# MTH 3240 Lab 5

Due Thu., Feb. 27

## 1 Part A: Two-Sample *t* Test and Confidence Interval

### 1.1 Contaminants in Bird's Eggs Data Set

A study was carried out to assess the risk to reproductive success of herons and egrets due to contaminants in their nesting areas. An egg was taken from each of **nine** randomly selected **Little Egret** nests in the Mai Po Marshes Nature Reserve in Hong Kong and from each of **nine** randomly selected **Black-Crowned Heron** nests in the A Chau egretry.

In each egg, the **PCBs** (organochlorine compounds) were measured. The data are below.

| Little Egret PCBs | Black-Crowned Heron PCBs |
|---|---|
| 1700, 1000, 800, 970, 1600, 1000, 270, 370, 970 | 530, 140, 110, 600, 160, 85, 170, 150, 120 |

We want to decide whether the **PCB** concentrations **differ** for the two species of birds' eggs.

1. Use `c()` to create two data vectors, one containing the **Little Egret PCBs** and the other the **Black-Crowned Heron PCBs**.

2. Recall that `boxplot()` will produce a boxplot of a data set. The main argument passed to `boxplot()` is a data vector `x`, but it accepts other optional arguments too. Among its arguments are:

   | | |
   |---|---|
   | `x` | a data vector. |
   | `col` | a color used to fill the body of the box. |
   | `xlab` | a label for the x-axis. |
   | `ylab` | a label for the y-axis. |
   | `main` | a main title. |
   | `names` | a character vector of group names for the x-axis. |

   It turns out that `boxplot()` will accept any number of vectors as arguments and plot them as ***side-by-side boxplots*** (e.g. `boxplot(x, y, z)` plots the data vectors `x`, `y`, and `z`).

   For example, if you named your **PCBs** vectors `le` and `bch`, then the following produces the boxplots:

   ```
   boxplot(le, bch, col = "lightblue", names = c("Egrets", "Herons"),
           main = "Boxplots of PCBs")
   ```

   Use `boxplot()` to make ***side-by-side boxplots*** of the **PCB** concentrations.

3. We want to decide if the **PCB** concentrations **differ** for the two species of birds' eggs.

   The `t.test()` function, when passed **two** vectors, will carry out a ***two-sample t test*** (and compute a ***95% two-sample t confidence interval***) for two population means $\boldsymbol{\mu_x}$ and $\boldsymbol{\mu_y}$. Among its arguments are:

   | | |
   |---|---|
   | `x` | a data vector. |
   | `y` | another data vector. |
   | `alternative` | the direction for the alternative hypothesis, one of `"two.sided"`, `"less"`, or `"greater"`. |
   | `mu` | the null hypothesized value for the unknown difference between population means, with default value 0. |
   | `conf.level` | the confidence level for a confidence interval for the unknown difference between population means, with default value 0.95. |

   Use `t.test()` to carry out a ***two-sample t test*** of

   $$\begin{aligned} H_0 : \mu_x - \mu_y &= 0 \\ H_a : \mu_x - \mu_y &\neq 0 \end{aligned}$$

   where $\boldsymbol{\mu_x}$ is the true (unknown) mean **PCB** concentration for **Little Egret** eggs and $\boldsymbol{\mu_y}$ is the true mean for**Black-Crowned Heron** eggs. For example, if your data vectors are called `le` and `bch`, then you'd type:

```
t.test(x = le, y = bch, mu = 0, alternative = "two.sided")
```

4. The function `t.test()` also reports a ***95% confidence interval*** for the true (unknown) difference $\boldsymbol{\mu_x - \mu_y}$, or ***effect size***. Find the endpoints of the ***95% confidence interval*** for the ***effect size***.