Applied Bayesian Data Analysis — Exercise 2

Kim Albertsson

CERN and Luleå University of Technology

kim.albertsson@ltu.se

November 13, 2019

Task A

Task A — Question 1

a Recreate Figure 4.1 in the book by creating a function that simulates coin tosses from a fair coin.

b Modify the function so that the coin is biased with $\theta = 0.25$ (θ is proportion of heads).

c Sample tosses with the biased coin and plot a histogram with relative frequencies and true probability mass.

Recreate Figure 4.1 in the book by creating a function that simulates coin tosses from a fair coin. (Bonus: Also show for biased coin with $\theta = 0.25$!)

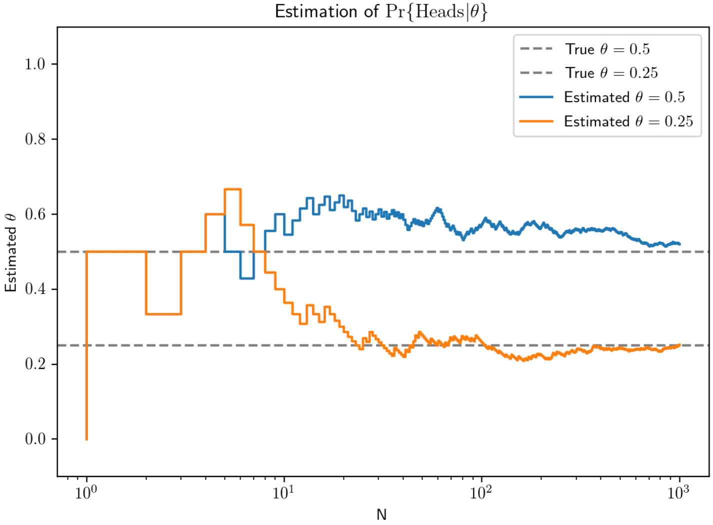
Task A — Q 1A-B

```python
import numpy.random.uniform as uniform
def coin(shape=(1,), t=0.5):
    '''
    t: Proportion of 1's in output
        This means `coin(x, t).sum()/x` tends to `t`
        as `x` tends to infinity.
    '''
    assert(0.0 <= t <= 1.0)
    return uniform(shape=shape, 1.0) <= (1.0-t)
```

Task A — Q 1A-B

```
import numpy as np
import matplotlib.pyplot as plt

np.random.seed(1234)
X = np.asarray(range(1, 1001))
Y1 = coin(shape=1000, t=0.5)
Y2 = coin(shape=1000, t=0.25)

plt.axhline(0.5, linestyle='dashed',
            color='grey', label='True␣t=0.5')
plt.axhline(0.25, linestyle='dashed',
            color='grey', label='True␣t=0.25')
plt.step(X, Y1.cumsum()/X, label='Est.␣t=0.5')
plt.step(X, Y2.cumsum()/X, label='Est.␣t=0.25')
plt.show()
```

Task A — Q 1A-B



Estimation of $\Pr\{\text{Heads}|\theta\}$
True $\theta = 0.5$
True $\theta = 0.25$
Estimated $\theta = 0.5$
Estimated $\theta = 0.25$
Estimated θ
1.0
0.8
0.6
0.4
0.2
0.0
10^0
10^1
10^2
10^3
N

Task A — Q 1C

Sample tosses with the biased coin ($\theta = 0.25$). Compare histogram with relative frequencies to the true probability mass function.

What will happen when n grows?

Hypothesis: Histogram and pmf approach each other!

Task A — Q 1C

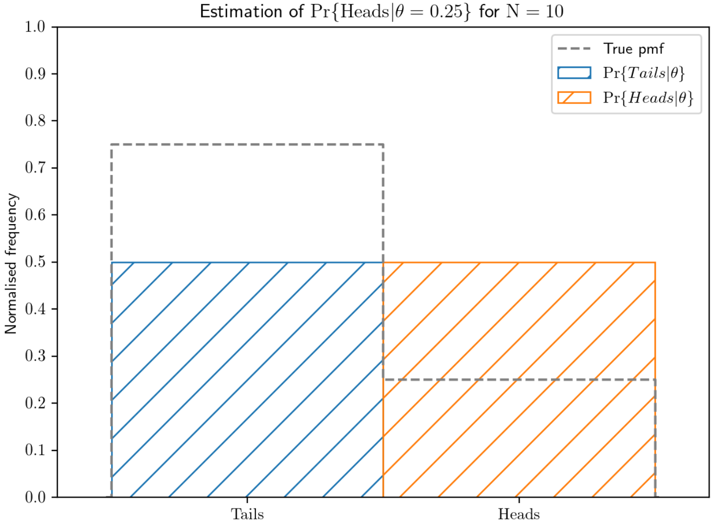
```python
import numpy as np
import matplotlib.pyplot as plt

np.random.seed(1234)
X = coin(100, 0.25)

bins = [0, 0.5, 1.0]
plt.hist(X[X==0], histtype='step', bins=bins,
         weights=...)
plt.hist(X[X==1], histtype='step', bins=bins,
         weights=...)
```

The pmf should sum to 1, hence weighting is required.

Task A — Q 1C

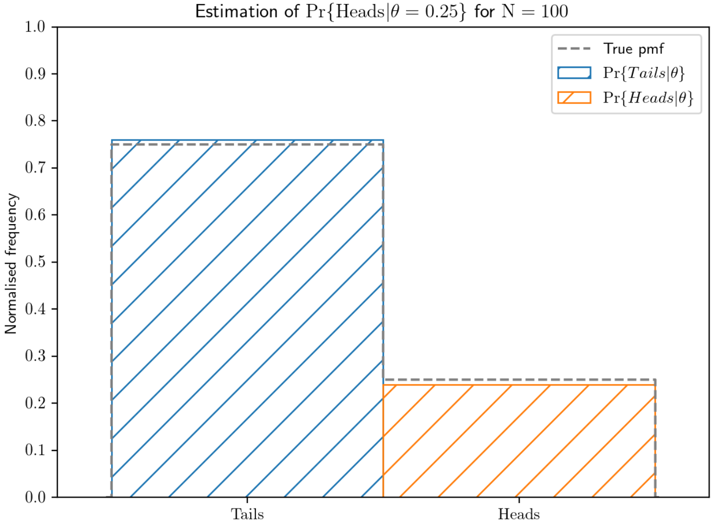
```python
import scipy.interpolate.interp1d as interp1d
pmf = interp1d([-10., 0., 0.5, 1., 10.],
               [0., 0.75, 0.25, 0.0, 0.0],
               kind='zero')
x_pmf = np.linspace(-0.01, 1.01, 1001)
plt.step(x_pmf, pmf(x_pmf), where='post')
```

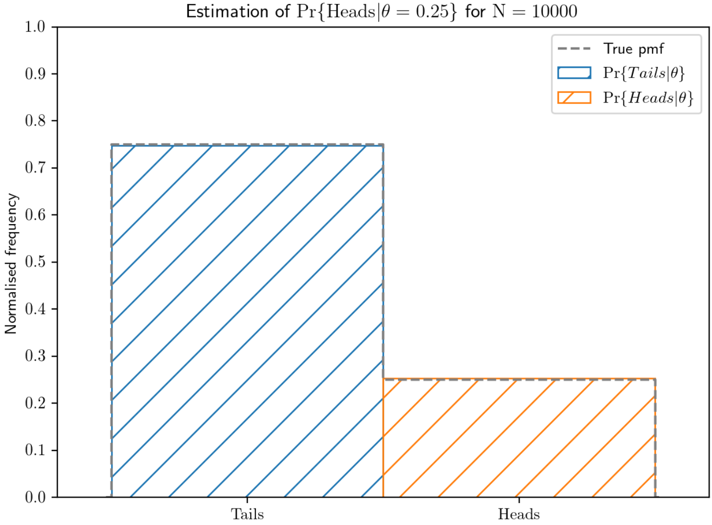
Task A — Q 1C



Estimation of $\Pr\{\text{Heads}|\theta = 0.25\}$ for $\text{N} = 10$
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Normalised frequency
True pmf
$\Pr\{Tails|\theta\}$
$\Pr\{Heads|\theta\}$
Tails
Heads

Task A — Q 1C


Estimation of $\Pr\{\text{Heads}|\theta = 0.25\}$ for $\text{N} = 100$
Normalised frequency
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Tails
Heads
True pmf
$\Pr\{Tails|\theta\}$
$\Pr\{Heads|\theta\}$


Estimation of $\Pr\{\text{Heads}|\theta = 0.25\}$ for $\text{N} = 10000$
True pmf
$\Pr\{Tails|\theta\}$
$\Pr\{Heads|\theta\}$
Normalised frequency
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Tails
Heads

Task A — Question 2

- a Draw 10000 samples from $\mathcal{N}(\mu = 3.4, \sigma^2 = 3.0)$. Plot frequencies and pdf.
- b Calculate expectation using Riemann summing of 1) relative freq.s, 2) pdf. Compare with parameter μ, and sample mean.
- c Do the same for variance.
- d Draw from a lognormal distribution ($Y = e^X$ for $X \sim \mathcal{N}$), plot and compare relative freq.s and pdf. Find the mode of the distribution.

Task A — Q 2A

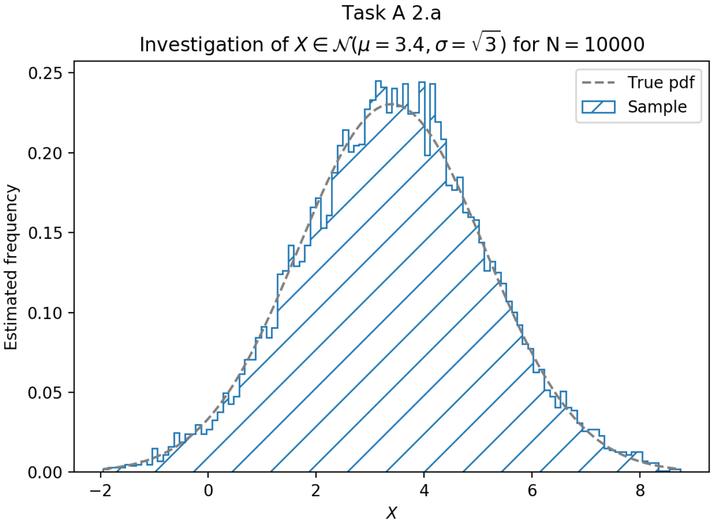
Draw 10000 samples from $\mathcal{N}(\mu = 3.4, \sigma^2 = 3.0)$. Plot frequencies and pdf.

Task A — Q 2A

$$\mathrm{pdf}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp - \frac{(x-\mu)^2}{2\sigma}$$

```
# Sample
import numpy.random.normal as sample_normal
Y_samp = sample_normal(10000)

# True
import scipy.stats.norm as norm
pdf = norm(loc=3.4, scale=np.sqrt(3))
bins_true = np.linspace(norm.ppf(0.001),
                        norm.ppf(0.999),
                        1000)
Y_true = norm.pdf(bins_true)
```

Task A — Q 2A



Task A 2.a
Investigation of $X \in \mathcal{N}(\mu = 3.4, \sigma = \sqrt{3})$ for N = 10000
True pdf
Sample
Estimated frequency
0.25
0.20
0.15
0.10
0.05
0.00
−2
0
2
4
6
8
X

Task A — Q 2B-C

Calculate, using the PDF above and a Riemann sum to numerically integrate, the expected value using the definition of expectation in Eq. (4.6) page 85. Compare against the sample mean value of the draws and against the true . Do they match?

Task A — Q 2B-C

Reminder, properties to calculate

$$E[X] = \sum x \ p(x), \ \ x \in X \tag{1}$$

$$Var[X] = \sum (x - E[X])^2 \ p(x), \ \ x \in X \tag{2}$$

Reminder, Riemann summation

$$\int_a^b dt\, f(t) \approx \sum_i f(t_i)(x_{i+1} - x_i) \tag{3}$$

```python
L = norm.ppf(.001)
R = norm.ppf(.999)

# Sample
bins_samp = np.linspace(L, R, int((R-L)/0.1))
h, b = np.histogram(Y_samp, bins=bins_samp,
                    density=True)
ex_samp = sum([h[i]*b[i]*0.1 for i in range(len(h))])

# True
dx = (R-L) / 1000
ex_true = np.asarray([Y_true[i]*bins_true[i]*dx
                      for i in range(n)]).cumsum()
```

Task A — Q 2B-C

$$Var(X) = E[(X-\mu)^2] = E[(X-E[X])^2] = E[X^2] - E[X]^2$$

```
mean_sq_samp = sum([bins_samp[i]**2*h_samp[i]*h
                    for i in range(n)])
sq_mean_samp = (sum([bins_samp[i]*h_samp[i]*h
                     for i in range(n)]))**2
var_samp = (mean_sq_samp - sq_mean_samp) * n/(n-1)
```

Task A — Q 2B-C

Using 10000 samples:

	Expected Value	Variance
Riemann data	3.3 (3.349)	3.0 (3.04)
Riemann pdf	3.4 (3.39)	3.0 (2.967)
Mean data	3.4 (3.428)	—

Task A — Q 2D

Show a lognormal distribution. Compare with pdf.

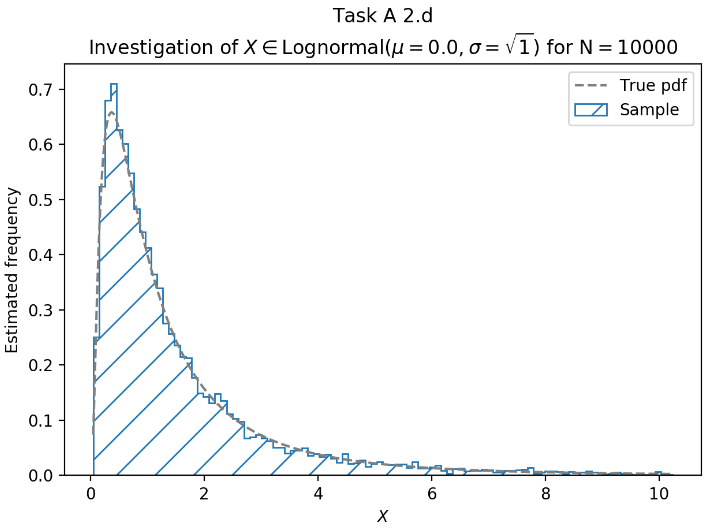
Task A — Q 2D

```
import numpy as np
import scipy as sp

Y_samp = np.random.normal(loc=0.0, scale=np.sqrt(1.),
Y_samp = np.exp(Y_samp)

norm = sp.stats.lognorm(s=np.sqrt(1),
                        scale=np.exp(0.0))
L = norm.ppf(0.001)
R = norm.ppf(0.999)
bins_pdf = np.linspace(L, R, 1000)
Y_true = norm.pdf(bins_pdf)
```

Task A — Q 2D

```
import matplotlib.pyplot as plt

plt.hist(data, histtype='step',
         bins=np.linspace(L, R,
                          int((R-L) / (0.1))),
         density=True, label='Sample')
plt.plot(bins_pdf, Y_true)
```

Task A — Q 2D


Task A 2.d
Investigation of $X \in \mathrm{Lognormal}(\mu = 0.0, \sigma = \sqrt{1})$ for N = 10000
True pdf
Sample
Estimated frequency
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
2
4
6
8
10
X

Task A — Q 2D

Find the mode of the lognormal distribution.

Task A — Q 2D

```
h_data, bins_data = np.histogram(
    data, density=True
    bins=np.linspace(L, R, int((R-L) / (0.1))))

mode_data = bins_data[np.argmax(h_data)]
mode_pdf = x_pdf[np.argmax(pdf)]
mode_pdf2 = scipy.optimize.fmin(lambda x: -norm.pdf(x)
```

Task A — Q 2D

Using 10000 samples:

	Estimated Mode
sample (argmax)	0.25 (0.2494)
pdf (argmax)	0.37 (0.3721)
pdf (Nelder-Mead)	0.37 (0.3679)