# Forward-Backward Kolmogorov Equations Library

## Introduction

This notebook provides an explanation of the Forward-Backward Kolmogorov Equations library that has been created. The library is designed to solve stochastic differential equations and related Kolmogorov equations.

### Setting Up

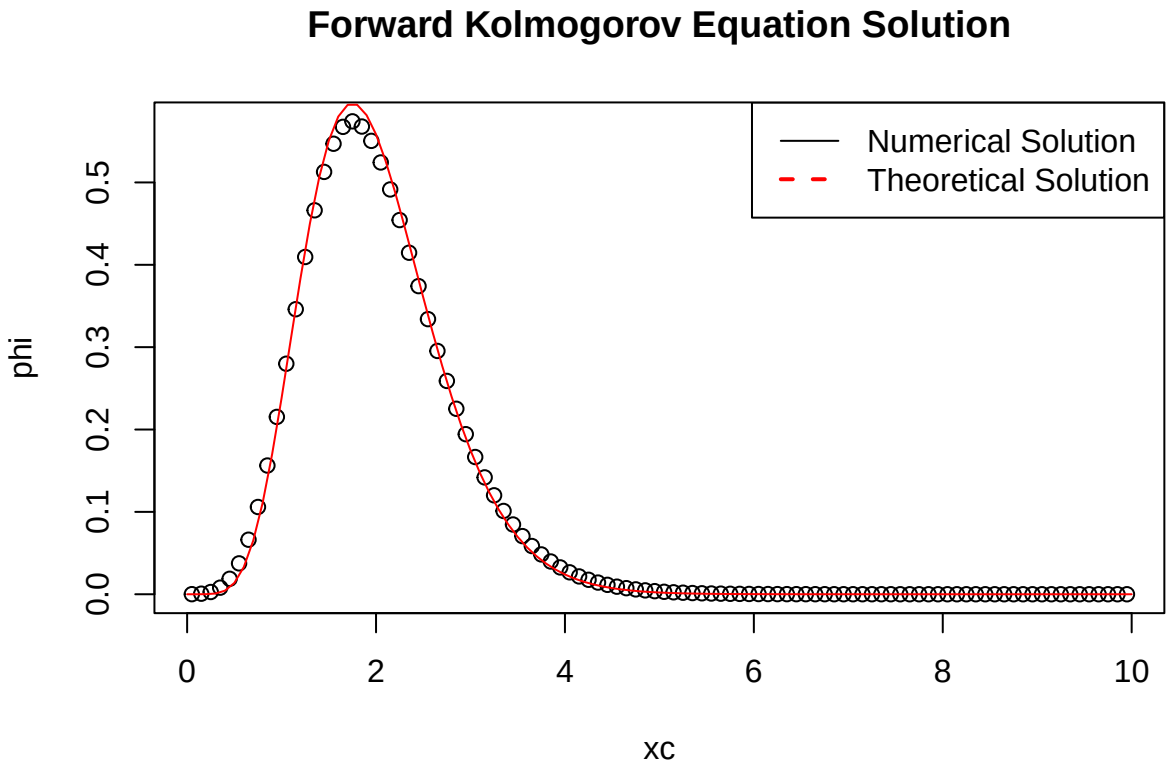
```
# Setting the working directory
setwd('C:/Users/utilisateur/Documents/DTU/Première Année 2023-2024/Cours/Stochastic Differential Equati

# Sourcing required scripts
source('kolmogorov-equations.R')
```

### Parameters

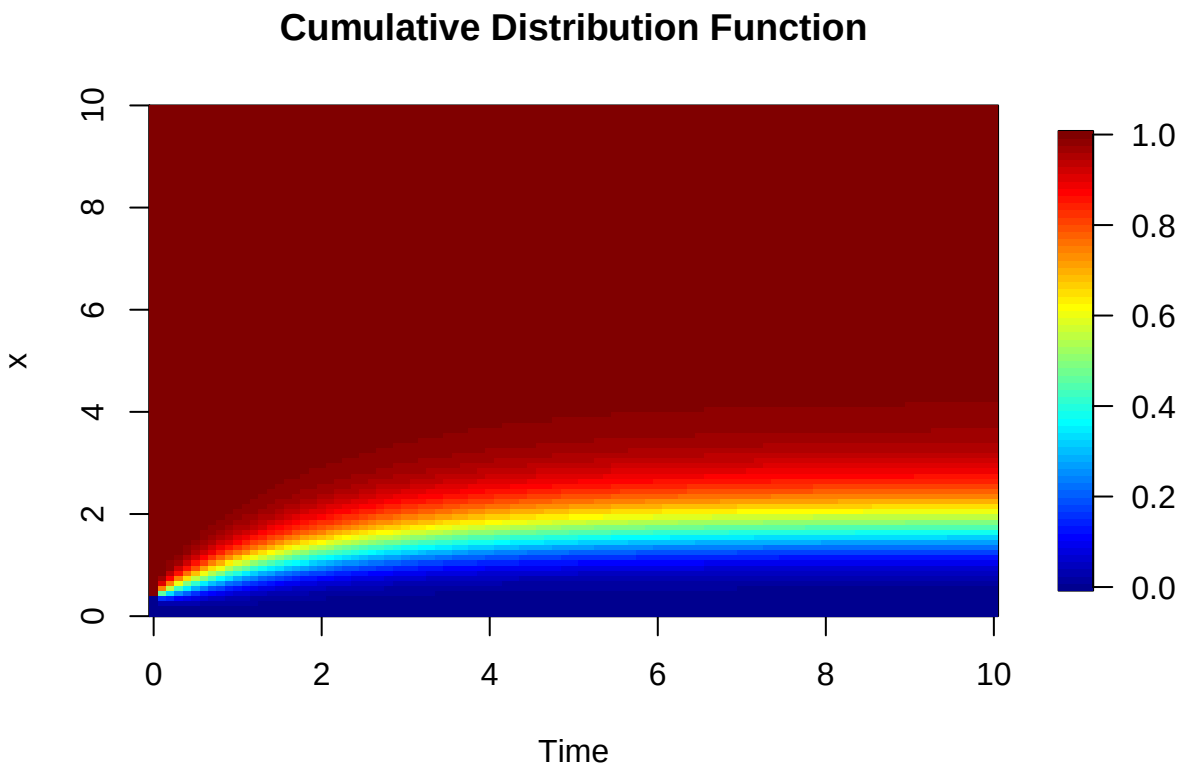
```
# Parameters
xi <- 2
gamma <- 1/2
lambda <- 1/2

# Drift and intensity functions
f <- function(x) lambda * (xi - x)
g <- function(x) gamma * sqrt(abs(x))

# Diffusivity and its spatial derivative
D <- function(x) 0.5 * gamma^2 * x
Dp <- function(x) 0.5 * gamma^2

# Advective flow field
u <- function(x) f(x) - Dp(x)
```

## Forward Kolmogorov Equation

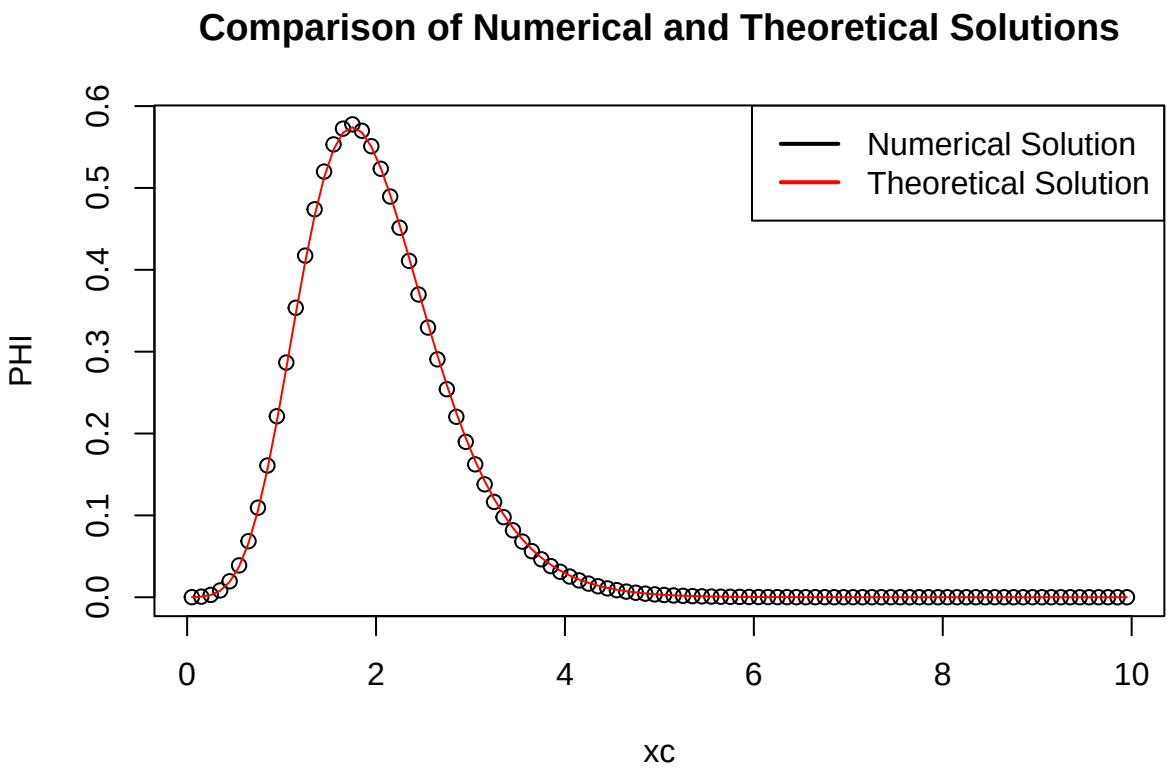
```
result <- forwardkolmogorov(u, D)
G <- result$G
phi <- result$phi

plot(xc, phi, main="Forward Kolmogorov Equation Solution", xlab="xc", ylab="phi")
curve(dgamma(x, rate=2*lambda/gamma^2, shape=2*lambda*xi/gamma^2),
      from=0, to=xmax, add=TRUE, col="red")
legend("topright", legend=c("Numerical Solution", "Theoretical Solution"),
       col=c("black", "red"), lty=c(1, 2), lwd=c(1, 2))
```




```
x0 <- xi/4

## Initial condition for the FKE is a dirac
phi0 <- numeric(length(xc))
phi0[sum(xc<x0)] <- 1

## Time grid
tv <- seq(0,10,0.1)

## Solve the Forward Kolmogorov Equation
PHI <- solve_forwardEquation(tv,phi0,G)

CDF <- apply(PHI*dx, 2, cumsum)
image.plot(tv, xc, t(CDF), main="Cumulative Distribution Function", xlab="Time", ylab="x")
```

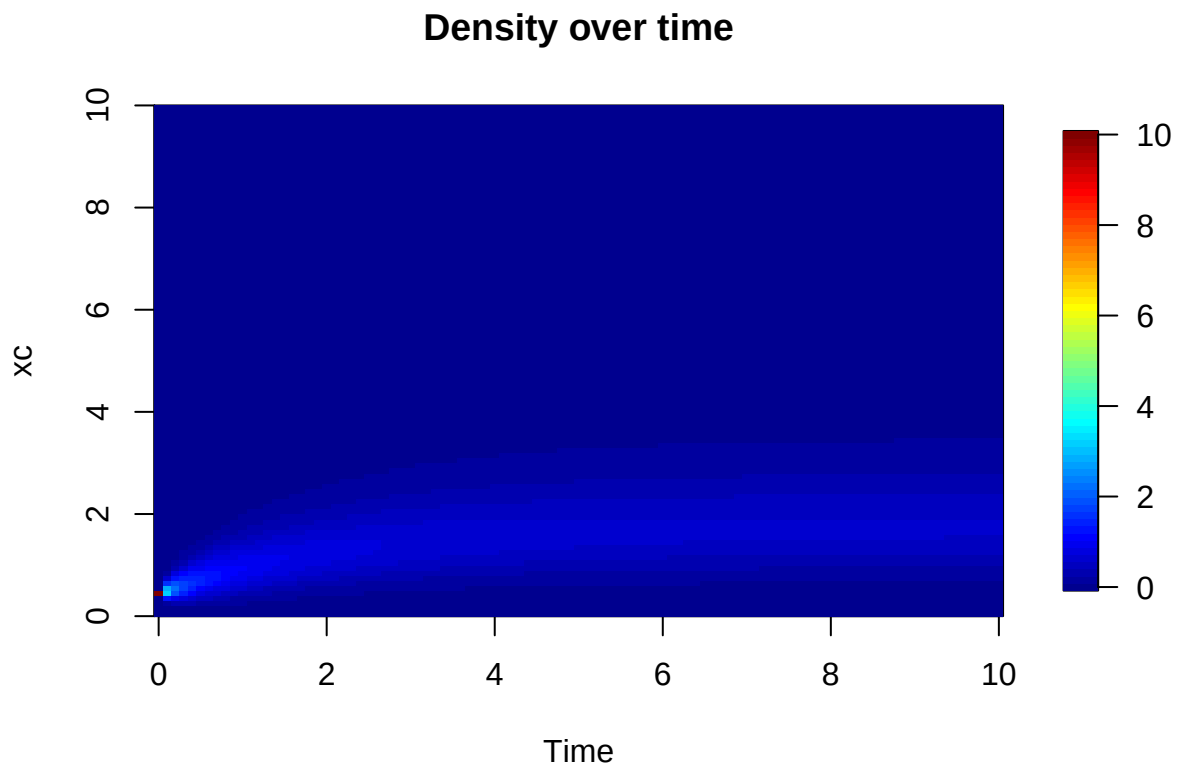



```r
plot(xc, PHI[, length(tv)], col="black",
     main="Comparison of Numerical and Theoretical Solutions",
     xlab="xc", ylab="PHI")

# Ajout de la courbe théorique
lines(xc, phi, col="red")

# Ajout d'une légende
legend("topright", legend=c("Numerical Solution", "Theoretical Solution"),
       col=c("black", "red"), lwd=c(2, 2))
```



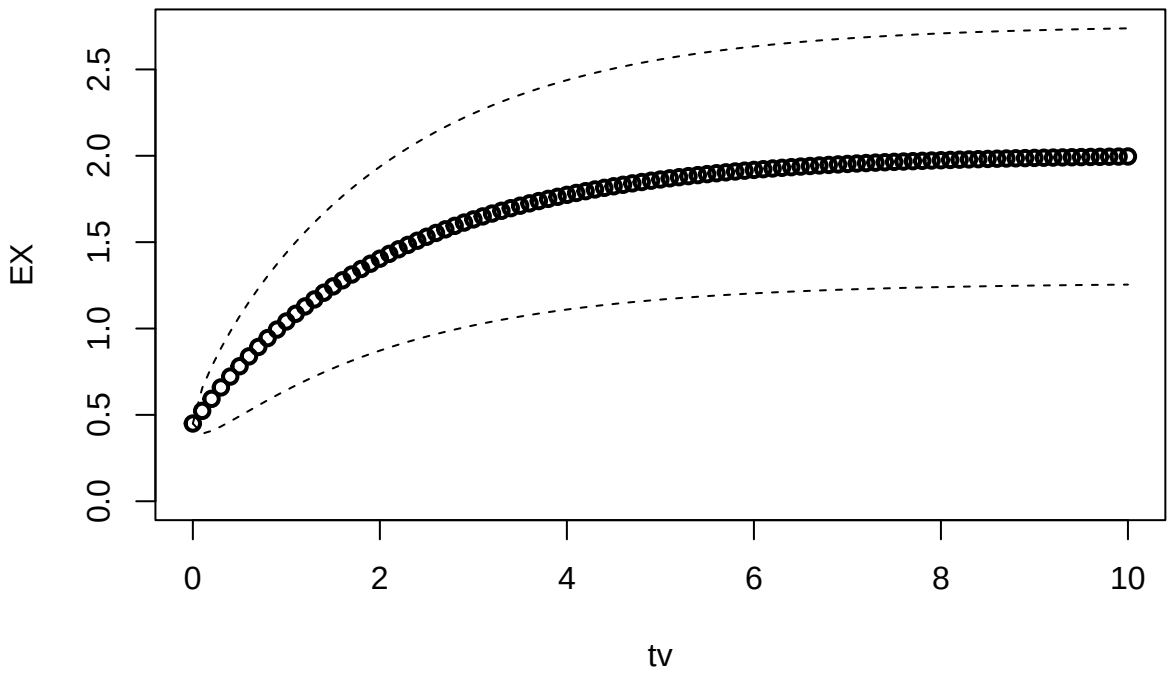

### Solving Forward Kolmogorov Equation

```
# Initial conditions
x0 <- xi/4
phi0 <- numeric(length(xc))
phi0[sum(xc < x0)] <- 1

# Time grid
tv <- seq(0, 10, 0.1)

# Solving the Forward Kolmogorov Equation
PHI <- solve_forwardEquation(tv, phi0, G)

# Plotting the solution
image.plot(tv, xc, t(PHI), main="Density over time", xlab="Time", ylab="xc")
```



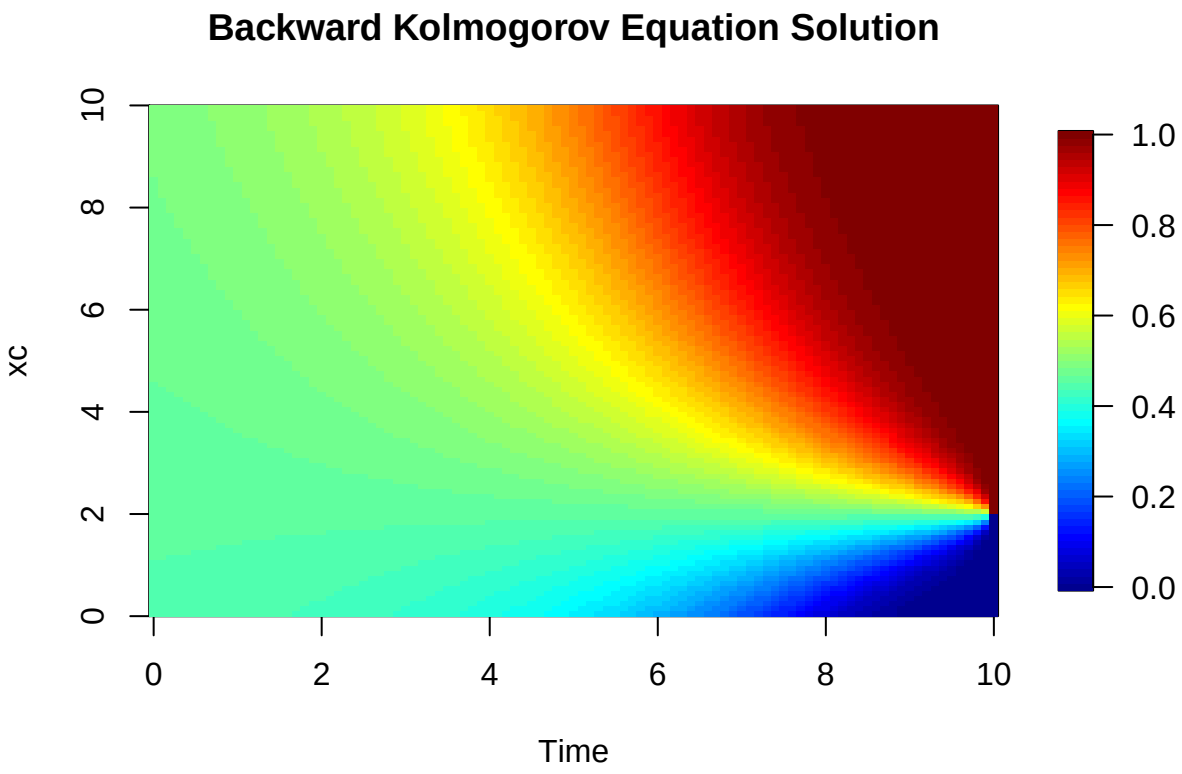

### Mean, Variance, and Standard Deviation

```
# Computing moments
EX <- computeMoment(PHI, 1)
VX <- computeVariance(PHI)
sX <- computeStandardDeviation(PHI)

# Plotting the results
plot(tv, EX, lwd=2, ylim=c(0, max(EX + sX)))
lines(tv, EX + sX, lty="dashed")
lines(tv, EX - sX, lty="dashed")
```




## Backward Kolmogorov Equation

```
h <- (xc >= 2)
T <- tail(tv, 1)
psi <- solve_backwardEquation(tv, h, G)

# Plotting the solution with title and labels
image.plot(tv, xc, t(psi), main="Backward Kolmogorov Equation Solution", xlab="Time", ylab="xc")
```




## Conclusion

This notebook provides an overview of the Forward-Backward Kolmogorov Equations library, its key functions, and visualizations of the solutions. It serves as a guide for users to understand and utilize the capabilities of the library in solving stochastic differential equations.