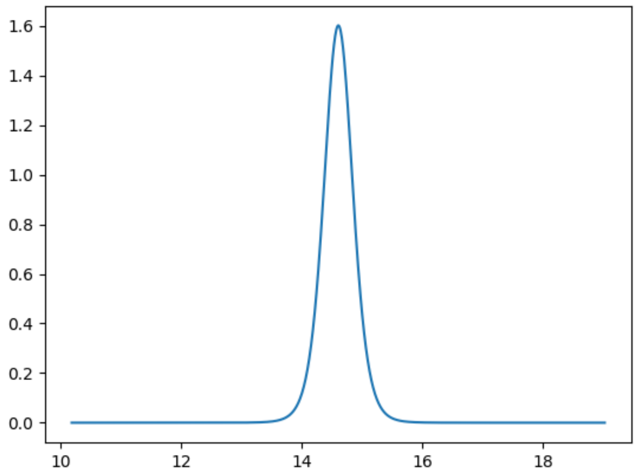
Bayesian Data Analysis - Assignment 3

September 30, 2018

1. *Inference for normal mean and deviation.*

   a. *What can you say about the unknown μ?*
   To answer to the questions it would be reasonable to plot the probability distribution function. It looks like this:

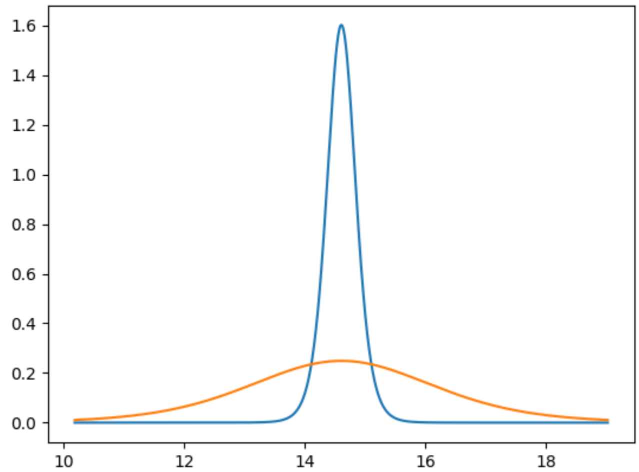
   

   From that we can say that μ is concentrated between **intervals [14.054, 15.168]** and has an **average hardness of 14.611** with estimated **variance 2.173** and estimated **standard deviation 1.474**.

   ```
   $ intervals: [14.054, 15.168]
   $ estimated mean: 14.611
   $ estimated variance: 2.173
   $ estimated standard deviation: 1.474
   ```

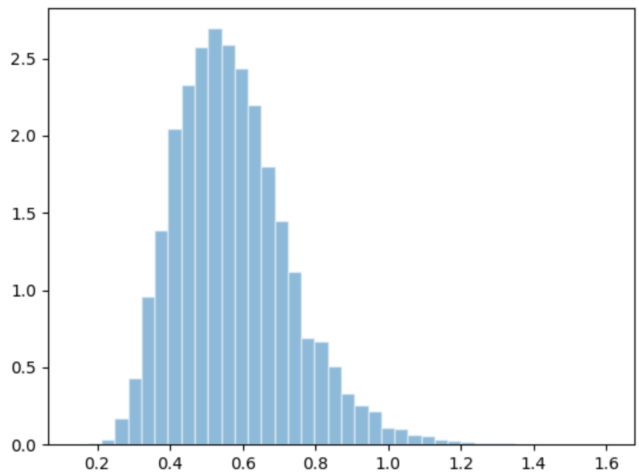
   Please see *Appendix A Source code for 1 "a" and "b"* for reference.

   b. *What can you say about the hardness of the next windshield coming from the production line before actually measuring the hardness?*
   We can say with the **95% confidence** that the hardness of the next windshield coming from the production line will be between **intervals [14.054, 15.168]**. Here is a plot:



2. *Inference for difference between proportions:*

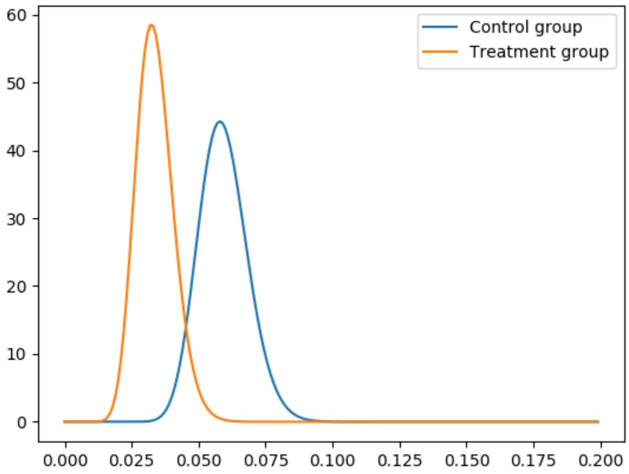
   a. *Compute the point and interval estimates and plot the histograms:*




```
Control central percentile 95%:  [0.0424, 0.078]
Treatment central percentile 95%:  [0.0217, 0.0488]
Control posterior mean:  0.0593
Treatment posterior mean:  0.0338
```

   Please see *Appendix B Source code for 2 "a" and "b"* for reference.

   b. *Discuss the sensitivity of your inference to your choice of prior density with a couple of sentences:*
   Based on the plotted graph that depicts the control group and treatment group, we can conclude that the treatment does not have a big positive impact.



Since we have chosen an uninformative prior, it does not impact on our inference.

Appendix A Source code for 1 ”a” and ”b”

```
import matplotlib
matplotlib.use('TkAgg')
from math import sqrt
from scipy import stats
import matplotlib.pyplot as plt
import numpy as np

data = [
    13.357,
    14.928,
    14.896,
    15.297,
    14.82,
    12.067,
    14.824,
    13.865,
    17.447,
]
n = len(data)
estimated_mean = np.mean(data)
estimated_variance = stats.tvar(data)
x_range = np.arange(
    estimated_mean - 3 * sqrt(estimated_variance),
    estimated_mean + 3 * sqrt(estimated_variance),
    0.01
)

'''
a) What can you say about the unknown $\hat{I}_4^1$?
'''
```

```python
y_range = stats.t.pdf(
    x=x_range,
    df=n-1,
    loc=estimated_mean,
    scale=estimated_variance/n
)
intervals = stats.t.interval(
    0.95,
    df=n-1,
    loc=estimated_mean,
    scale=estimated_variance/n
)
low, up = intervals
print('a)')
print('-- intervals:', [round(low, 3), round(up, 3)])
print('-- estimated mean:', round(estimated_mean, 3))
print('-- estimated variance:', round(estimated_variance, 3))
print('-- estimated standard deviation:', round(sqrt(estimated_variance), 3))
figure = plt.plot(x_range, y_range)
plt.savefig('./ex3/report/3_1_a_mean_student_distribution.png')

'''
b)
'''
std_y = np.std(data, ddof=1)
scale = sqrt(1 + 1/n) * std_y
y_posterior_range = stats.t.pdf(
    x=x_range,
    df=n-1,
    loc=estimated_mean,
    scale=scale
)
figure = plt.plot(x_range, y_posterior_range)
plt.savefig('./ex3/report/3_1_b_posterior_mean.png')
```

Appendix B Source code for 2 ”a” and ”b”

```python
import matplotlib
matplotlib.use('TkAgg')
from scipy import stats
import matplotlib.pyplot as plt
import numpy as np

x_range = np.arange(0, 0.2, 0.001)

'''
a)
'''
control = 674
control_died = 39
```

```python
control_alpha = control_died + 1
control_beta = control - control_alpha + 1
control_posterior = control_alpha/control
control_pdf = stats.beta.pdf(x_range, control_alpha, control_beta)
control_pdf_line = plt.plot(x_range, control_pdf)

treatment = 680
treatment_died = 22
treatment_alpha = treatment_died + 1
treatment_beta = treatment - treatment_alpha + 1
treatment_posterior = treatment_alpha/treatment
treatment_pdf = stats.beta.pdf(x_range, treatment_alpha, treatment_beta)
treatment_pdf_line = plt.plot(x_range, treatment_pdf)

plt.legend(
    [*control_pdf_line, *treatment_pdf_line],
    ['Control group', 'Treatment group']
)
plt.savefig('./ex3/report/3_2_a_control_beta.png')
plt.figure(0)

'''
b)
'''
p_control = stats.beta.rvs(control_alpha, control_beta, size=10000)
p_treatment = stats.beta.rvs(treatment_alpha, treatment_beta, size=10000)
odd_ratio = (p_treatment/(1-p_treatment))/(p_control/(1-p_control))

plt.hist(odd_ratio, density=True, alpha=0.5, bins=40, ec='white')
plt.savefig('./ex3/report/3_2_b_histog.png')
plt.figure(0)

'''
intervals
'''
control_percentile_25 = np.percentile(p_control, q=2.5)
control_percentile_95 = np.percentile(p_control, q=97.5)

treatment_percentile_25 = np.percentile(p_treatment, q=2.5)
treatment_percentile_95 = np.percentile(p_treatment, q=97.5)
print('-----------')
print(
    'Control central percentile 95%: ',
    [round(control_percentile_25, 4), round(control_percentile_95, 4)]
)
print(
    'Treatment central percentile 95%: ',
    [round(treatment_percentile_25, 4), round(treatment_percentile_95, 4)]
)
print('Control posterior mean: ', round(np.mean(control_posterior), 4))
print('Treatment posterior mean: ', round(np.mean(treatment_posterior), 4))
```