12-778 Fall 2022: Assignment #3

Mario Bergés

Instructions

These solutions could be wrong, so tread with care!

Measurement Error and Uncertainty (100%)

And these are from lectures 17 and 18, along with their lecture notes and references.

Uncertainty Estimation (30%):

The density of air (ρ) can be estimated by measuring the absolute air pressure (p) and the aboslute temperature (T):

$$\rho = \frac{p}{RT}$$

where $R = 287.04$ J/(kg K) is the specific gas constant for dry air, ρ is in units of kg/m^3, T is in Kelvin and p is in mm Hg.

Assuming that the accuracy of the pressure and temperature measurements are 1 mm Hg and 1 K respectively, what is the uncertainty of the air density estimation when $T = 24$ Celsius and $p = 760$ mm Hg?

> 💡 Answer
>
> Our function, through which we will propagate the uncertainty in our measurements, is as follows:
>
> $$\rho(p, T) = \frac{p}{RT}$$
>
> We know that $u_p = 1$ mmHg, $u_T = 1$ Kelvin and R is a constant with a value of 287.04 J/(kg K).

Assuming that p and T are uncorrelated, the uncertainty of ρ is thus:

$$u_\rho = \sqrt{\left(\frac{\partial}{\partial p}\rho(p,T)u_p\right)^2 + \left(\frac{\partial}{\partial T}\rho(p,T)u_T\right)^2}$$

Substituting the values and finding the partial derivatives we have:

$$u_\rho = \sqrt{\left(\frac{1}{RT}u_p\right)^2 + \left(\frac{p}{RT^2}u_T\right)^2}$$

Which we can compute as:

```python
import numpy as np

u_p = 1 # mmHg
u_t = 1 # Kelvin
T = 24 + 273.15 # Kelvin
p = 760 # mmHg
R = 287.04 # J/(kg K)

u_rho = np.sqrt((1/(R*T)*u_p)**2+(p/(R*T**2)*u_t)**2)
print(f'The uncertainty of the air density estimation is {u_rho:.4f} mmHg kg / J')
```

```
The uncertainty of the air density estimation is 0.0000 mmHg kg / J
```

Total uncertainty (30%):

The supply air pressure in a duct is going to be kept constant at a set value for a series of tests. A damper controls the air flow (and pressure) supplied by the air handling unit. The duct is equiped with a dial gauge (resoluton of 1 psi and accuracy of 0.5 psi) to monitor the air pressure. Thirty trials of maintaining a static pressure of 50 psi are attempted to estimate how precisely the pressure can be controlled via the damper. The results show that the standard deviation in the set pressure is 2 psi. Estimate the total uncertainty at 95% confidence in the set pressure that would be expected during normal operation.

Note: since we do not know the population standard deviation, you will need to use Studennt's t-distribution.

> **💡 Answer**
>
> We have a sample standard deviation of $s_p = 2$ psi. Given that this was with 30 trials (i.e., $\nu = 29$ degrees of freedom), and we want the 95% confidence interval, which means we want $t_{95\%,29}$, which can be gleaned from a table (or through any other means) and is 2.045. This is enough for us to calculate the uncertainty of the measurements (not the total uncertainty) as $u_p = t_{95\%,29} \frac{s_p}{\sqrt{30}}$. Then we need to add half of the resolution error ($u_0 = 1/2$ psi) as the design stage uncertainty, and the accuracy $u_1 = 0.5$ psi using the RSS method, namely: total uncertainty $= \sqrt{u_0^2 + u_1^2 + u_p^2}$. To calculate we do:
>
> ```python
> import numpy as np
>
> s_p = 2 # psi
> t_95 = 2.045
> u_p = s_p/np.sqrt(30) * t_95
> u_1 = 0.5 # psi
> u_0 = 0.5 # psi
>
> u_total = np.sqrt(u_0**2 + u_1**2 + u_p**2)
>
> print(f'The total uncertainty is {u_total:.2f} psi')
> ```
>
> ```
> The total uncertainty is 1.03 psi
> ```

Multiple-Measurement Uncertainty Analysis (40%):

The stress on a loaded wing for an electric drone is measured using a system consisting of a strain gague, a Wheatstone bridge, an amplifier, and a data acquisition system. The bias and precision uncertainties arising from calibration, data acquisition and data reduction are listed below:

Error source	Bias uncertainty (N/cm^2)	Precision uncertainty (N/cm^2)	# of Samples
Calibration	1.0	4.6	15
Data acquisition	2.1	10.3	38
Data reduction	0.0	1.2	9

Assume 100% reliability in the values of all bias errors and that there are no correlated uncertainties.

For 95% confidence, determine the range that contains the true mean value of the stress given that the average value is 223.4 N/cm^2.

> ### 💡 Answer
>
> The "expanded uncertainty" (u_σ) that we are calculating here is the result of finding a t value for the combined degrees of freedom at 95% confidence and then multiplying that by the random standard uncertainty (combining precision errors). Finally we combine this – in the RSS sense – with the total systematic uncertainty (combining all bias errors). Thus:
>
> $$u_\sigma = \sqrt{b_{\bar{\sigma}}^2 + (t_{\nu_{combined},95\%} s_{\bar{\sigma}})^2}$$
>
> Where
>
> $$b_{\bar{\sigma}} = \sqrt{b_c^2 + b_{daq}^2 + b_{dr}^2}$$
>
> $$s_{\bar{\sigma}} = \sqrt{s_c^2 + s_{daq}^2 + s_{dr}^2}$$
>
> and
>
> $$\nu_{combined} = \frac{\left(s_c^2 + s_{daq}^2 + s_{dr}^2\right)^2}{\left(\frac{s_c^4}{14} + \frac{s_{daq}^4}{37} + \frac{s_{dr}^4}{8}\right)}$$
>
> and `c` indicates "calibration", `daq` indicates "data acqusition" and `dr` is "data reduction". We can now combine these to find the result:

```
from scipy.stats import t
q = 0.975 # Quantile

df_c = 14  # Degrees of freedom for calibration statistics
df_daq = 37 # Degrees of freedom for data acquisition stats
df_dr = 8 # Degrees of freedom for data reduction stats

b_c = 1.0
s_c = 4.6
b_daq = 2.1
s_daq = 10.3
b_dr = 0.
s_dr = 1.2

df_combined = (s_c**2 + s_daq**2 + s_dr**2)**2/(s_c**4/df_c + s_daq**4/df_daq + s_dr**4/

t_combined = t.ppf(q, df_combined)

u_total = np.sqrt(b_c**2 + b_daq**2 + b_dr**2 + (t_combined * np.sqrt(s_c**2 + s_daq**2
average = 223.4

low_limit = average - u_total
high_limit = average + u_total

print(f'A 95% confidence interval containing the true mean value of the stress is [{low_
```

A 95% confidence interval containing the true mean value of the stress is [200.487, 246.313

Let's compare that to the naive approach where we treat each uncertainty separately and
later combine them.
The calibration uncertainty is:

$$u_c = \sqrt{b_c^2 + (t_{\nu_c, 95\%} s_c)^2}$$

Similarly, the data acquisition uncertainty is:

$$u_{daq} = \sqrt{b_{daq}^2 + (t_{\nu_{dad}, 95\%} s_{daq})^2}$$

And the data reduction uncertainty, is similarly defined:

$$u_{dr} = \sqrt{b_{dr}^2 + (t_{\nu_{dr}, 95\%} s_{dr})^2}$$

```python
t_c = t.ppf(q, df_c)
t_daq = t.ppf(q, df_daq)
t_dr = t.ppf(q, df_dr)

u_c = np.sqrt(b_c**2 + (t_c*s_c)**2)
u_daq = np.sqrt(b_daq**2 + (t_daq*s_daq)**2)
u_dr = np.sqrt(b_dr**2 + (t_dr*s_dr)**2)

u_total = np.sqrt(u_c**2 + u_daq**2 + u_dr**2)

average = 223.4
low_limit = average - u_total
high_limit = average + u_total

print(f'A 95% confidence interval containing the true mean value of the stress is [{low_
```

A 95% confidence interval containing the true mean value of the stress is [200.034, 246.766