Summer Project

Contents

List of Figures

List of Tables

1 Gaussian curve and its fourier transform

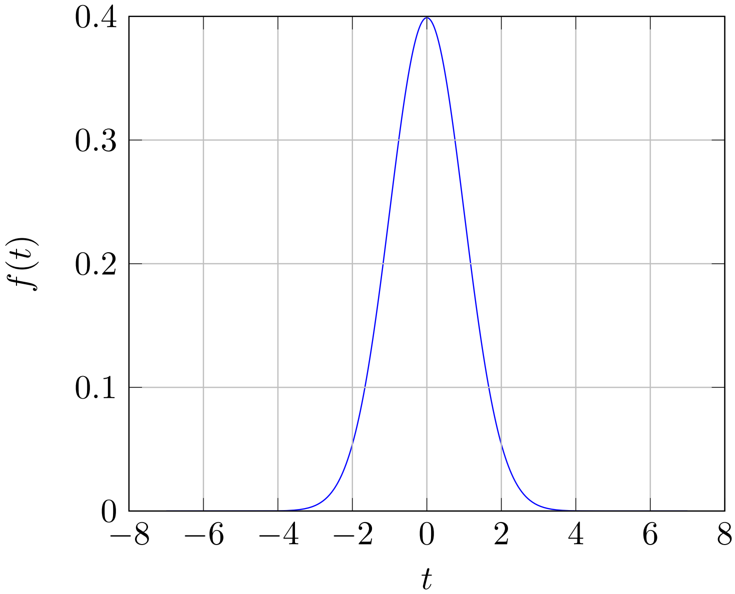

Figure 1: Standard Normal curve

Code Block

```
import matplotlib.pyplot as plt
import numpy as np
plt.style.use("classic")

def f(t):
    return np.exp(-(t)**2/2)/(np.sqrt(2*np.pi))

xv = np.linspace(-7,7,1000)
yv = f(xv)

plt.plot(xv, yv, lw=1)
plt.xlabel("$t$")
plt.ylabel("$f(t)$")
plt.grid(True)
```

1.1 Fourier transform of Standard normal

If $f(t) = \int_{-\infty}^{\infty} g(\omega)e^{i\omega t}d\omega$, then fourier transform of that is $g(\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt$. Here, $f(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$

$$
\begin{aligned}
g(\omega) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} f(t)e^{-i\omega t}dt \\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty} f(t)\cos(\omega t)dt - i\frac{1}{2\pi}\int_{-\infty}^{\infty} f(t)\sin(\omega t)dt
\end{aligned}
$$

Code Block

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import quad

def f(t):
    return np.exp(-(t)**2/2)/(np.sqrt(2*np.pi))

def ft(y):
    int_re = lambda t: f(t)*np.sin(y*t)
    int_im = lambda t: f(t)*np.cos(y*t)
    g_re = quad(int_re,-np.inf,np.inf)[0]/(2*np.pi)
    g_im = quad(int_im,-np.inf,np.inf)[0]/(2*np.pi)
    return g_re + 1j*g_im
g = np.frompyfunc(ft, 1, 1)

xv = np.linspace(-7,7,1000)
yv = np.abs(g(xv))
plt.plot(xv, yv, lw=1)
plt.xlabel("$\omega$")
plt.ylabel("$g(\omega)$")
plt.grid(True)
```

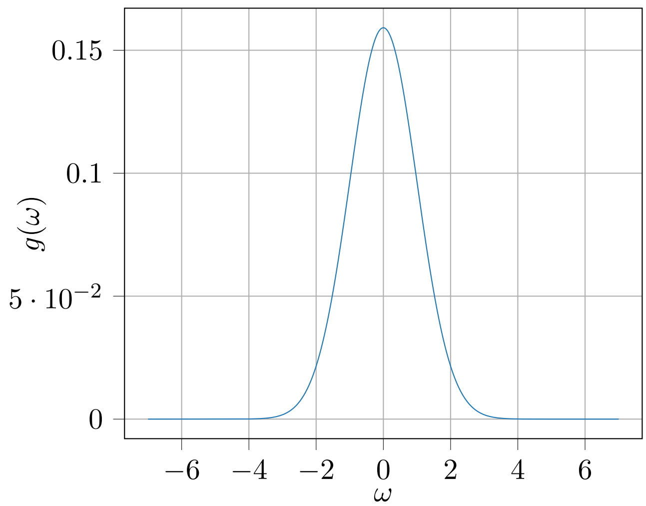

Figure 2: Fourier transform of Standard Normal