extended kalman filters

Prakhar

September 2020

1 Introduction

There were 2 main assumptions in kalman filter.The two assumptions are-:
1. Kalman Filter will always work with Gaussian Distribution.
2. Kalman Filter will always work with Linear Functions.
The prediction and update step both will contain Linear Functions only.

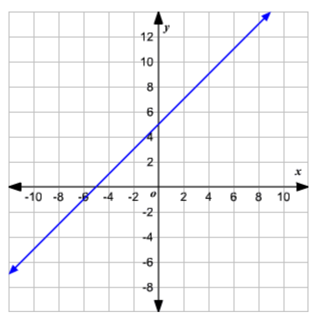

Figure 1: A linear function somewhat looks like this:

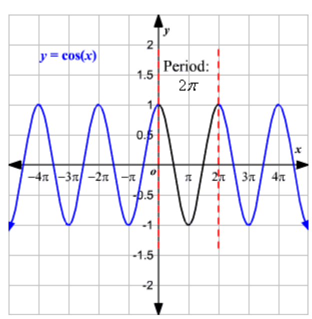

Figure 2: On the other hand a non linear functions looks like this:

Then what is the problem now with KF? Most real world problems involve non linear functions. In most cases, the system is looking into some direction and taking measurement in another direction. This involves angles and sine, cosine functions which are non linear functions which then lead to problems. How does non linear function create problems?if input in a Gaussian is a linear function then the output is also a Gaussian.

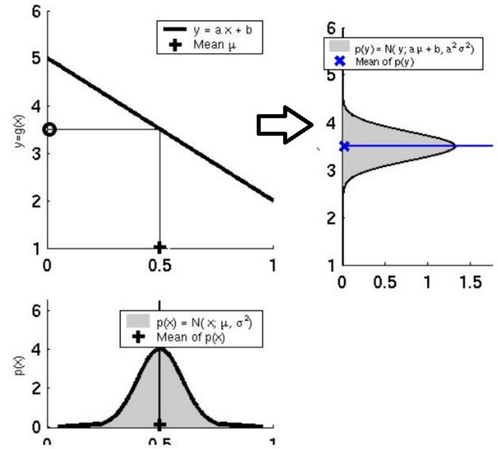

Figure 3: Gaussian + Linear Function = Gaussian

If you feed a Gaussian with a Non linear function then the output is not a Gaussian. Non Linear functions lead to Non Gaussian Distributions.

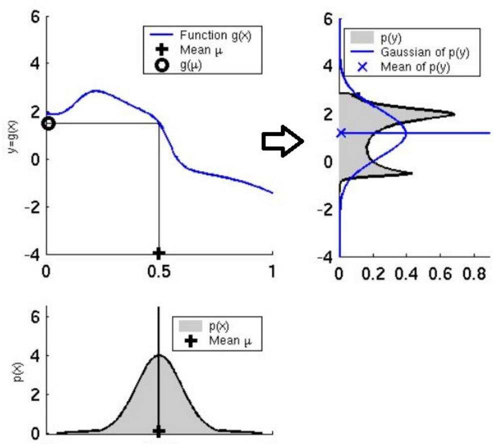

Figure 4: Gaussian + Non Linear Function = Non Gaussian

So if we apply a non linear function it will not end up as a Gaussian Distribution on which we can't apply Kalman Filter anymore. Non linearity destroys the Gaussian and it does not makes sense to compute the mean and variances.

but the functions are non linear so, we will make them Linear by approximation. Here, we will take help of a powerful tool called Taylor Series, which will help us to get a Linear Approximation of the Non Linear function. After applying the approximation what we get is an Extended Kalman Filter.

$$f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots .$$

Figure 5: Taylor Series

we are interested in linearizing, so we are just interested in the first derivative of Taylor series. For every non linear function, we just draw a tangent around the mean and try to approximate the function linearly.

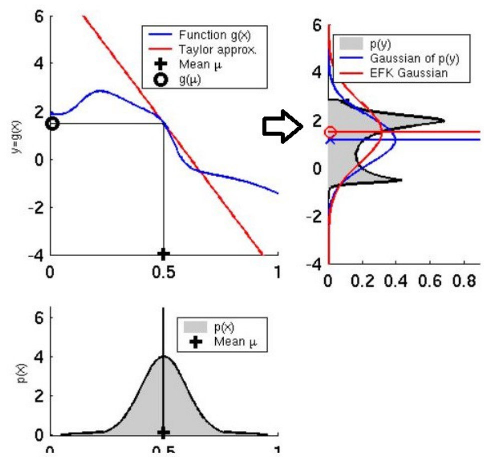

Figure 6: Scenario after applying Taylor's Approximation to Linearize our function

Suppose we have two sensors LIDAR and RADAR. A LIDAR provides us the distance in the form of Cartesian coordinate system. On the other hand, a RADAR provides us the distance and velocity in Polar coordinate system.

Lidar => {px, py}
Radar =>{ ρ, Φ , ρ_dot}

px, py -> Coordinates of object in Cartesian System
ρ -> is the distance to the object
Φ -> is the counter clockwise angle between ρ and x- axis
ρ_dot -> is the change of ρ
The x-axis is always in the direction where the car is heading.

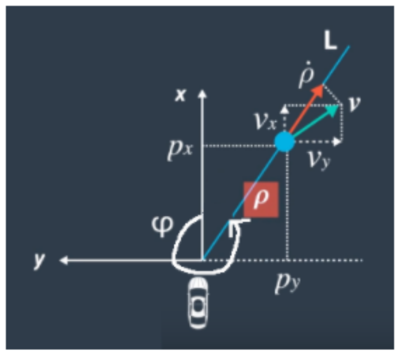

Figure 7. Polar Coordinates as reported by Radar. (Source)

Prediction Step

$x' = F.x + B.\mu + \nu$
$P' = FPF^T + Q$

The prediction step is exactly the same as that of Kalman Filter. It does not matters whether the data is coming from LIDAR or RADAR the prediction step is exactly the same.

Update Step (Only in case of EKF i.e. Non Linear Measurements coming from RADAR)

> *Equation 1:*
>
> ***y= z - h(x')***
>
> *z -> actual measurement in polar coordinates*
> *h -> function that specifies how our speed and position are mapped to polar coordinates*
> *x' -> Predicted Value*
> *y -> Difference between Measured Value and Actual Value*

h(x') - it is is a function that specifies the mapping between our predicted values in Cartesian coordinates and Polar coordinates. This mapping is required because we are predicting in Cartesian coordinates but our measurement (z) that is coming from the sensor is in Polar Coordinates.

$$h(x') = \begin{pmatrix} \rho \\ \phi \\ \dot{\rho} \end{pmatrix} = \begin{pmatrix} \sqrt{p'^2_x + p'^2_y} \\ \arctan(p'_y/p'_x) \\ \frac{p'_x v'_x + p'_y v'_y}{\sqrt{p'^2_x + p'^2_y}} \end{pmatrix}$$

Figure 7: Mapping between Cartesian and Polar coordinates

> *Equation 2:*
>
> $S = H_j P' H_j^T + R$
> $K = P' H_j^T S^{-1}$
>
> *R -> Measurement Noise*
> *K -> Kalman Gain*
> *S-> Total Error*
> *S^{-1} -> The inverse of S*
> *H_j -> The Jacobian Matrix*

H_j

H_j is the Jacobian Matrix. The Jacobian matrix is the first order derivative that we just discussed in Taylor Series. Since here we are dealing with matrices, we need to find differential in the form of a matrix.

> *J_kl = d F_k / dX_l*
>
> *J_kl is the k,l element of the Jacobian matrix, F_k is the kth element of the vector function F, and X_l is the lth element of the vector variable X.*

Here F_k = { *ρ, Φ , ρ_dot}*
X_l = {px,py,vx,vy}

Since in case of RADAR we have 4 measurements, 2 for distance and 2 for velocity.

$$H_j = \begin{bmatrix} \frac{\partial \rho}{\partial p_x} & \frac{\partial \rho}{\partial p_y} & \frac{\partial \rho}{\partial v_x} & \frac{\partial \rho}{\partial v_y} \\ \frac{\partial \varphi}{\partial p_x} & \frac{\partial \varphi}{\partial p_y} & \frac{\partial \varphi}{\partial v_x} & \frac{\partial \varphi}{\partial v_y} \\ \frac{\partial \dot{\rho}}{\partial p_x} & \frac{\partial \dot{\rho}}{\partial p_y} & \frac{\partial \dot{\rho}}{\partial v_x} & \frac{\partial \dot{\rho}}{\partial v_y} \end{bmatrix}$$

Figure 8: Jacobian Matrix

$$H_j = \begin{bmatrix} \frac{p_x}{\sqrt{p_x^2+p_y^2}} & \frac{p_y}{\sqrt{p_x^2+p_y^2}} & 0 & 0 \\ -\frac{p_y}{p_x^2+p_y^2} & \frac{p_x}{p_x^2+p_y^2} & 0 & 0 \\ \frac{p_y(v_x p_y - v_y p_x)}{(p_x^2+p_y^2)^{3/2}} & \frac{p_x(v_y p_x - v_x p_y)}{(p_x^2+p_y^2)^{3/2}} & \frac{p_x}{\sqrt{p_x^2+p_y^2}} & \frac{p_y}{\sqrt{p_x^2+p_y^2}} \end{bmatrix}$$

Figure 9: Jacobian Matrix after applying derivatives

Equation 3:

$x = x' + K.y$

$P = (I - KH_j)P'$

So in case of a LIDAR we will apply a Kalman Filter because the measurements from the sensor are Linear. But in case of a Radar we need to apply Extended Kalman Filter because it includes angles that are non linear, hence we do an approximation of the non linear function using first derivative of Taylor series called Jacobian Matrix (H) . Then we convert our cartesian space to polar space using h(x') and finally we replace H with H in all further equations of a KF.