ME8135 State Estimation for Robotics and Computer Vision HW1

Matthew Lisondra   Sajad Saeedi

May 19, 2020

Question 1

x is a random variable of length K: $x = N(0, 1)$.

a) What type of random variable is the following random variable? $y = x^T x$.

$x = N(0, 1)$ is a Gaussian random variable, standard normalized. x has a mean of 0 and variance of 1.

$$y = x^T x = \begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix}^T \begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix} = x_1^2 + \cdots + x_k^2$$

is a dot product, which is a sum of K random variable, squared, x_i. It is a Chi-squared distribution. We will prove this for varying K.

b) Calculate the mean and variance of y.

The mean is given by $E(y) = E(x^T x) = E(x_1^2 + ... + x_k^2) = E(x_1^2) + ...E(x_k^2)$. To find single $E(x_i^2)$ we use the trick of focusing instead on the variance $Var(x_i)$. See that,

$$\begin{aligned} Var(x_i) &= E(x_i^2) - \big(E(x_i)\big)^2 \\ Var(x_i) &= E(x_i^2) \end{aligned}$$

where we see that $E(x_i) = 0$ because $x = N(0, 1)$ because the mean is 0. Continuing from the last line, we know that $Var(x_i) = 1$ so $E(x_i^2) = 1$. And so we see that,

$$\begin{aligned} E(y) &= E(x^T x) \\ &= E(x_1^2 + \cdots + x_k^2) \\ &= E(x_1^2) + E(x_2^2) + \cdots + E(x_k^2) \\ &= 1 + 1 + \cdots + 1 \\ &= k \end{aligned}$$

And so the mean is $\mu_y = E(y) = k$. The variance is calculated as,

$$\begin{aligned} Var(y) &= E(y^2) - \big(E(y)\big)^2 \\ &= E((x^T x)^2) - \big(E(x^T x)\big)^2 \\ &= E(x^T x x^T x) - \big(E(x^T x)\big)^2 \\ &= E\left(\left(x_1^2 + \cdots + x_k^2\right)^2\right) - k^2 \\ &= E\left(x_1^4 + \cdots + x_k^4 + x_1^2 x_2^2 + \cdots + x_k^2 x_{k-1}^2\right) - k^2 \end{aligned}$$

$$= E\left(\sum_{i=1}^{k} x_i^4\right) + E\left(\sum_{i=1}^{k(k-1)} x_i^2 x_{k-i}^2\right) - k^2$$
$$= \sum_{i=1}^{k} E(x_i^4) + \sum_{i=1}^{k(k-1)} E(x_i^2 x_{k-i}^2) - k^2$$
$$= \sum_{i=1}^{k} 3\sigma^4 + \sum_{i=1}^{k(k-1)} \sigma^2 - k^2$$
$$= 3k + k(k-1) - k^2$$
$$= 3k + k^2 - k - k^2$$
$$= 2k$$

And so the variance is $Var(y) = 2k$.

c) Using Python, plot the PDF of y for $K = 1, 2, 3, 10, 100$.

Recall we mentioned y is a dot product, which is a sum of K random variable, squared, x_i. We said it is a Chi-squared distribution. We will prove this for varying K.

For $K = 1$ we see that it is in fact a Chi-squared distribution and we see that the peak is to the left. We expect that for increasing K the distribution peak will shift to the right and dampen vertically. TAKE NOTE of how the vertical axes values decrease as K goes up. This is because for the probability (or area under this distribution curve) to go to sum 1, the y-value must decrease to compensate for constant 1 area-under-curve.

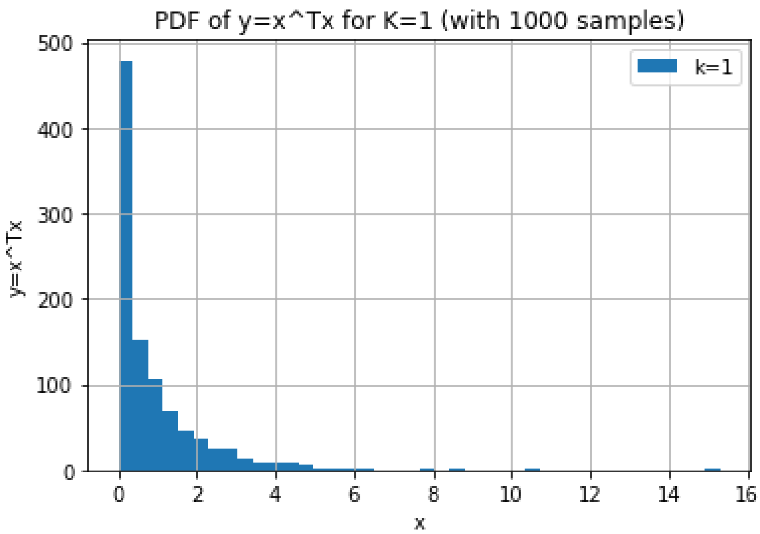

Figure 1: PDFs of $y = x^T x$ with increasing $K = 1$.

Looking at Figure 1-5 in sequential order (as shown below) and then focusing on Figure 5, we see that at $K = 100$ the Chi-squared distribution resembles (for 1000 sampled points) close to a Gaussian. This makes sense and if we sampled more points, we would really showcase the smooth-curve of the Chi-squared distribution.

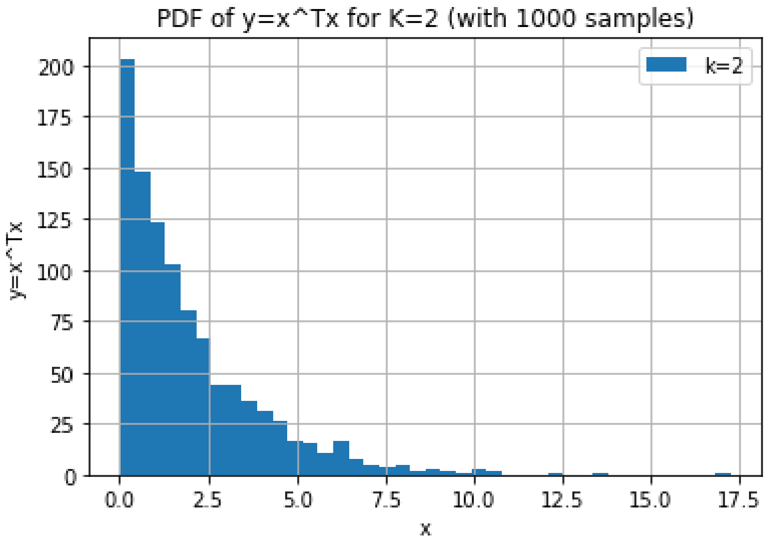

Figure 2: PDFs of $y = x^T x$ with increasing $K = 2$.

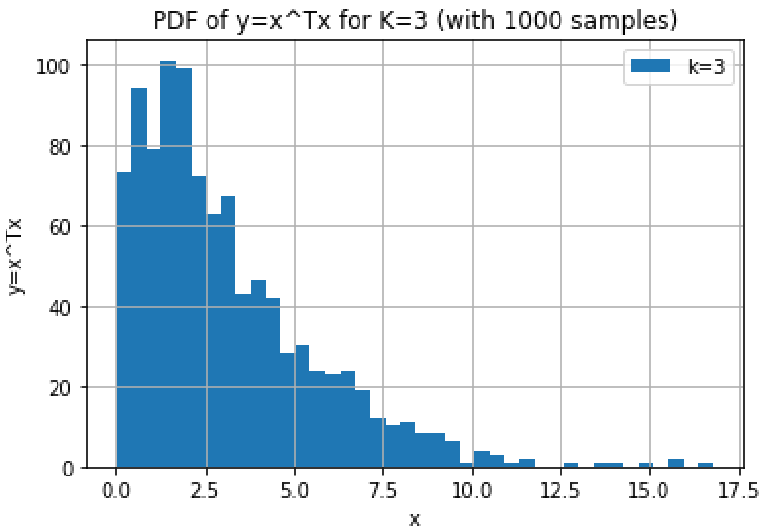

Figure 3: PDFs of $y = x^T x$ with increasing $K = 3$.

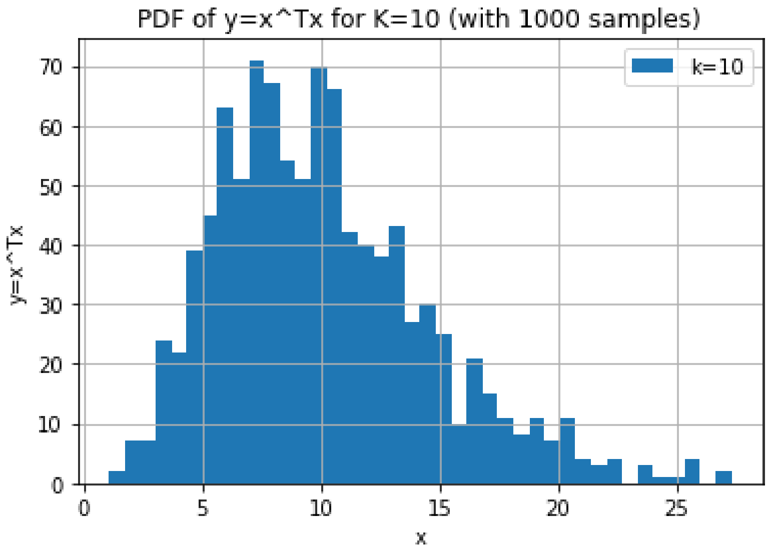

Figure 4: PDFs of $y = x^T x$ with increasing $K = 10$.

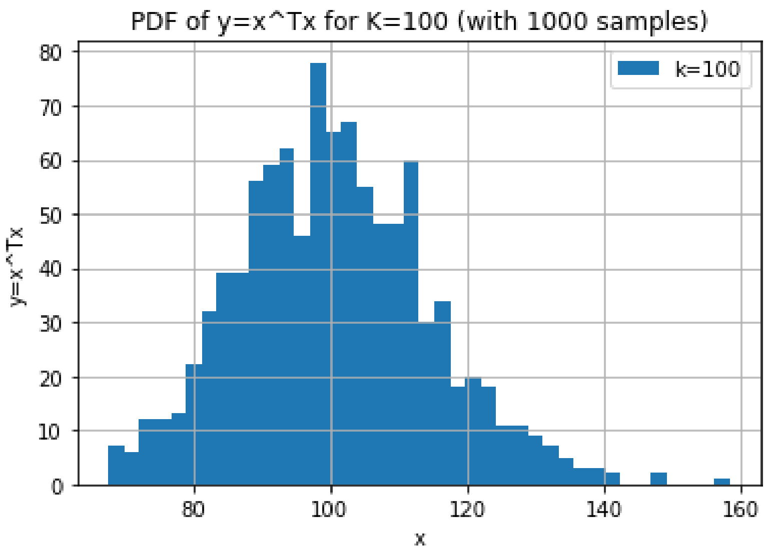

Figure 5: PDFs of $y = x^T x$ with increasing $K = 100$.

Question 2

x is a random variable of length N: $x = N(\mu, \Sigma)$.

a) Assume x is transformed linearly, i.e. $y = Ax$, where A is an $N \times N$ matrix. Calculate the mean and covariance of y. Show the derivations.

$$\begin{aligned} E(y) &= E(Ax) \\ &= AE(x) \\ &= A\mu_x \end{aligned}$$

where $y = Ax$ is a linear map and A is $N \times N$.

And so the mean is $\mu_y = E(y) = A\mu_x$. The variance is calculated as,

$$\begin{aligned} \Sigma_{yy} &= E\left((y-\mu_y)(y-\mu_y)^T\right) \\ &= E\left((Ax - A\mu_x)(Ax - A\mu_x)^T\right) \\ &= E(A(x-\mu_x)(x-\mu_x)^T A^T) \\ &= AE\left((x-\mu_x)(x-\mu_x)^T\right)A^T \\ &= A\Sigma_{xx}A^T \end{aligned}$$

And so the covariance is $Var(y) = A\Sigma_{xx}A^T$.

b) Repeat a) when $y = A_1x + A_2x$.

$$\begin{aligned} E(y) &= E(A_1x + A_2x) \\ &= E(A_1x) + E(A_2x) \\ &= A_1E(x) + A_2E(x) \\ &= A_1\mu_x + A_2\mu_x \\ &= A\mu_x \end{aligned}$$

where $y = A_1x + A_2x$ is a linear map and A_1, A_2 are $N \times N$.

And so the mean is $\mu_y = E(y) = A_1\mu_x + A_2\mu_x = A\mu_x$. The variance is calculated as,

$$\begin{aligned} \Sigma_{yy} &= E\left((y-\mu_y)(y-\mu_y)^T\right) \\ &= E((A_1x + A_2x - A_1\mu_x + A_2\mu_x)(A_1x + A_2x - A_1\mu_x + A_2\mu_x)^T) \\ &= E((A_1 + A_2)(x-\mu_x)(x-\mu_x)^T(A_1 + A_2^T)) \\ &= (A_1 + A_2)E(x-\mu_x)(x-\mu_x)^T(A_1 + A_2)^T \\ &= (A_1 + A_2)\Sigma_{xx}(A_1 + A_2)^T \\ &= A\Sigma_{xx}A^T \end{aligned}$$

And so the covariance is $Var(y) = (A_1 + A_2)\Sigma_{xx}(A_1 + A_2)^T = A\Sigma_{xx}A^T$.

c) If x is transformed by a nonlinear differentiable function, i.e. $y = f(x)$, compute the covariance of y. Show the derivation.

Process of passing a Gaussian PDF through a stochastic non-linearity is done by computing,

$$p(y) = \int_{-\infty}^{\infty} p(y|x)p(x)dx$$

with the conditions that,

$$p(y|x) = N(f(x), R)$$
$$p(x) = N(\mu_x, \Sigma_{xx})$$

and $f(.)$ is a nonlinear map $f : x \mapsto y$ that is then corrupted by zero-mean Gaussian noise with covariance R (we will derive with general R but in our case we will later set this to $R = 0$). It must be noted that for the above, $p(y)$ cannot be computed in closed form for every $f(.)$. We must linearize the nonlinear map such that,

$$f(x) = \mu_y + \bar{F}(x - \mu_x)$$
$$\bar{F} = \left.\frac{\partial f(x)}{\partial x}\right|_{x=\mu_x}$$
$$\mu_y = f(\mu_x)$$

Note $\bar{F}$ is the Jacobian of $f(.)$ with respect to x. And so with this linearization we continue and do,

$$\begin{aligned}
p(y) &= \int_{-\infty}^{\infty} p(y|x)p(x)dx \\
&= \eta \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}\left(y - \left(\mu_y + \bar{F}(x - \mu_x)\right)\right)^T \times R^{-1}\left(y - \left(\mu_y + \bar{F}(x - \mu_x)\right)\right)\right) \\
&\qquad \times \exp\left(-\frac{1}{2}(x - \mu_x)^T \Sigma_{xx}^{-1}(x - \mu_x)\right) dx \\
&= \eta \exp\left(-\frac{1}{2}(y - \mu_y)^T R^{-1}(y - \mu_y)\right) \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2}(x - \mu_x)^T(\Sigma_{xx}^{-1} + \bar{F}^T R^{-1}\bar{F})(x - \mu_x)\right) \\
&\qquad \times \exp\left((y - \mu_y)^T R^{-1}\bar{F}(x - \mu_x)\right) dx
\end{aligned}$$

where η is a normalization constant. We must define F such that,

$$F^T(\bar{F}^T R^{-1}\bar{F} + \Sigma_{xx}^{-1}) = R^{-1}\bar{F}$$

and in doing so we are able to complete the square inside the integral such that,

$$\begin{aligned}
&\exp\left(-\frac{1}{2}(x - \mu_x)^T(\Sigma_{xx}^{-1} + \bar{F}^T R^{-1}\bar{F} + \Sigma_{xx}^{-1})(x - \mu_x)\right) \times \exp\left((y - \mu_y)^T R^{-1}\bar{F}(x - \mu_x)\right) \\
&= \exp\left(-\frac{1}{2}\left((x - \mu_x) - F(y - \mu_y)\right)^T \times (\bar{F}^T R^{-1}\bar{F} + \Sigma_{xx}^{-1})\left((x - \mu_x) - F(y - \mu_y)\right)\right) \\
&\qquad \times \exp\left(\frac{1}{2}(y - \mu_y)^T F^T(\bar{F}^T R^{-1}\bar{F} + \Sigma_{xx}^{-1})F(y - \mu_y)\right)
\end{aligned}$$

We simplify further the integral now, and we absorb terms into the η constant to get,

$$\begin{aligned}
p(y) &= \rho \exp\left(-\frac{1}{2}(y - \mu_y)^T \times (R^{-1} - F^T(\bar{F}^T R^{-1}\bar{F} + \Sigma_{xx}^{-1})F(y - \mu_y)\right) \\
&= \rho \exp\left(-\frac{1}{2}(y - \mu_y)^T \times (R^{-1} - R^{-1}\bar{F}(\bar{F}^T R^{-1}\bar{F} + \Sigma_{xx}^{-1})^{-1}\bar{F}^T R^{-1})(y - \mu_y)\right)
\end{aligned}$$

$$R^{-1} - R^{-1}\bar{F}(\bar{F}^T R^{-1}\bar{F} + \Sigma_{xx}^{-1})^{-1}\bar{F}^T R^{-1} = (R + \bar{F}\Sigma_{xx}\bar{F}^T)^{-1}$$

$$= \rho \exp\left(-\frac{1}{2}(y - \mu_y)^T(R + \bar{F}\Sigma_{xx}\bar{F}^T)^{-1}(y - \mu_y)\right)$$

where ρ is the new normalization constant. This is a Gaussian for y:

$$y \sim N(\mu_y, \Sigma_{yy}) = N(f(\mu_x), R + \bar{F}\Sigma_{xx}\bar{F}^T)$$

And so the covariance is $Cov(y) = R + \bar{F}\Sigma_{xx}\bar{F}^T$ (remember, we will set $R = 0$ later. This derivation is with general R).

d) Apply c) when

$$x = \begin{pmatrix} \rho \\ \theta \end{pmatrix} \qquad \Sigma = \begin{pmatrix} \sigma_{\rho p}^2 & \sigma_{\rho\theta}^2 \\ \sigma_{\rho\theta}^2 & \sigma_{\theta\theta}^2 \end{pmatrix} \qquad y = \begin{pmatrix} \rho cos\theta \\ \rho sin\theta \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$

Compute the covariance of y analytically. This models how range-bearing measurements in the polar coordinate frame are converted to a Cartesian coordinate frame.

Generally the covariance is $Cov(y) = R + \bar{F}\Sigma_{xx}\bar{F}^T$. We mentioned before though that we did this with general R. We now set $R = 0$ and calculate the covariance as $Cov(y) = \bar{F}\Sigma_{xx}\bar{F}^T$.

We see that to calculate the covariance we first must calculate the Jacobian:

$$\begin{aligned}\bar{F} &= \begin{pmatrix} \dfrac{\partial f_1}{\partial \rho} & \dfrac{\partial f_1}{\partial \theta} \\ \dfrac{\partial f_2}{\partial \rho} & \dfrac{\partial f_2}{\partial \theta} \end{pmatrix} \\ &= \begin{pmatrix} cos\theta & -\rho sin\theta \\ sin\theta & \rho cos\theta \end{pmatrix}\end{aligned}$$

We then continue and the covariance is found to be,

$$\begin{aligned} Cov(y) &= \bar{F}\Sigma_{xx}\bar{F}^T \\ &= \begin{pmatrix} cos\theta & -\rho sin\theta \\ sin\theta & \rho cos\theta \end{pmatrix} \begin{pmatrix} \sigma_{\rho p}^2 & \sigma_{\rho\theta}^2 \\ \sigma_{\rho\theta}^2 & \sigma_{\theta\theta}^2 \end{pmatrix} \begin{pmatrix} cos\theta & -\rho sin\theta \\ sin\theta & \rho cos\theta \end{pmatrix}^T \\ &= \begin{pmatrix} cos\theta(\sigma_{\rho p}^2 cos\theta - \sigma_{\rho\theta}^2 \rho cos\theta) - \rho cos\theta(\sigma_{\rho\theta}^2 cos\theta - \sigma_{\theta\theta}^2 \rho cos\theta) & (\sigma_{\rho p}^2 cos\theta - \sigma_{\rho\theta}^2 \rho cos\theta) sin\theta + (\sigma_{\rho\theta}^2 cos\theta - \sigma_{\theta\theta}^2 \rho cos\theta)\rho cos\theta \\ cos\theta(\sigma_{\rho p}^2 sin\theta + \sigma_{\rho\theta}^2 \rho cos\theta) - \rho cos\theta(\sigma_{\rho\theta}^2 sin\theta + \sigma_{\theta\theta}^2 \rho cos\theta) & (sin\theta\sigma_{\rho p}^2 + \rho cos\theta\sigma_{\rho\theta}^2) sin\theta + (sin\theta\sigma_{\rho\theta}^2 + \rho cos\theta\sigma_{\theta\theta}^2)\rho cos\theta \end{pmatrix} \end{aligned}$$

e) Simulate d) using the Monte Carlo simulation, i.e. assume

$$x = \begin{pmatrix} 1m \\ 0.5^\circ \end{pmatrix} \qquad \Sigma = \begin{pmatrix} 0.01 & 0 \\ 0 & 0.005 \end{pmatrix}$$

Sample 1000 points from this distribution and plot the transformed results on $x - y$ coordinates. Plot the uncertainty ellipse, calculated from part d). Overlay the ellipse on the point samples.

We start by calculating y as,

$$y = \begin{pmatrix} (1m)cos0.5^\circ \\ (1m)sin0.5^\circ \end{pmatrix} = \begin{pmatrix} 0.99996192306 \\ 0.00872653549 \end{pmatrix}$$

We continue by next calculating the Jacobian as see that,

$$\bar{F} = \begin{pmatrix} cos0.5^\circ & -(1m)sin0.5^\circ \\ sin0.5^\circ & (1m)cos0.5^\circ \end{pmatrix}$$
$$= \begin{pmatrix} 0.99996192306 & -0.00872653549 \\ 0.00872653549 & 0.99996192306 \end{pmatrix}$$

We now calculate the covariance of y and we see that it is given by,

$$Cov(y) = \bar{F}\Sigma_{xx}\bar{F}^T$$
$$= \begin{pmatrix} 0.99996192306 & -0.00872653549 \\ 0.00872653549 & 0.99996192306 \end{pmatrix} \begin{pmatrix} 0.01 & 0 \\ 0 & 0.005 \end{pmatrix} \begin{pmatrix} 0.99996192306 & -0.00872653549 \\ 0.00872653549 & 0.99996192306 \end{pmatrix}^T$$
$$= \begin{pmatrix} 0.00999 & 0.00004 \\ 0.00004 & 0.00500 \end{pmatrix}$$

The plot of our transformed results is given on $x - y$ coordinates. The uncertainty ellipse is found and surrounds the sampled distribution points of y:

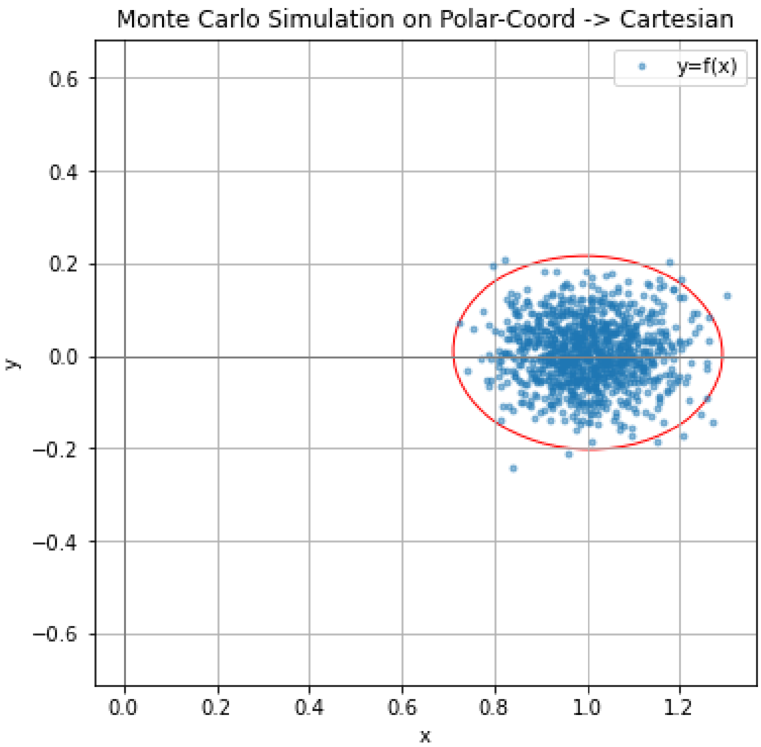

Figure 6: Sampled 1000 points (by Monte Carlo simulation) the y distribution and put uncertainty ellipse. uncertainty ellipse encloses 98.9% of the points if the data is normally distributed. [2].

References

[1] Barfoot, Timothy D. State estimation for robotics. Cambridge University Press, 2017.

[2] "Plot a Confidence Ellipse of a Two-Dimensional Dataset." Plot a Confidence Ellipse of a Two-Dimensional Dataset - Matplotlib 3.7.1 Documentation, *matplotlib.org/stable/gallery/statistics/confidence_ellipse.html*. Accessed 19 May 2023.