);
muf = zeros(n,N);

%setup initial state PDF
mup(:,1)  = [0;0];
Pp(:,:,1) = 0.001*eye(n);
```

$$p(x_1) = \mathcal{N}(\mu_{1|0}, P_{1|0})$$

$$C_t = J_{g_t}(\mu_{t|t-1})$$

$$\mu_{t|t} = \mu_{t|t-1} + P_{t|t-1}C_t^T\left(C_tP_{t|t-1}C_t^T + R_t\right)^{-1}(y_t - g_t(\mu_{t|t-1}))$$

$$P_{t|t} = P_{t|t-1} - P_{t|t-1}C_t^T\left(C_tP_{t|t-1}C_t^T + R_t\right)^{-1}C_tP_{t|t-1}$$

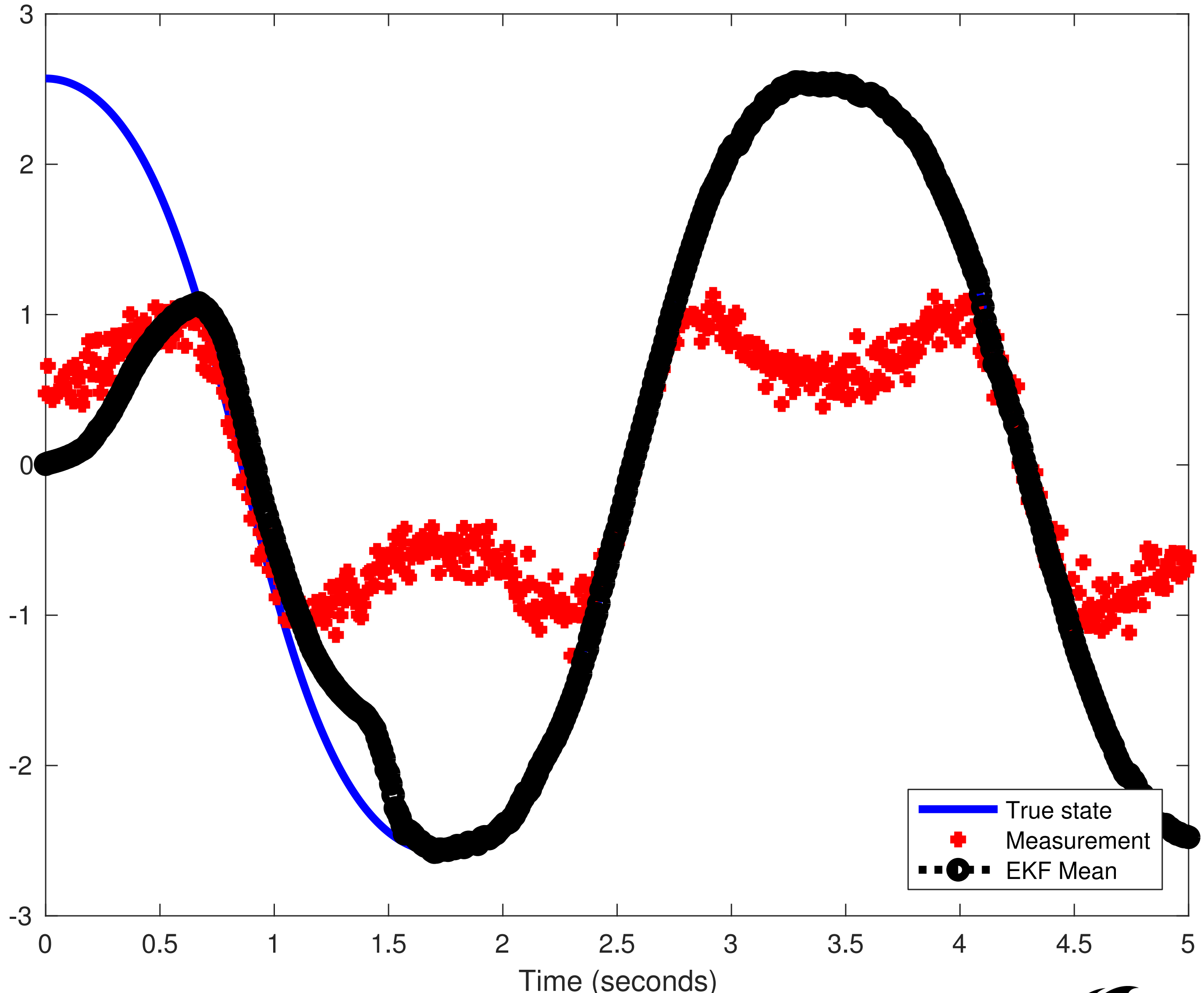
```
%Run the Extended Kalman Filter
for t=1:N,
    %Update with new measurement
    C           = [cos(mup(1,t)) 0];
    muf(:,t)    = mup(:,t)  + Pp(:,:,t)*C'*((C*Pp(:,:,t)*C' + R)\(y(t) - sin(mup(1,t))));
    Pf(:,:,t)   = Pp(:,:,t) - Pp(:,:,t)*C'*((C*Pp(:,:,t)*C' + R)\(C*Pp(:,:,t)));

    %Predict forward
    A           = [1  D ; D*(-g*cos(muf(1,t)))  1 ];
    mup(:,t+1)  = [muf(1,t) + D*muf(2,t); muf(2,t) + D*(-g*sin(muf(1,t)))];
    Pp(:,:,t+1) = A*Pf(:,:,t)*A'  + Q;
end
```

$$A_t = J_{f_t}(\mu_{t|t})$$

$$\mu_{t+1|t} = f_t(\mu_{t|t})$$

$$P_{t+1|t} = A_tP_{t|t}A_t^T + Q_t$$

Extended Kalman Filtering Example



Random Numbers vs Random Variables

We have been using random variables X, Y, which are ***functions*** from the sample space to the space of reals.

Random numbers are ***realisations*** of the random variable for some fixed element of the sample space,

$$X : \Omega \to \mathbb{R}^n, \qquad x = X(\omega), \quad \omega \in \Omega$$

If we have a sequence of i.i.d. random variables

$$X_i \sim p_X(x), \qquad i = 1, \dots, N$$

Then we can generate random numbers for this sequence as

$$\Big[x_1, \dots, x_N\Big] = \Big[X_1(\omega), \dots, X_N(\omega)\Big], \qquad \omega \in \Omega$$

Random Numbers from the MVN

How do we ***draw realisations*** from the MVN distribution?

Recall that the MVN is defined as:

Definition (Multivariate Normal (MVN)): Let $X : \Omega \to \mathbb{R}^n$ be a vector of continuous random variables (i.e. $X_i : \Omega \to \mathbb{R}$ as usual). Then X has a Multivariate Normal Distribution if the joint PDF is

$$\begin{aligned} p_X(x) &= \frac{1}{\sqrt{|2\pi\Sigma|}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \\ &= \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \end{aligned}$$

where $x \in \mathbb{R}^n$, $\mu \in \mathbb{R}^n$ is the mean vector and $\Sigma \in \mathbb{R}^{n\times n}$ is a positive definite matrix and $|A| = \det A$. We often write this as $X \sim \mathcal{N}(\mu, \Sigma)$.

Note that Σ is called the co-variance matrix and is defined in a similar manner to variance, but now we must keep track of all the cross terms. Specifically, the co-variance is defined as

$$\begin{aligned} \Sigma &= E\left\{(X - E\{X\})(X - E\{X\})^T\right\} \\ &= E\left\{(X - \mu)(X - \mu)^T\right\} \end{aligned}$$

Random Numbers from the MVN

How do we ***draw realisations*** from the MVN distribution?

We can generate samples from a MVN distribution $\mathcal{N}(\mu, P)$ by noting that

$$\begin{aligned} z_i &= Z_i(\omega), \\ Z_i &= \mu + P^{1/2} V_i, \qquad V_i \sim \mathcal{N}(0,\, I) \end{aligned}$$

where the matrix $P^{1/2}$ is such that

$$P^{1/2}(P^{1/2})^T = P$$

This works since

$$\begin{aligned} E\{Z_i\} = E\{\mu + P^{1/2} V_i\} = E\{\mu\} + E\{P^{1/2} V_i\} = \mu + P^{1/2} E\{V_i\} \\ = \mu \end{aligned}$$

And the co-variance matrix is given by

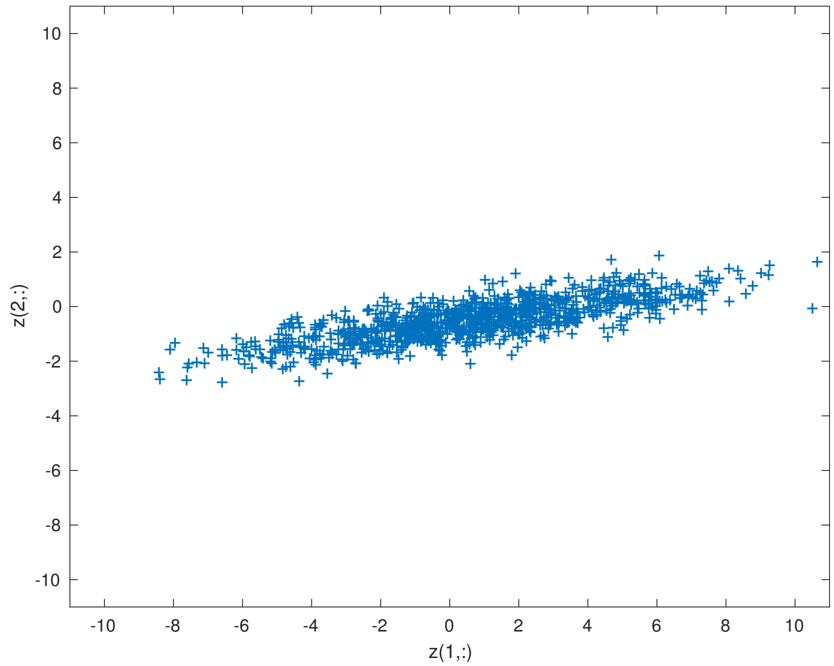
$$\begin{aligned} E\{(Z_i - E\{Z_i\})(Z_i - E\{Z_i\})^T\} &= E\{(\mu + P^{1/2} V_i - \mu)(\mu + P^{1/2} V_i - \mu)^T\} \\ &= E\{(P^{1/2} V_i)(P^{1/2} V_i)^T\} \\ &= E\{P^{1/2} V_i V_i^T P^{T/2}\} \\ &= P^{1/2} \underbrace{E\{V_i V_i^T\}}_{=I} (P^{1/2})^T \\ &= P \end{aligned}$$

Random Numbers from the MVN

How do we ***draw realisations*** from the MVN distribution?

In Matlab, this looks like:

```
%Draw N samples from n-dimensional MVN with mean mu and covariance P
n  = 2;
N  = 1000;
sP = randn(n);
P  = sP*sP';
mu = randn(n,1);

z = mu + sP*randn(n,N);

plot(z(1,:),z(2,:),'+');
xlabel('z(1,:)')
ylabel('z(2,:)')
```



How can we estiamte the mean and co-variance from the samples?

Random Numbers from the MVN

How can we estiamte the mean and co-variance from the samples?

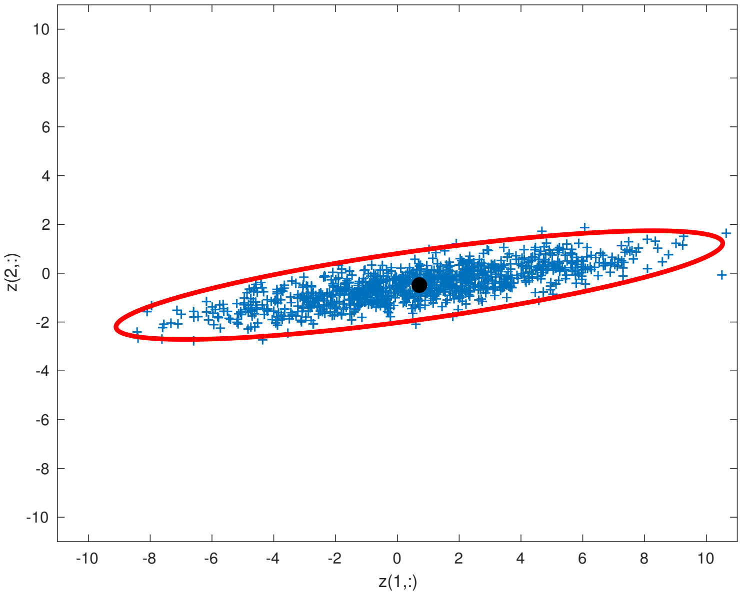
Based on these samples z_i for $i = 1, \ldots, N$ for some positive integer N, we can construct the sample mean $\hat{\mu}$ and the sample co-variance $\hat{P}$ according to

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} z_i,$$

$$\hat{P} = \frac{1}{N}\sum_{i=1}^{N}(z_i - \hat{\mu})(z_i - \hat{\mu})^T$$

or in Matlab as

```
muh = sum(z,2)/N;
Ph  = (z-muh)*(z-muh).'/N;
```



Nonlinear Mapping of Samples

Suppose we want to map the z_i's through a non-linear function $f : \mathbb{R}^n \to \mathbb{R}^m$ and for each i generate a new vector $y_i \in \mathbb{R}^m$ according to

$$y_i \triangleq f(z_i)$$

To make this more concrete, suppose that $z_i \in \mathbb{R}^2$ where the first element $z_i(1)$ corresponds to range (in metres) and the second component $z_i(2)$ corresponds to angle (in degrees) then the map f from polar to cartesian coordinates is given by

$$\begin{aligned} y_i(1) &= z_i(1)\cos(z_i(2)), \\ y_i(2) &= z_i(1)\sin(z_i(2)), \end{aligned}$$

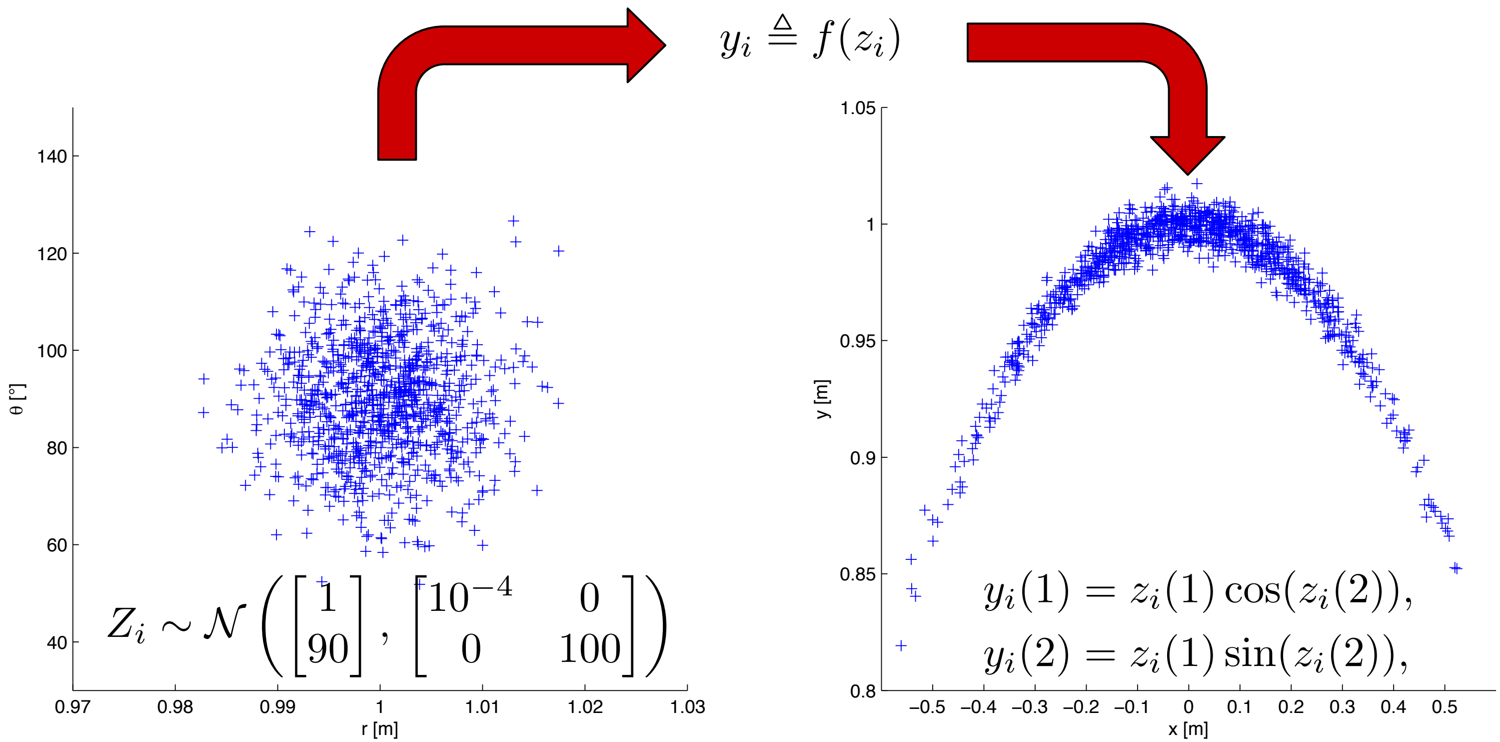
Suppose that the random variable Z_i is distributed according to

$$Z_i \sim \mathcal{N}\left(\begin{bmatrix} 1 \\ 90 \end{bmatrix}, \begin{bmatrix} 10^{-4} & 0 \\ 0 & 100 \end{bmatrix}\right)$$

In Matlab, generate samples from the random variable Y?

Nonlinear Mapping of Samples

Consider a polar to Cartesian transformation.



$$y_i \triangleq f(z_i)$$

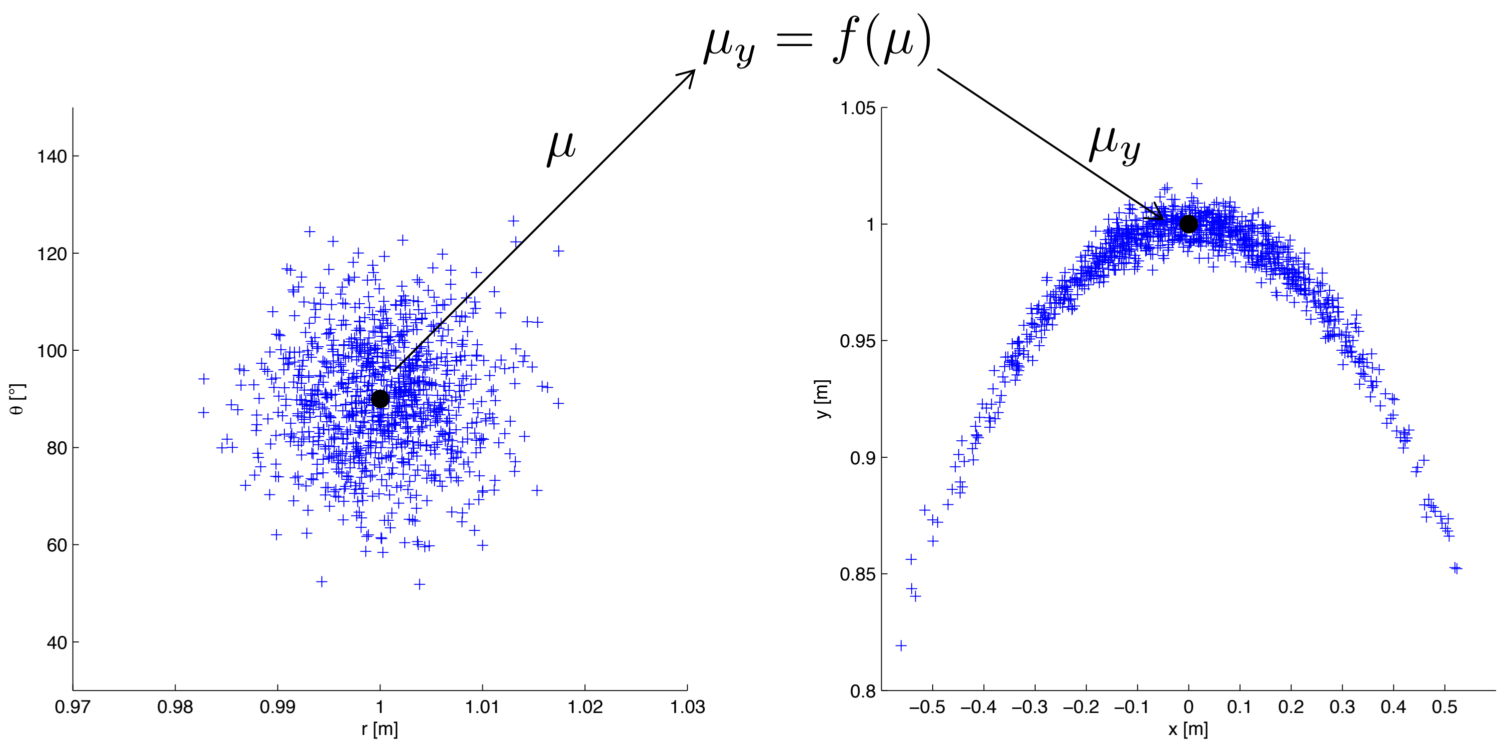
$$Z_i \sim \mathcal{N}\left(\begin{bmatrix}1\\90\end{bmatrix}, \begin{bmatrix}10^{-4} & 0\\0 & 100\end{bmatrix}\right)$$

$$y_i(1) = z_i(1)\cos(z_i(2)),$$
$$y_i(2) = z_i(1)\sin(z_i(2)),$$

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Nonlinear Mapping of Samples

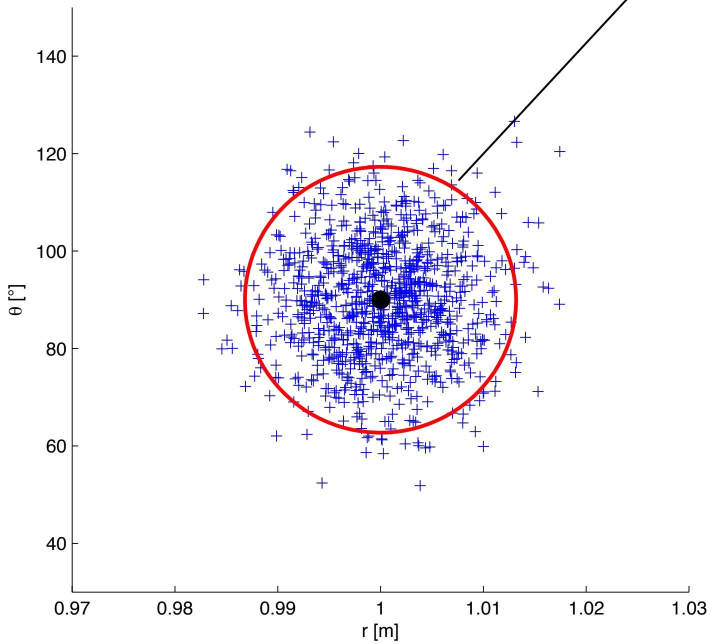
Propagate the mean through the linearised transformation.

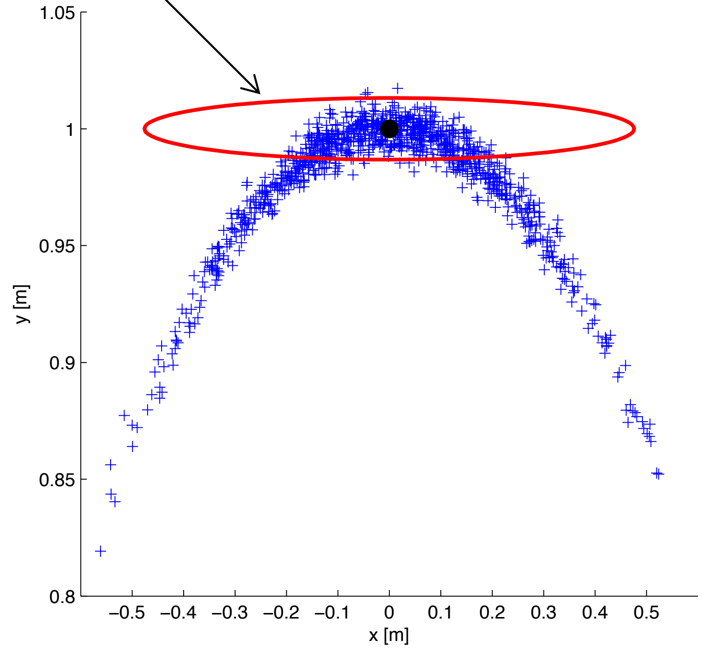


Nonlinear Mapping of Samples

Propagate the covariance through the linearised transformation.

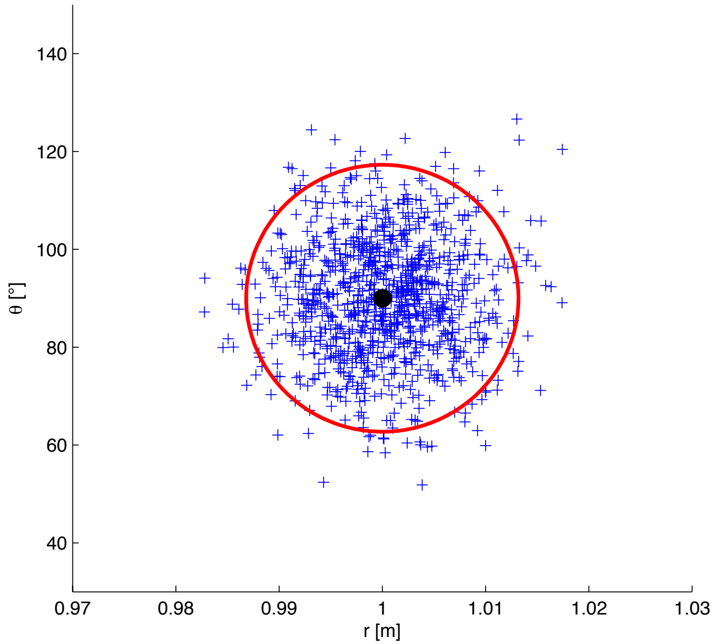
$$\Sigma = CPC^T \qquad C = \left.\frac{\partial f}{\partial z}\right|_{z=\mu}$$

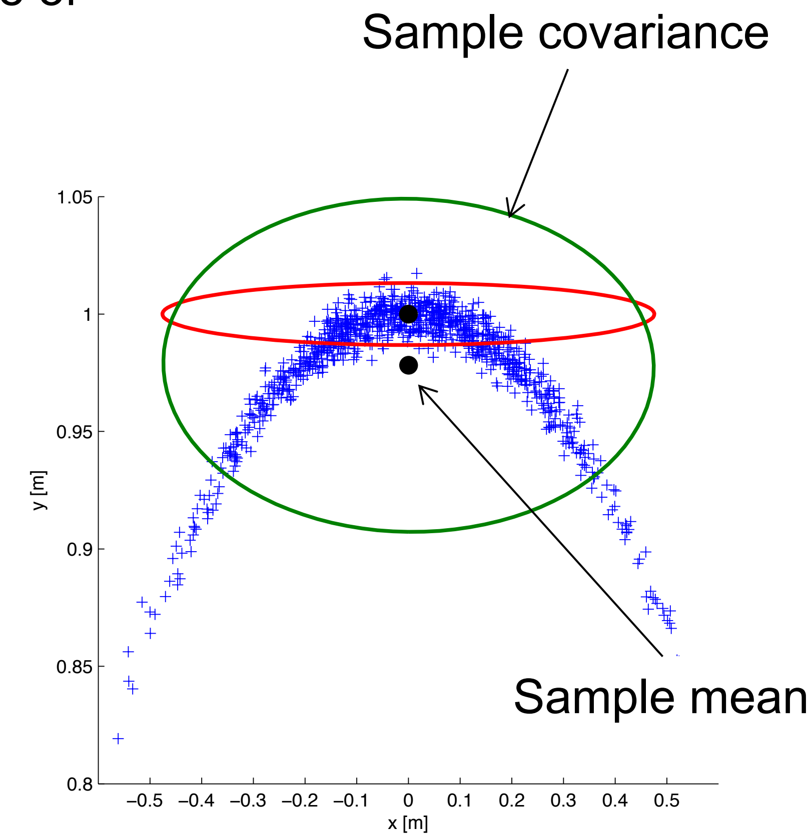




Nonlinear Mapping of Samples

The sample mean and sample covariance of the transformed data disagree.





Nonlinear Mapping of Samples

Clearly, in some cases, using the linearisation to propagate the mean and covariance does not provide a good approximation of the true mean and covariance of the transformed RV.

How can we more accurately calculate the mean and covariance of a transformed RV?

One answer is the Unscented Transform (UT), which leads to the Unscented Kalman Filter (UKF).

Another answer is just to push those thousands of points through the models and just see what happens. This approach leads to the Particle Filter (PF).

Unscented Transform

The basic idea is to pick a set of 2*n* **sigma points**, according to

$$\begin{aligned} x^{(i)} &= \mu + p^{(i)}, \qquad i = 1, \ldots, n \\ x^{(i+n)} &= \mu - p^{(i)}, \qquad i = 1, \ldots, n \end{aligned}$$

where $p^{(i)}$ is the i'th column of the matrix $\sqrt{n}\,P^{1/2}$

$$\left[p^{(1)}, \ldots, p^{(n)}\right] \triangleq \sqrt{n}\,P^{1/2}$$

and $P^{1/2}$ is a lower Cholesky factor of P such that

$$P^{1/2}\left(P^{1/2}\right)^T = P$$

Then, we propagate the sigma points through the function of interest, and compute the sample mean and sample covariance of the transformed points.

Unscented Transform

Definition (Unscented Transform): The Unscented Transform of the pair (μ, P) through the function $y = f(x)$ is denoted as $\mathcal{UT}_f(\mu, P) \to (\mu_y, P_y)$ and defined as

$$\mu_y \triangleq \frac{1}{2n}\sum_{i=1}^{2n} y^{(i)}, \qquad P_y \triangleq \frac{1}{2n}\sum_{i=1}^{2n}(y^{(i)} - \mu_y)(y^{(i)} - \mu_y)^T$$

where

$$y^{(i)} = f(x^{(i)}), \qquad i = 1, \ldots, 2n$$

and

$$\begin{aligned} x^{(i)} &= \mu + p^{(i)}, \qquad i = 1, \ldots, n \\ x^{(i+n)} &= \mu - p^{(i)}, \qquad i = 1, \ldots, n \end{aligned}$$

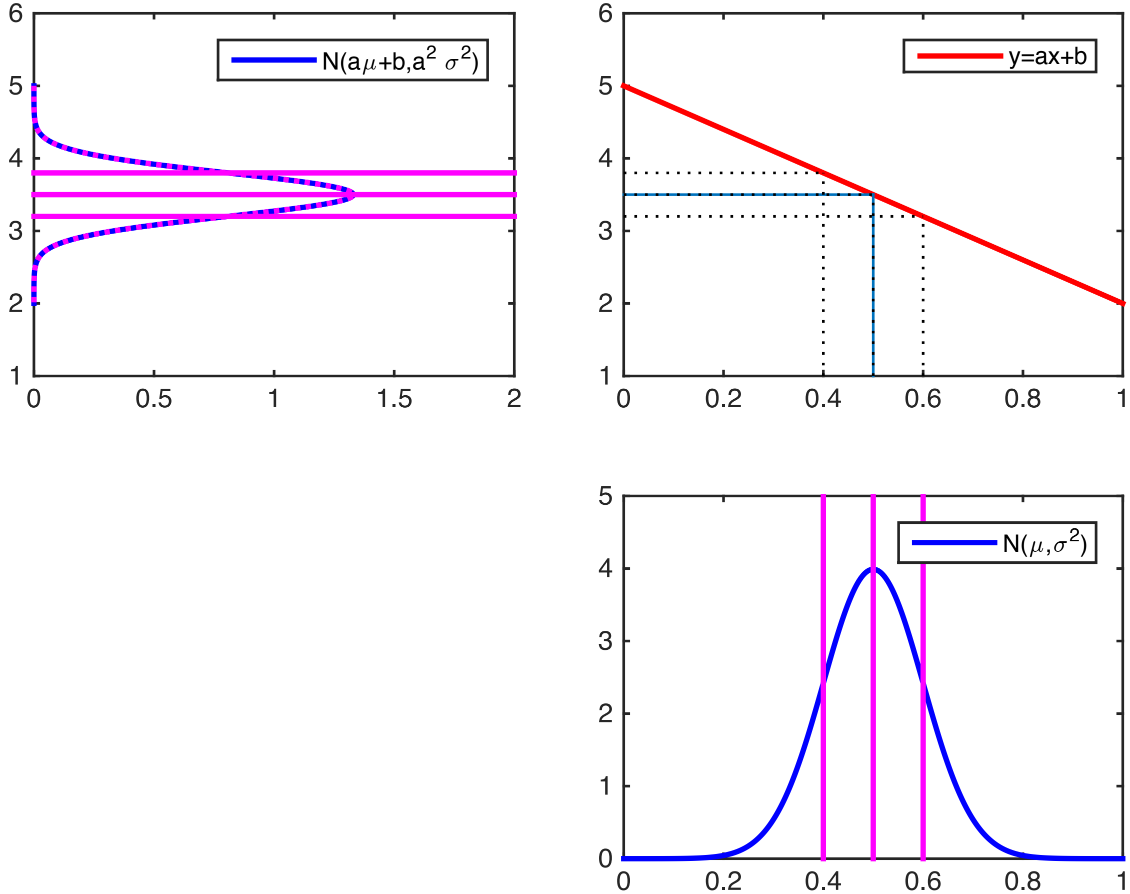
where $p^{(i)}$ is the i'th column of the matrix $\sqrt{n}\,P^{1/2}$

$$\left[p^{(1)}, \ldots, p^{(n)}\right] \triangleq \sqrt{n}\,P^{1/2}$$

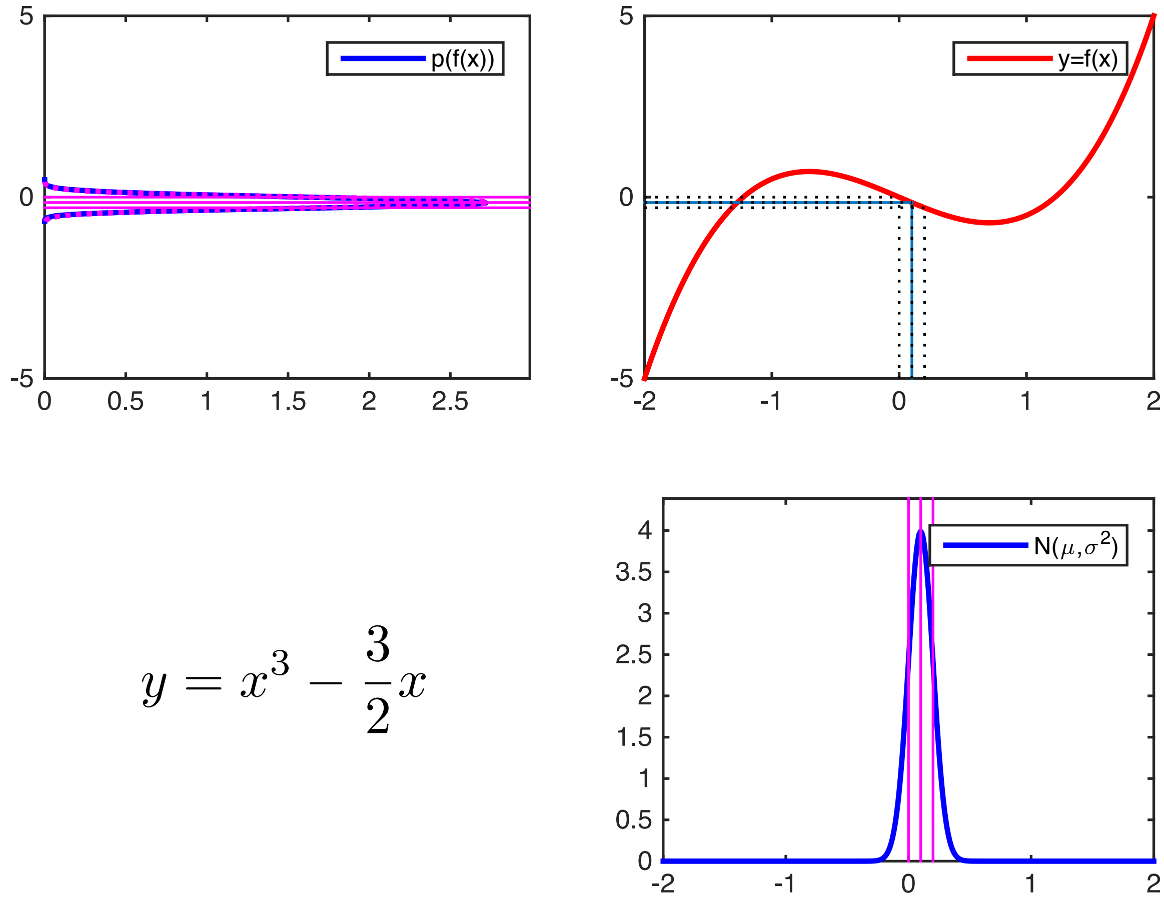
and $P^{1/2}$ is a lower Cholesky factor of P such that

$$P^{1/2}\left(P^{1/2}\right)^T = P$$

Unscented Transform

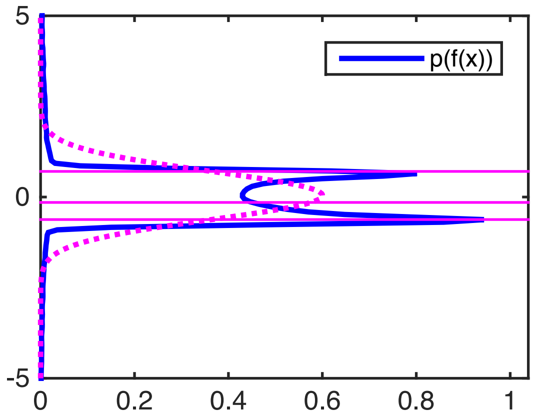


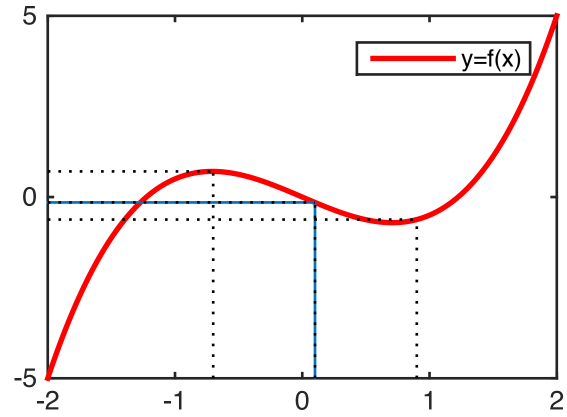
Unscented Transform



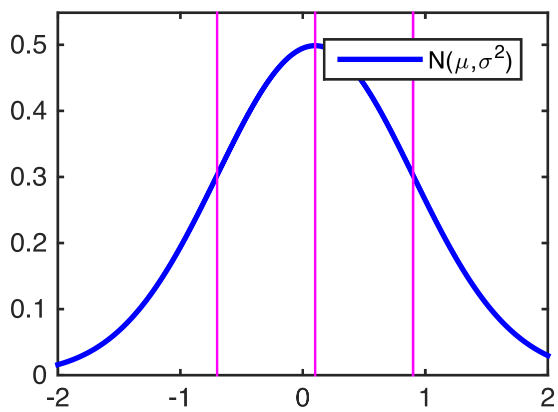
$$y = x^3 - \frac{3}{2}x$$

Unscented Transform





$$y = x^3 - \frac{3}{2}x$$



Unscented Transform


```
%Generate random samples
n  = 2;
N  = 2000;
mu = [1;90];
sP = diag([0.01,10]);
z  = mu + sP*randn(n,N);

y = [z(1,:).*cos(z(2,:)*pi/180);
     z(1,:).*sin(z(2,:)*pi/180)];

muy = sum(y,2)/N
Py  = (y-muy)*(y-muy).'/N

%Generate sigma points
p   = sqrt(n)*sP;
zsp = mu + p;
zsp = [zsp mu-p];

ysp = [zsp(1,:).*cos(zsp(2,:)*pi/180);
       zsp(1,:).*sin(zsp(2,:)*pi/180)];

muysp = sum(ysp,2)/(2*n)
Pysp  = (ysp-muysp)*(ysp-muysp).'/(2*n)
```

$$\left[p^{(1)}, \ldots, p^{(n)}\right] \triangleq \sqrt{n}\, P^{1/2}$$

$$P^{1/2}\left(P^{1/2}\right)^T = P$$

$$x^{(i)} = \mu + p^{(i)}, \qquad i = 1, \ldots, n$$

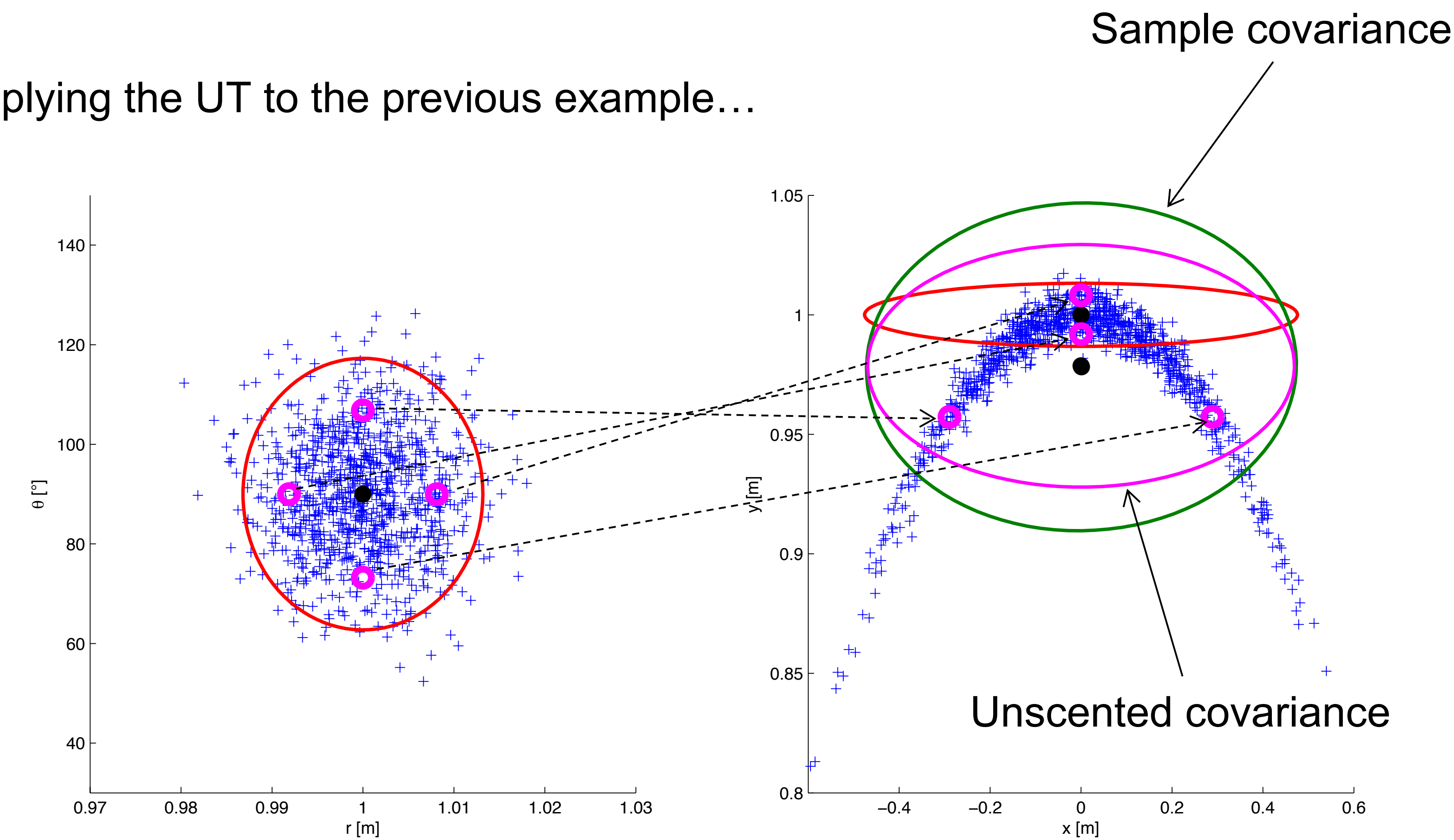
$$x^{(i+n)} = \mu - p^{(i)}, \qquad i = 1, \ldots, n$$

$$y^{(i)} = f(x^{(i)}), \qquad i = 1, \ldots, 2n$$

$$\mu_y \triangleq \frac{1}{2n}\sum_{i=1}^{2n} y^{(i)}, \qquad P_y \triangleq \frac{1}{2n}\sum_{i=1}^{2n}(y^{(i)} - \mu_y)(y^{(i)} - \mu_y)^T$$

Unscented Transform

Applying the UT to the previous example…



Bayesian Filtering

The ***Bayes' Filter*** we have arrived at looks like the following algorithm.

Algorithm 1 Bayesian Filtering Algorithm

Require: An initial state PDF $p(x_1)$ (note that we define $p(x_1 \mid y_{1:0}) = p(x_1)$).

1: Set the time index to $t = 1$.
2: **while true do**
3: **Measurement update: update the prior with new information y_t:**

$$p(x_t \mid y_{1:t}) = \frac{p(y_t \mid x_t)p(x_t \mid y_{1:t-1})}{\int p(y_t \mid x_t)p(x_t \mid y_{1:t-1})dx_t}. \tag{1}$$

4: **Prediction update: compute the state distribution at $t+1$:**

$$p(x_{t+1} \mid y_{1:t}) = \int p(x_{t+1} \mid x_t)p(x_t \mid y_{1:t})dx_t. \tag{2}$$

5: Increment the time counter $t \leftarrow t + 1$.
6: **end while**

Unscented Kalman Filter – Measurement

Assume that the state-space model can be described by

$$
\begin{aligned}
X_{t+1} &= f_t(X_t) + W_t \\
Y_t &= g_t(X_t) + V_t
\end{aligned}
\qquad
\begin{aligned}
X_1 &\sim \mathcal{N}(\mu_{1|0},\, P_{1|0}) \\
W_t &\sim \mathcal{N}(0,\, Q_t) \\
V_t &\sim \mathcal{N}(0,\, R_t)
\end{aligned}
$$

Recall that in the Kalman Filter proof for the measurement update we used

$$
\begin{aligned}
p(y_t \mid x_t)p(x_t \mid y_{1:t-1}) &= p(y_t \mid x_t, y_{1:t-1})p(x_t \mid y_{1:t-1}) \\
&= p(y_t,\, x_t \mid y_{1:t-1}) \\
&= \mathcal{N}\left(\begin{bmatrix} y_t - C_t\mu_{t|t-1} - D_t u_t \\ x_t - \mu_{t|t-1} \end{bmatrix},\, \begin{bmatrix} R_t + C_t P_{t|t-1} C_t^T & C_t P_{t|t-1} \\ P_{t|t-1} C_t^T & P_{t|t-1} \end{bmatrix}\right)
\end{aligned}
$$

From the Theorem on conditional distributions from a joint normal we have that

$$
\begin{aligned}
p(x_t \mid y_t,\, y_{1:t-1}) &= p(x_t \mid y_{1:t}) \\
&= \mathcal{N}(\mu_{t|t},\, P_{t|t})
\end{aligned}
$$

Let’s take the same approach, but here we will form the joint via an unscented transform!

Unscented Kalman Filter – Measurement

How do we create joint PDF's using the Unscented transform?

Assume that $X \sim \mathcal{N}(\mu_x, P_x)$ and that $Y = f(X) + V$ and $V \sim \mathcal{N}(0, R)$, then what is the joint Normal as approximated by the Unscented transform? That is, what is the Unscented transform estimate of

$$\begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} P_x & P_{xy} \\ P_{xy}^T & P_y \end{bmatrix} \right)$$

If we define the function

$$h(x, v) \triangleq \begin{bmatrix} x \\ f(x) + v \end{bmatrix}$$

And note that the joint random variable (X, V) is (under the assumption that X and V are independent) distributed according to

$$\begin{bmatrix} X \\ V \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_x \\ 0 \end{bmatrix}, \begin{bmatrix} P_x & 0 \\ 0 & R \end{bmatrix} \right)$$

Then we can just apply the unscented transform

$$\mathcal{UT}_h\left(\begin{bmatrix} \mu_x \\ 0 \end{bmatrix}, \begin{bmatrix} P_x & 0 \\ 0 & R \end{bmatrix} \right) \to \left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} P_x & P_{xy} \\ P_{xy}^T & P_y \end{bmatrix} \right)$$

Unscented Kalman Filter – Measurement

We can apply this directly to the measurement update

We know that $X_t \mid (Y_{1:t-1} = y_{1:t-1}) \sim \mathcal{N}(\mu_{t|t-1},\, P_{t|t-1})$ and that $Y_t \mid (X_t = x_t) \sim \mathcal{N}(g_t(x_t),\, R)$. So we can use the joint Unscented transform idea to create the joint normal

$$\begin{bmatrix} X_t \mid Y_{1:t-1} = y_{1:t-1} \\ Y_t \mid X_t = x_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_{t|t-1} \\ \eta_{t|t-1} \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & K_t \\ K_t^T & \Sigma_t \end{bmatrix} \right)$$

If we define the function

$$h_{t|t-1}(x_t, v_t) \triangleq \begin{bmatrix} x_t \\ g_t(x_t) + v_t \end{bmatrix}$$

Also

$$\begin{bmatrix} X_t \mid Y_{1:t-1} = y_{1:t-1} \\ V_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_{t|t-1} \\ 0 \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & 0 \\ 0 & R_t \end{bmatrix} \right)$$

Then we can just apply the unscented transform

$$\mathcal{UT}_{h_{t|t-1}}\left(\begin{bmatrix} \mu_{t|t-1} \\ 0 \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & 0 \\ 0 & R_t \end{bmatrix} \right) \to \left(\begin{bmatrix} \mu_{t|t-1} \\ \eta_{t|t-1} \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & K_t \\ K_t^T & \Sigma_t \end{bmatrix} \right)$$

Unscented Kalman Filter – Measurement

Theorem 1 (Conditional) *Suppose that the random variables* $X \in \mathbb{R}^n$ *and* $Y \in \mathbb{R}^p$ *have a joint Normal distribution according to*

$$\begin{pmatrix} X \\ Y \end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix} a \\ b \end{bmatrix}, \begin{bmatrix} A & C \\ C^T & B \end{bmatrix} \right).$$

The conditional distribution $p_{X|Y=y}(x \mid y)$

$$p_{X|Y=y}(x \mid y) = \mathcal{N}(a + CB^{-1}(y - b),\ A - CB^{-1}C^T).$$

Therefore,

$$\begin{aligned} p(x_t \mid y_{1:t}) &= \mathcal{N}(\mu_{t|t},\ P_{t|t}), \\ \mu_{t|t} &= \mu_{t|t-1} + K_t \Sigma_t^{-1}(y_t - \eta_{t|t-1}), \\ P_{t|t} &= P_{t|t-1} - K_t \Sigma_t^{-1} K_t^T \end{aligned}$$

Unscented Kalman Filter – Prediction

Recall that the state-space model can be described by

$$
\begin{aligned}
X_{t+1} &= f_t(X_t) + W_t \\
Y_t &= g_t(X_t) + V_t
\end{aligned}
\qquad
\begin{aligned}
X_1 &\sim \mathcal{N}(\mu_{1|0},\, P_{1|0}) \\
W_t &\sim \mathcal{N}(0,\, Q_t) \\
V_t &\sim \mathcal{N}(0,\, R_t)
\end{aligned}
$$

Further, recall that in the Kalman Filter proof for the time update we used

The joint distribution is given by

$$
\begin{aligned}
p(x_{t+1},\, x_t \mid y_{1:t}) &= \frac{1}{(2\pi)^{\frac{n}{2}}(2\pi)^{\frac{n}{2}}|Q_t|^{\frac{1}{2}}|P_{t|t}|^{\frac{1}{2}}} e^{-\frac{1}{2}\epsilon} = \frac{1}{(2\pi)^{\frac{n}{2}}(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}\epsilon} \\
&= \mathcal{N}\left(\begin{bmatrix} A_t\mu_{t|t} + B_t u_t \\ \mu_{t|t} \end{bmatrix}, \begin{bmatrix} Q_t + A_t P_{t|t} A_t^T & A_t P_{t|t} \\ P_{t|t} A_t^T & P_{t|t} \end{bmatrix} \right)
\end{aligned}
$$

So the marginal is given by the Theorem on marginal normal distributions as

$$
p(x_{t+1} \mid y_{1:t}) = \mathcal{N}(\mu_{t+1|t},\, P_{t+1|t})
$$

Again, let's take the same approach, but here we will form the joint via an unscented transform!

Unscented Kalman Filter – Prediction

We can apply this directly to the measurement update

We know that $X_t \mid (Y_{1:t} = y_{1:t}) \sim \mathcal{N}(\mu_{t|t}, P_{t|t})$ and that $X_{t+1} \mid (X_t = x_t) \sim \mathcal{N}(f_t(x_t), Q_t)$. So we can use the joint Unscented transform idea to create the joint normal

$$\begin{bmatrix} X_t \mid Y_{1:t} = y_{1:t} \\ X_{t+1} \mid X_t = x_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_{t|t} \\ \mu_{t+1|t} \end{bmatrix}, \begin{bmatrix} P_{t|t} & L_t \\ L_t^T & P_{t+1|t} \end{bmatrix} \right)$$

If we define the function

$$h_{t|t}(x_t, w_t) \triangleq \begin{bmatrix} x_t \\ f_t(x_t) + w_t \end{bmatrix}$$

Also

$$\begin{bmatrix} X_t \mid Y_{1:t} = y_{1:t} \\ W_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_{t|t} \\ 0 \end{bmatrix}, \begin{bmatrix} P_{t|t} & 0 \\ 0 & Q_t \end{bmatrix} \right)$$

Then we can just apply the unscented transform

$$\mathcal{UT}_{h_{t|t}}\left(\begin{bmatrix} \mu_{t|t} \\ 0 \end{bmatrix}, \begin{bmatrix} P_{t|t} & 0 \\ 0 & Q_t \end{bmatrix} \right) \rightarrow \left(\begin{bmatrix} \mu_{t|t} \\ \mu_{t+1|t} \end{bmatrix}, \begin{bmatrix} P_{t|t} & L_t \\ L_t^T & P_{t+1|t} \end{bmatrix} \right)$$

Multivariate Normal – Marginal Distribution

Theorem 1 (Marginal of Joint Multivariate Normal) *Suppose that the random variables $X \in \mathbb{R}^n$ and $Y \in \mathbb{R}^p$ have a joint Normal distribution according to*

$$\begin{pmatrix} X \\ Y \end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix} a \\ b \end{bmatrix}, \begin{bmatrix} A & C \\ C^T & B \end{bmatrix} \right).$$

The marginal distribution $p_Y(y)$ is given by

$$p_Y(y) = \mathcal{N}(b,\, B).$$

Therefore, from

$$\mathcal{UT}_{h_{t|t}} \left(\begin{bmatrix} \mu_{t|t} \\ 0 \end{bmatrix}, \begin{bmatrix} P_{t|t} & 0 \\ 0 & Q_t \end{bmatrix} \right) \to \left(\begin{bmatrix} \mu_{t|t} \\ \mu_{t+1|t} \end{bmatrix}, \begin{bmatrix} P_{t|t} & L_t \\ L_t^T & P_{t+1|t} \end{bmatrix} \right)$$

and the marginal Theorem, we can just read off the prediction density as

$$p(x_{t+1} \mid y_{1:t}) = \mathcal{N}(\mu_{t+1|t},\, P_{t+1|t})$$

Unscented Kalman Filter – Summary

Theorem 1 (Unscented Kalman Filter) *Assume that the state-space model can be described as*

$$X_{t+1} = f_t(X_t) + W_t, \qquad W_t \sim \mathcal{N}(0,\, Q_t), \qquad X_1 \sim \mathcal{N}(\mu_{1|0},\, P_{1|0})$$
$$Y_t = g_t(X_t) + V_t \qquad V_t \sim \mathcal{N}(0,\, R_t)$$

Then, according to these assumptions, the Unscented Kalman filter has the following solution to the measurement update

$$p(x_t \mid y_{1:t}) = \mathcal{N}(\mu_{t|t},\, P_{t|t}), \qquad \mu_{t|t} = \mu_{t|t-1} + K_t \Sigma_t^{-1}(y_t - \eta_{t|t-1}), \qquad P_{t|t} = P_{t|t-1} - K_t \Sigma_t^{-1} K_t^T$$

where the required terms come from

$$\mathcal{UT}_{h_{t|t-1}}\left(\begin{bmatrix}\mu_{t|t-1}\\ 0\end{bmatrix}, \begin{bmatrix}P_{t|t-1} & 0\\ 0 & R_t\end{bmatrix}\right) \to \left(\begin{bmatrix}\mu_{t|t-1}\\ \eta_{t|t-1}\end{bmatrix}, \begin{bmatrix}P_{t|t-1} & K_t\\ K_t^T & \Sigma_t\end{bmatrix}\right), \qquad h_{t|t-1}(x_t, v_t) \triangleq \begin{bmatrix}x_t\\ g_t(x_t) + v_t\end{bmatrix}$$

and the time update

$$p(x_{t+1} \mid y_{1:t}) = \mathcal{N}(\mu_{t+1|t},\, P_{t+1|t})$$

where the required terms come from

$$\mathcal{UT}_{h_{t|t}}\left(\begin{bmatrix}\mu_{t|t}\\ 0\end{bmatrix}, \begin{bmatrix}P_{t|t} & 0\\ 0 & Q_t\end{bmatrix}\right) \to \left(\begin{bmatrix}\mu_{t|t}\\ \mu_{t+1|t}\end{bmatrix}, \begin{bmatrix}P_{t|t} & L_t\\ L_t^T & P_{t+1|t}\end{bmatrix}\right), \qquad h_{t|t}(x_t, w_t) \triangleq \begin{bmatrix}x_t\\ f_t(x_t) + w_t\end{bmatrix}$$

Unscented Kalman Filtering Example

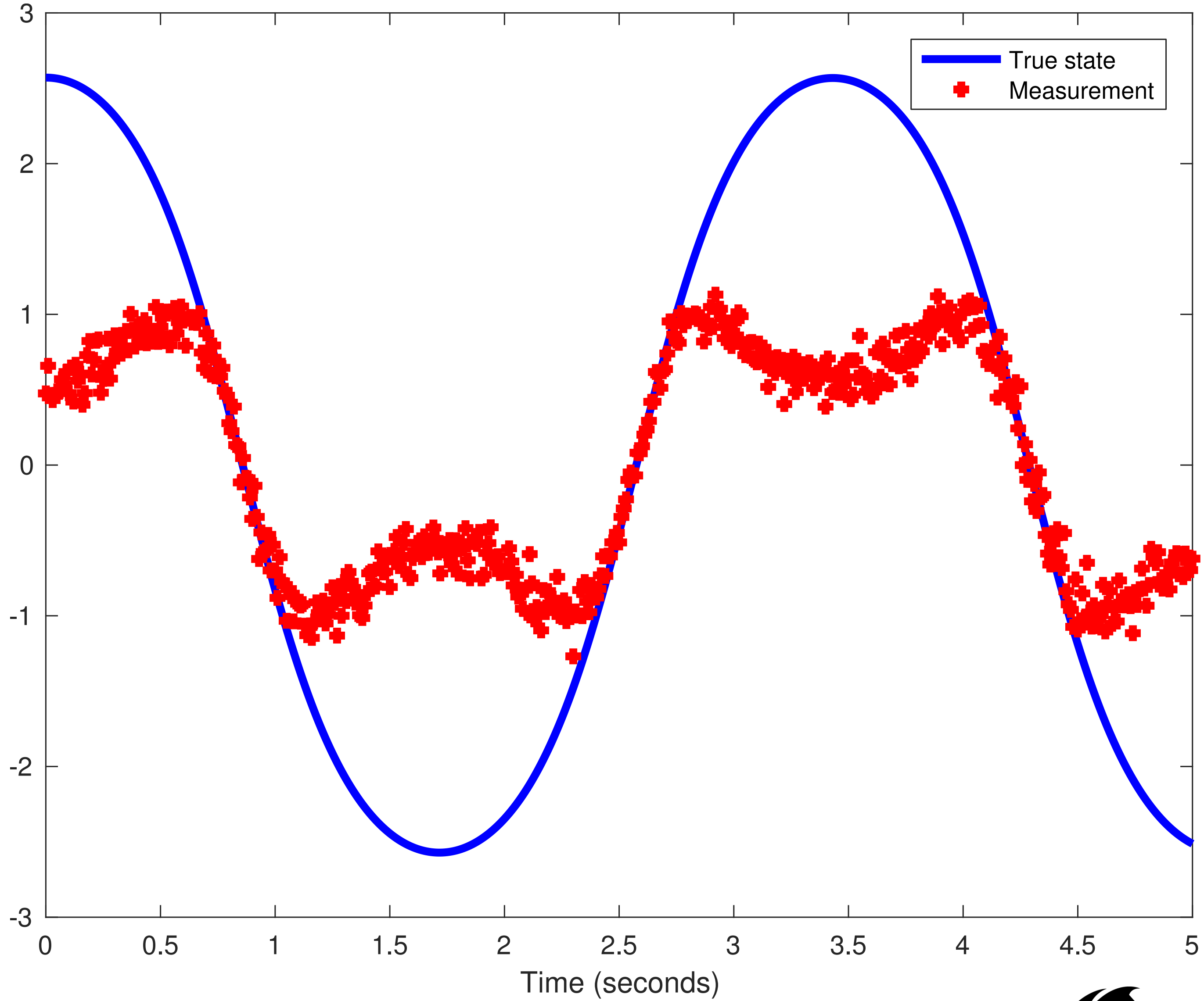

```
%Simulate a pendulum swinging with random force input
%  d^2 a(t) / dt^2 = -g sin(a(t)) + w(t)
%
% In discrete-time state-space form we have
%
% d_/dt [x_1(t)] = [x_2(t)          ]     +    [ 0   ]
%       [x_2(t)]   [-g*sin(x_1(t))]          [w(t)]
%
%           y(t) = sin(x_1(t))          +     e(t)
%
% A discrete-time model is given by (with sample interval D = constant)
%
% [x_1(k+1)] = [x_1(k) + D*x_2(k)            ]  + [q_1(k)]
% [x_2(k+1)]   [x_2(k) + D*(-g*sin(x_1(t)))]    [q_2(k)]
%
%       y(k) = sin(x_1(k))                     +  e(k)
%
% where e(k) ~ N(0,1) and
%
%  [q_1(k)] ~ N(0,Q),   where Q = [(D^3)/3  (D^2)/2]
%  [q_1(k)]                       [(D^2)/2     D   ]
%
%  Let x_1(1) = pi/2

%Generate the data
D      = 0.01;
x0     = [1+pi/2; 0];
g      = 9.81;
fcn    = @(tt,xx) [xx(2,:);-g*sin(xx(1,:))];
[tx,x] = ode45(fcn,[0:D:5],x0);
x = x.';

%Obtain the measurements
y = sin(x(1,:)) + sqrt(0.01)*randn(1,N);
```

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = \begin{bmatrix} x_2(t) \\ -g\sin(x_1(t))\end{bmatrix}$$

$$y(k) = \sin(x_1(k)) + e(k)$$

THE UNIVERSITY OF NEWCASTLE AUSTRALIA

Unscented Kalman Filtering Example



Unscented Kalman Filtering Example


```
%Now we run the UKF
n = 2;
N = length(tx);
R =  0.1;
Q =  [(D^3)/3  (D^2)/2;
       (D^2)/2     D];

%Now run the EKF
%Make some room for the filter
Pp  = zeros(n,n,N+1);
Pf  = zeros(n,n,N);
mup = zeros(n,N+1);
muf = zeros(n,N);
```

$$p(x_1) = \mathcal{N}(\mu_{1|0},\, P_{1|0})$$

```
%setup initial state PDF
mup(:,1)   = [0;0];
Pp(:,:,1) = 0.001*eye(n);
```

$$h_{t|t-1}(x_t, v_t) \triangleq \begin{bmatrix} x_t \\ g_t(x_t) + v_t \end{bmatrix}, \qquad h_{t|t}(x_t, w_t) \triangleq \begin{bmatrix} x_t \\ f_t(x_t) + w_t \end{bmatrix}$$

```
meas = @(x) [x(1:n); sin(x(1))+x(end)];
pred = @(x) [x(1:n); [x(1) + D*x(2); x(2) + D*(-g*sin(x(1)))] + x(n+1:end)];
```

$$\mathcal{UT}_{h_{t|t-1}}\left(\begin{bmatrix} \mu_{t|t-1} \\ 0 \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & 0 \\ 0 & R_t \end{bmatrix}\right) \to \left(\begin{bmatrix} \mu_{t|t-1} \\ \eta_{t|t-1} \end{bmatrix}, \begin{bmatrix} P_{t|t-1} & K_t \\ K_t^T & \Sigma_t \end{bmatrix}\right),$$

$$p(x_t \mid y_{1:t}) = \mathcal{N}(\mu_{t|t},\, P_{t|t}), \qquad \mu_{t|t} = \mu_{t|t-1} + K_t\Sigma_t^{-1}(y_t - \eta_{t|t-1}), \qquad P_{t|t} = P_{t|t-1} - K_t\Sigma_t^{-1}K_t^T$$

```
%Run the Kalman Filter
for t=1:N,
    %Update with new measurement
    [mu,P]        = UnscentedTransform(meas,[mup(:,t);0],blkdiag(Pp(:,:,t),R));
    muf(:,t)      = mu(1:n)  + P(1:n,n+1:end)*(P(n+1:end,n+1:end)\(y(t) - mu(n+1:end)));
    Pf(:,:,t)     = P(1:n,1:n) - P(1:n,n+1:end)*(P(n+1:end,n+1:end)\P(n+1:end,1:n));
```

$$\mathcal{UT}_{h_{t|t}}\left(\begin{bmatrix} \mu_{t|t} \\ 0 \end{bmatrix}, \begin{bmatrix} P_{t|t} & 0 \\ 0 & Q_t \end{bmatrix}\right) \to \left(\begin{bmatrix} \mu_{t|t} \\ \mu_{t+1|t} \end{bmatrix}, \begin{bmatrix} P_{t|t} & L_t \\ L_t^T & P_{t+1|t} \end{bmatrix}\right),$$

$$p(x_{t+1} \mid y_{1:t}) = \mathcal{N}(\mu_{t+1|t},\, P_{t+1|t})$$

```
    %Predict forward
    [mu,P]        = UnscentedTransform(pred,[muf(:,t);zeros(n,1)],blkdiag(Pf(:,:,t),Q));
    mup(:,t+1)    = mu(n+1:end);
    Pp(:,:,t+1)   = P(n+1:end,n+1:end);
end
```

Unscented Kalman Filtering Example


```
function [muy,Py] = UnscentedTransform(fcn,mu,P)

%First generate the sigma points
n  = size(P,1);
sP = chol(P).';
p  = sqrt(n)*sP;
x  = mu + p;
x  = [x mu-p];

%Propagate through the nonlinearity
for i=1:2*n
    y(:,i) = fcn(x(:,i));
end

%Generate the sample mean and covariance
muy = sum(y,2)/(2*n);
Py  = (y - muy)*(y - muy).'/(2*n);
```

$$P^{1/2}\left(P^{1/2}\right)^T = P$$

$$\left[p^{(1)}, \ldots, p^{(n)}\right] \triangleq \sqrt{n}\, P^{1/2}$$

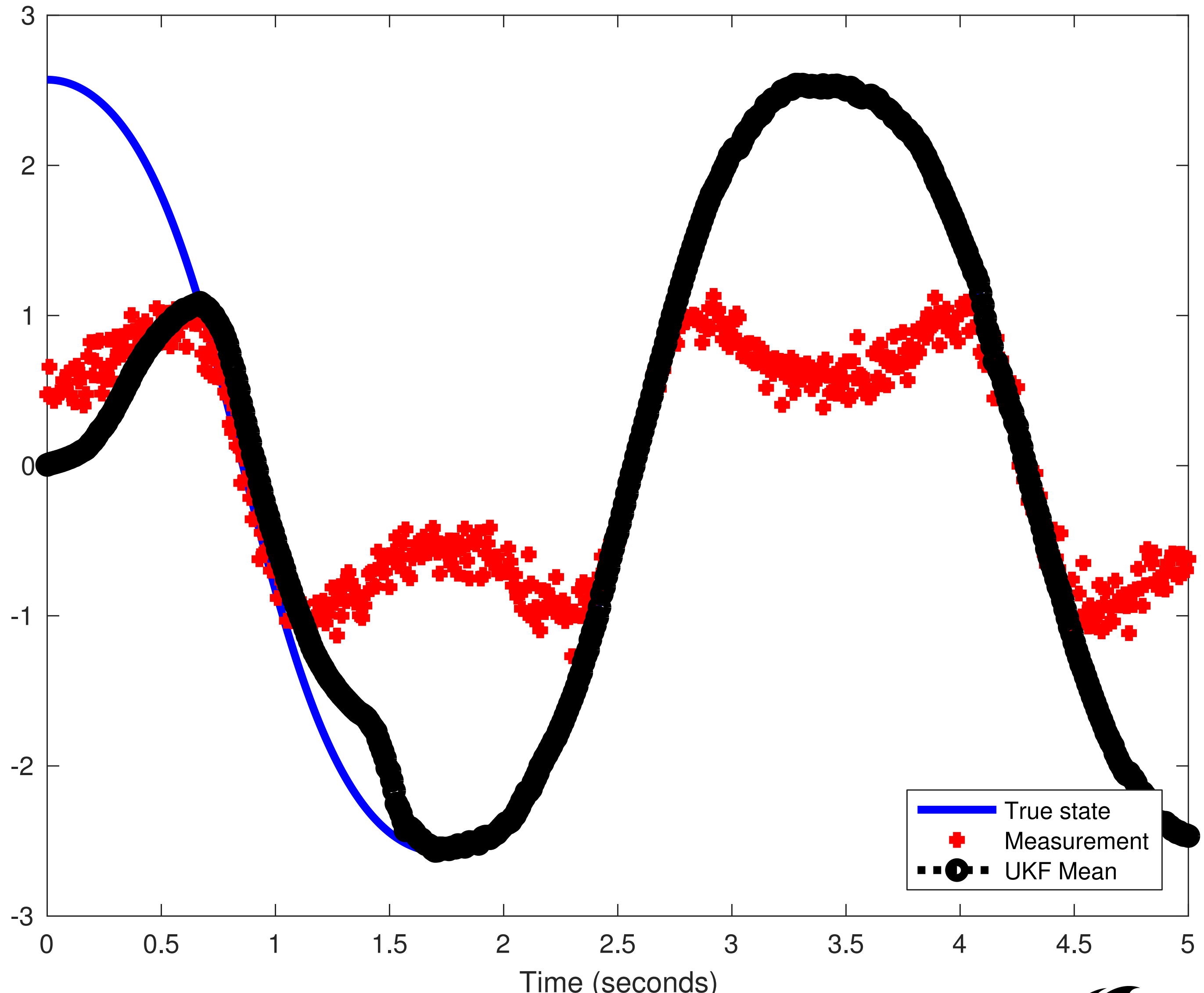
$$x^{(i)} = \mu + p^{(i)}, \qquad i = 1, \ldots, n$$

$$x^{(i+n)} = \mu - p^{(i)}, \qquad i = 1, \ldots, n$$

$$y^{(i)} = f(x^{(i)}), \qquad i = 1, \ldots, 2n$$

$$\mu_y \triangleq \frac{1}{2n}\sum_{i=1}^{2n} y^{(i)}, \qquad P_y \triangleq \frac{1}{2n}\sum_{i=1}^{2n}(y^{(i)} - \mu_y)(y^{(i)} - \mu_y)^T$$

Unscented Kalman Filtering Example



THE UNIVERSITY OF NEWCASTLE AUSTRALIA