EGB103 Assignment 3
Data Integration and Solving Engineering Problems

Individual only (strictly no group work or collaboration allowed)

Due: Friday 10th June (Week 13)

Worth: 50%

Part A (worth 40%)

In assignment 2 we studied one of the five chillers in P Block Gardens Point and searched for patterns in the performance of that chiller. In this assignment we will study the two chillers in Q Block at Kelvin Grove campus with the goal of reducing the electrical energy consumed.

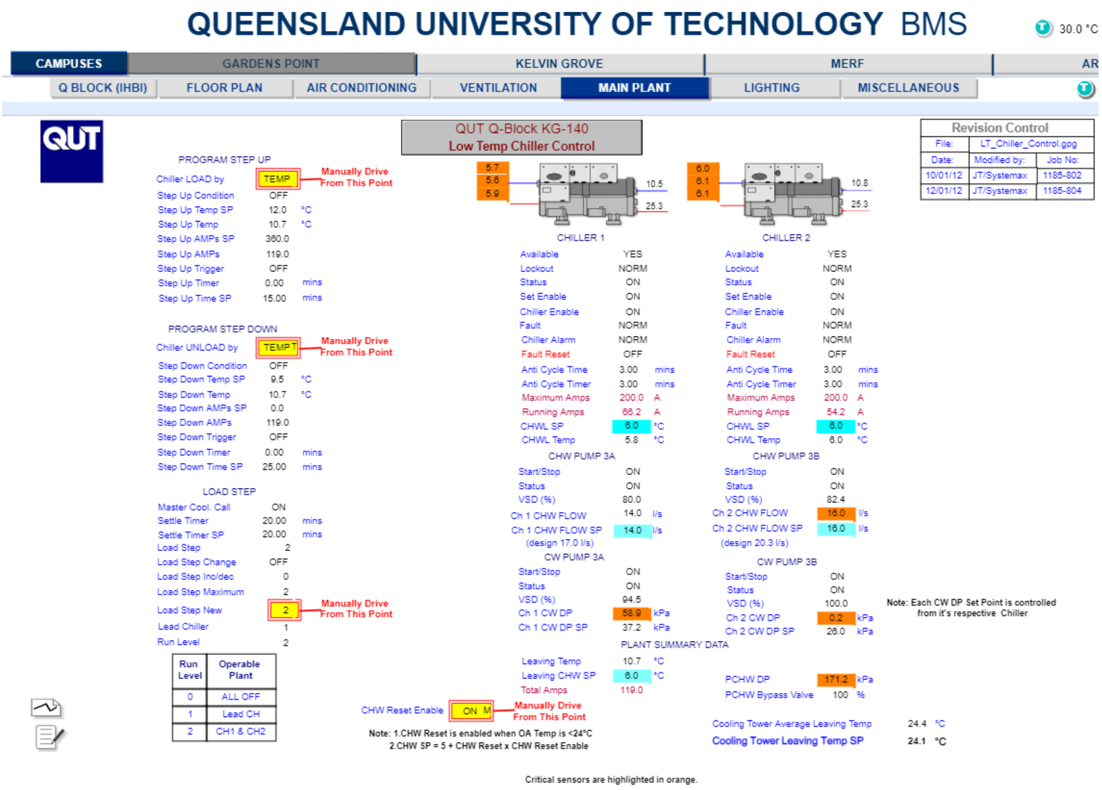

Figure 1: BMS for Q Block Chillers at Kelvin Grove Campus

The data that we will study is again time-series data that includes for Chiller 1 and Chiller 2, a historical record of the following data at 5-minute intervals:

- The chilled water flow rate (L/S)
- The chilled water entering temperature (°C)
- The chilled water leaving temperature (°C)
- The electrical energy consumed by the chiller (Kilowatts), abbreviated kWE.

Task A - Computing Total Electrical Energy Consumed

The kWE tells us the *instantaneous* energy used at a point in time. To compute the total electricity consumed (measured in kilowatt-hours), we need to multiply this instantaneous energy use by the duration for which it is sustained. Since the data readings are made at 5-minute intervals, we will *assume* that this instantaneous energy remains constant for that 5-minute period. We can then sum over these 5-minute periods to compute the total electrical energy consumed over the one-month period that the data covers. You should compute the result (displayed in megawatt-hours) separately for each chiller, as well as displaying the total. This tells us approximately how much electricity was actually consumed during that period and will provide a baseline for comparison when trying to improve the efficiency of the system.

Task B – Deriving Performance Curves for each Chiller

Let us assume that we know the chilled water flow rate to a particular chiller, and we know the entering and leaving temperatures. From that data we can compute the refrigeration load (kWR). How could we then predict the *expected* instantaneous energy use (kWE)?

> *Models allow us to make predictions, but they require us to make certain assumptions. These assumptions may not be perfectly true, but provided they are approximately correct, we can potentially make reasonable predictions. Ideally, we are able to assess whether these assumptions lead to reasonable predictions by applying our model to situations from the past where we know what actually happened. In that way we are able to compare actual and predicted to give us confidence in applying our model to other situations.*

If we knew the coefficient of performance (COP), then we could estimate the kWE. Unfortunately, the coefficient of performance is not constant and depends on a number of factors, including the current refrigeration load and the model of chiller being used. Since our two Q block chillers are different models, we start by analysing for each chiller, how its coefficient of performance varies with the refrigeration load (hopefully as you already did in assignment 2 for one of the P block chillers). As the refrigeration load is a continuous quantity, we will start by dividing the range of refrigeration loads experienced during the data collection period into fixed sized bins.

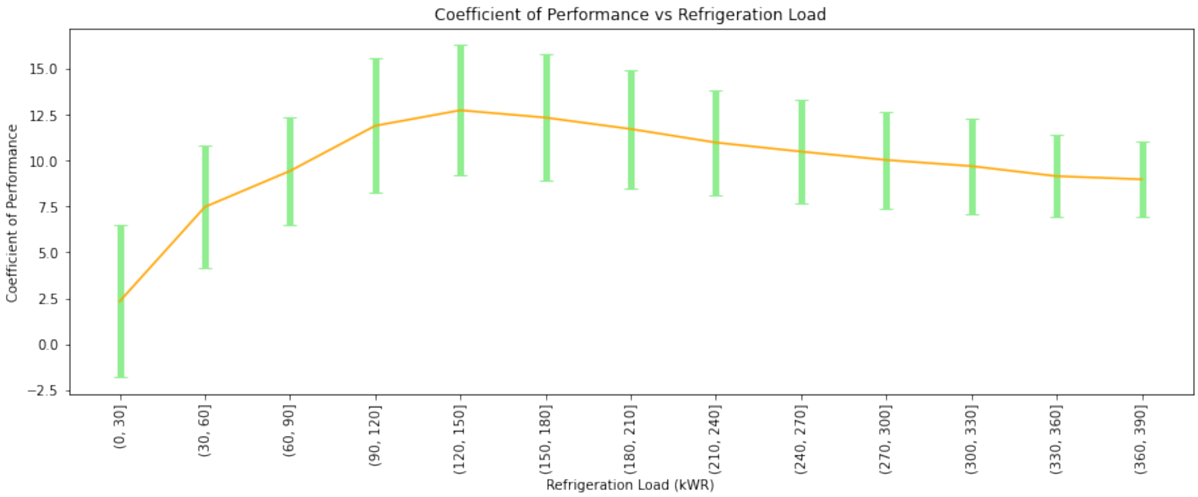

Figure 3: Example Performance Curve

In the example illustrated in Figure 3, the bins are 0 to 30KW, 30 to 60KW, 60 to 90KW, etc. Note: the bin size needs to be large enough so that each range contains at least one data point. For each of these bins, we compute the mean COP (using a pandas group by query). As indicated by the green standard deviation bars, there is considerable variation within each bin, but a clear trend is visible, with a refrigeration load of 120 to 150kW yielding the best coefficient of performance (on average) for that particular example chiller. The two chillers in Q block are slightly different models, so their performance curves will differ. Include a plot that shows the performance curves for both Q block chillers.

Task C - Predicting kWE for each chiller

We are going to use their performance curves to help predict the kWE that would be required to achieve a given load (kWR). For example, if chilled water enters the above chiller at 12.5 °C and leaves at 6.9 °C and the flow rate is 23 L/s then the kWR will be approximately 211kW which belongs to the bin (210,240] which has a mean COP of approximately 11. We can then estimate the Electrical Energy consumed (kWE) based on this kWR and predicted COP. Again, note that these calculations depend on the performance curve that differs from one chiller to another. You should therefore create two separate Python functions to estimate kWE, one for each of the Q block chillers. Both functions will take a flow rate and a temperature difference and will return the estimated kWE. These two functions will use the performance curve data for their respective Q block chillers. This performance curve data should be pre-computed (via a function) and stored in a variable for each chiller (rather than recomputing the performance curves ever time one of these estimate kWE functions is called).

Task D - Testing the accuracy of our model predictions

We can now use the above functions to compute a new column in the data frames for each chiller, that contains an estimate of the kWE at each point in time based on the actual flow rate and temperatues recorded at that time point. We can then compute the *total estimated* electrical energy consumed over the one-month period (measured in megawatt-hours). In that way we can compare the *actual* megawatt-hours for the month to the *estimated* megawatt-hours for each chiller. Are they fairly similar? Does our model seem accurate enough to make reasonable predictions?

Understanding the Common Return and Supply Data

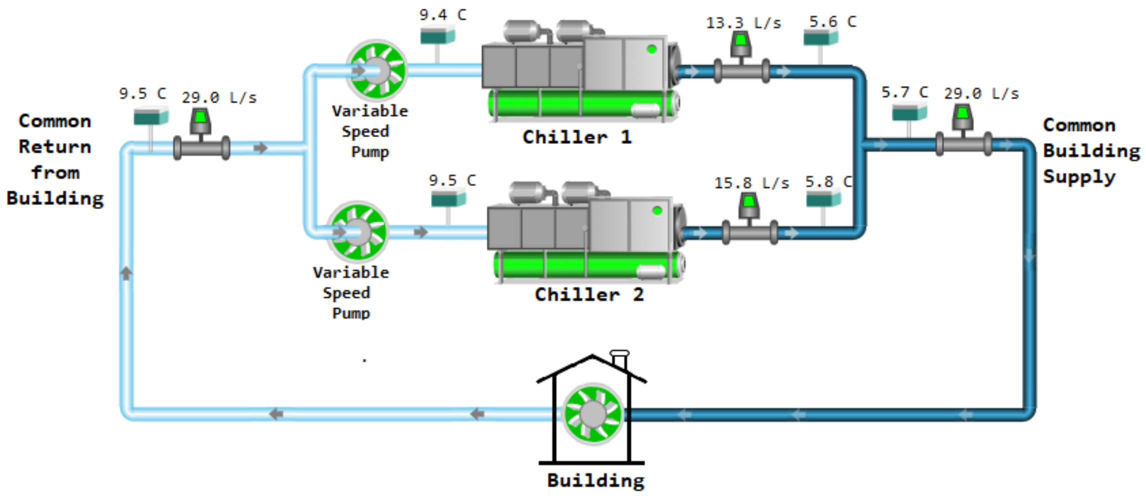

Figure 2: Chilled Water System Schematic

As shown in Figure 2, each chiller has its own variable speed pump and a flow meter. Each chiller also has temperature sensors to measure the incoming and outgoing chilled water temperatures. However, if we look at the system as a whole, we see that there is a *common return point* where the chilled water returns from the building before being split between the two chillers. There is also a *common building supply point* where the chilled water from the two chillers is recombined before being sent to the rest of the Q block building. These *common points* also have *flow* meters and *temperature* sensors. We see that a total of 29.0 L/S is currently passing through the chillers; 13.3 L/S is currently being sent to chiller 1, while 15.8 L/S is currently being sent to chiller 2. The chilled water is currently returning from the building at 9.5°C. This common return temperature will be approximately the same as the incoming temperature as sensed by each chiller, however, it may vary very slightly due to either imprecise temperature sensors or small actual changes to water temperature as it travels through the pipes (due to either convection or friction). The common supply temperature will be approximately the average temperature of the outgoing chilled water from each chiller (weighted by their respective flow rates). *We have provided data from these common sensors in a separate data file*. From this common data we could compute at each time point, the total refrigeration (kWR) achieved by the system as a whole. This total refrigeration (kWR) reflects the cooling demand from the building at that time, i.e., if it is hot outside or there are lots of people in the building, then the total refrigeration required will tend to be higher. So, this common data provides a historical record of how the cooling demand of the building varied over that month.

Understanding the Stepping Algorithm

As the Q block system is made up of two chillers, whenever the refrigeration load changes, the system needs to decide which of the chillers to use at that point in time. We see in Figure 1 that the system is currently operating at *run level 2*, which means that both chillers are being used. We can also see that at *run level 1*, only one of the chillers (the "*lead*" chiller) will operate. The system is controlled by an *algorithm* that automatically *steps up or down* between *run levels* based on different sensors and thresholds (as shown on the left of Figure 1). The actual algorithm is quite complex and considers many factors that we will ignore (for example, in practice it is not good to turn chillers on and off again too rapidly, but for simplicity, our model will ignore some of these constraints). You do not need to understand how the actual algorithm works; the data in Figure 1 is provided only for your interest.

Task E - Modelling alternative stepping algorithms

We are now going to use the common sensor data file to create a *simulation/model* that *replays* that month (minute by minute) and considers what effect using an alternative stepping algorithm may have had. For our alternative stepping algorithm, we are again going to make certain assumptions. We know the common return temperature, and we will *assume* that will also be the temperature that the chilled water will reach each of the chillers. We also know the common supply temperature. If we are using both chillers, then we will *assume* that both chillers will achieve that common supply temperature. That may not precisely happen in practice, but we know that the system is configured so that both chillers have the same temperature *set point.* The set point is the temperature that each chiller is striving to achieve. We also know at each point in time, the total flow (*t*) of chilled water in L/S. The only question is, how is this flow divided between the two chillers? This is therefore the decision that we will seek to optimize. We have three options:

1. Use only chiller 1 to handle all *t* L/S of flow (and switch off chiller 2)
2. Use only chiller 2 to handle all *t* L/S of flow (and switch off chiller 1)
3. Use both chillers, with *a* L/S sent to chiller 1 and *b* L/S sent to chiller 2 (where $a + b = t$)

Our algorithm will choose the option that produces the lowest estimated kWE. The kWE for options 1 and 2 can be computed using the functions that we created above to estimate kWE for each chiller. For option 3, if we knew *a* and *b* then we could again use the functions created above for each chiller and simply sum the electricity consumed by each chiller. To find the optimal values of *a* and *b*, we will use a loop to consider all possible values of *a* (from *0* to *t* in *1* litre increments) and will use the value that leads to the lowest overall electricity consumed by both chillers. Create a Python function that implements this algorithm.

Task F - Capacity Constraints

Each chiller has a *minimum* and *maximum* flow rate. The maximum flow rate is governed by the physical limits of the chiller pump. If a chiller tries to operate with a flow below its minimum flow rate, then the chiller may overheat and be damaged, so if a chiller is operating, its flow rate must never be below the minimum flow rate. Each chiller also has a maximum capacity (kWR). It is not possible for the chiller to achieve cooling above that maximum capacity. We need to consider these capacity constraints when estimating kWE for a given required flow rate and temperature difference. If the required flow rate is less than the minimum flow rate, then we simply assume that the minimum flow rate will be delivered instead. If the required flow rate or required refrigeration is beyond the chillers capacity, you should simply return the estimated kWE as infinity (∞) to indicate that the cooling load cannot be achieved by that chiller alone. Both chillers have a minimum flow of 12.0 L/S, a maximum flow of 23.0 L/S and a maximum capacity of 600 kWR. These capacity constraints should be incorporated into the functions that you use for estimating the kWE for each chiller.

As always, be careful not to repeat yourself – so create extra helper functions as required to avoid code duplication.

Task G - Evaluating our alternative stepping algorithm

For each row in the common sensor data frame, you should compute a column that contains the total estimated kWE based on our alternative algorithm (developed in Task E). From that you can then compute the total electrical consumption in megawatt-hours over that one-month period. How does it compare to the total electricity that was actually consumed in that month? Would we have achieved any cost savings? Why might our alternative stepping algorithm not be feasible in practice?

Submission Requirements

Everything for Part A should be included in a single Jupyter notebook. The Python code included should follow best practices as outlined in the lectures, including using well chosen identifier names, writing clear simple code, and not repeating yourself.

All data processing should be done using the Pandas library.

All of these features are covered in the lectures and/or practical exercises, so please do not use Python or Pandas features outside of what has been covered in class. Note, ***you will get Zero marks*** if you use some other programming language, library or system such as R, MATLAB or Excel.

Note: the following section is separate and does not relate to the Chiller data processing or Pandas.

Part B (worth 10%)

Task H - A case study of computing and data in action

This part of the assessment task gives you a chance to explore and present an example of computing and data being put to work. In high-level terms, we want you to find and investigate a situation where computing and data are being used to solve problems or design solutions, then describe the underlying algorithm or computational method to us as a short (1-2 page) case study, using the elements of "motivation", "materials" and "methods" like case studies presented in class.

You are free to choose whatever situation interests you, but we expect you to treat the topic seriously with an engineering mindset. You could explore an algorithm that has been around for centuries, or a method that is at the current frontier of knowledge, or anything in between. This is a chance for you to explore something that is new and interesting to you, maybe something that could help you decide on your engineering major or motivate your future career?

Your response must conform to the template provided for part B. To guide you, we have used blue text to describe what we expect you to put in each section or element of the case study. You must delete or replace all that blue text with your own words in your submission.

We also provide an annotated exemplar case study to demonstrate the quality and content of work that would be awarded full marks.

For Part B, you should use the provided Juypter Notebook template file, however for submission you should convert your Juypter Notebook into a PDF file using `Save and Export Notebook As` PDF from the File Menu.

On Blackboard you will find the following attachments related to Part B:

1. `EGB103Assignment3PartB.ipynb` *(a template for you to use)*
2. `ExampleCaseStudyPartB.pdf` *(an example of what is expected)*
3. A video explaining Part B

What to Submit

Please submit precisely two files named as follows:

1. `EGB103Assignment3PartA.ipynb` *(do **not** export as PDF)*
2. `EGB103Assignment3PartB.pdf` *(generated from EGB103Assignment3PartB.ipynb)*

If you have been granted an extension **– please do not attach any extra files** – we already know precisely who has been granted an extension!