

```

In [2]: import os
import urllib.request
import thinkstats2
import thinkplot
import nsfg # Ensure ThinkStats2 library is installed and configured

# Download the files
if not os.path.exists("2002FemResp.dct"):
    urllib.request.urlretrieve(
        "https://github.com/AllenDowney/ThinkStats2/raw/master/code/2002FemResp.dct"
    )
if not os.path.exists("2002FemResp.dat.gz"):
    urllib.request.urlretrieve(
        "https://github.com/AllenDowney/ThinkStats2/raw/master/code/2002FemResp.dat.gz"
    )

# Read the data
resp = nsfg.ReadFemResp()

# Create a PMF
pmf = thinkstats2.Pmf(resp.numkdhh, label="numkdhh")

# Plot the PMF
thinkplot.Pmf(pmf)
thinkplot.Config(xlabel="Number of children", ylabel="PMF")

# Define BiasPmf if not already available
def BiasPmf(pmf, label=None):
    """Computes a biased PMF."""
    new_pmf = pmf.Copy(label=label)
    for x, p in pmf.Items():
        new_pmf.Mult(x, x)
    new_pmf.Normalize()
    return new_pmf

# Compute biased PMF
biased = BiasPmf(pmf, label="biased")

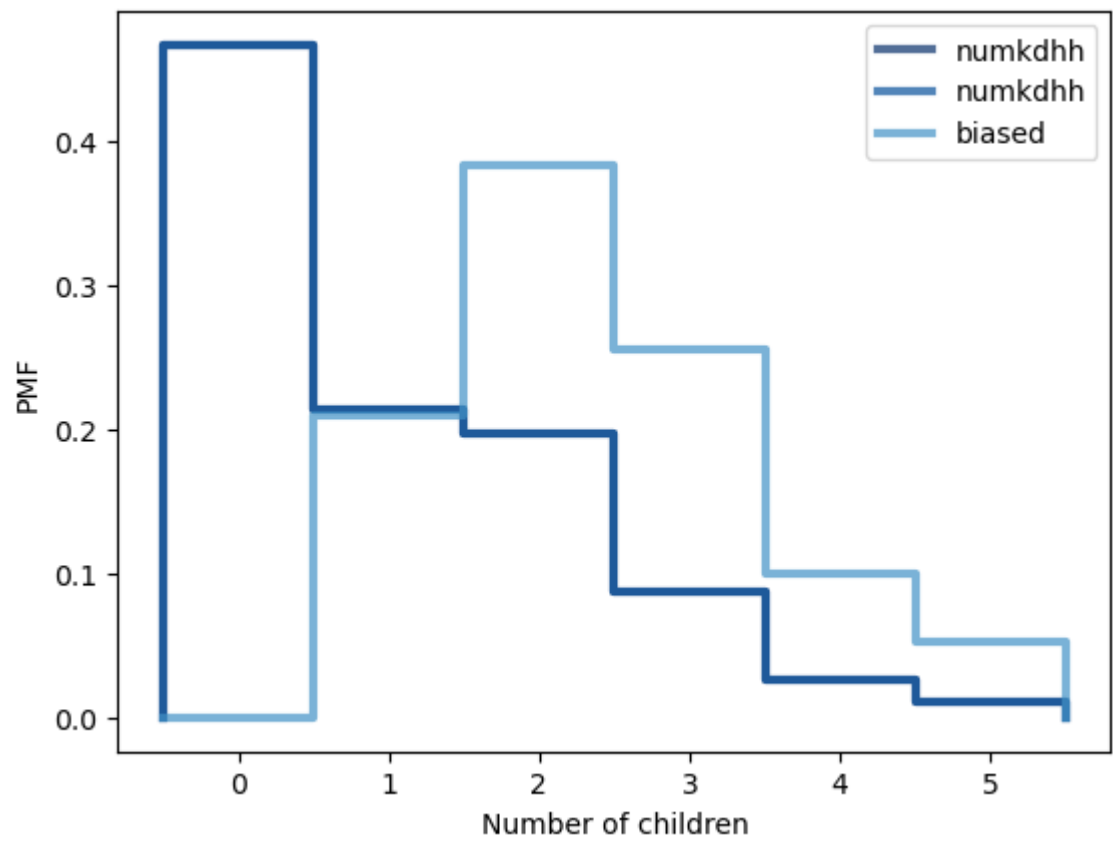
# Plot both PMFs
thinkplot.PrePlot(2)
thinkplot.Pmfs([pmf, biased])
thinkplot.Config(xlabel="Number of children", ylabel="PMF")

# Calculate means
original_mean = pmf.Mean()
biased_mean = biased.Mean()

print(f"Original Mean: {original_mean}")
print(f"Biased Mean: {biased_mean}")

Original Mean: 1.024205155043831
Biased Mean: 2.403679100664282

```



In []:

In []: