**Programme de détermination de l’occupation des bureaux.**

Pour trouver les horaires d’occupation des bureaux, nous avons choisi d’étudier la luminosité de ceux-ci car cette donnée nous semblait plus pertinente que les autres. En effet, le bruit, la température, le taux d’humidité et le CO2 ne dépendent pas uniquement de la présence de personnes dans les bureaux alors que la lumière est nécessaire au travail.

Pour comparer la luminosité à différents moments de la journée, nous devions trouver un seuil permettant de trancher sur la possibilité de travailler ou non avec cette quantité de lumière. D’après différents sites internet, la luminosité dans les lieux de travail est généralement supérieure à 150 lux. Nous avons donc pris cette valeur comme référence : lorsque la luminosité est supérieure à cette valeur, nous considérons que les bureaux sont occupés.

Nous avons écrit dans un premier temps un programme qui renvoie la liste des horaires pour lesquels la luminosité est supérieure à ce seuil de 150 lux. Cette liste correspond donc aux horaires d’occupation des bureaux pour un capteur.

Cette valeur renvoie des listes cohérentes pour les capteurs 1 et 3. Cependant, on remarque que pour les autres capteurs (2,4,5 et 6), la luminosité ne diminue pas les deux dernières nuits. Elles sont donc comptabilisées par notre programme comme des horaires de travail. Cette incohérence peut être due à un oubli d’éteindre la lumière deux nuits de suite, ou à un mauvais fonctionnement des capteurs. Cette deuxième option nous semble plus probable car les valeurs de luminosité pour ces mêmes capteurs lors des deux derniers jours sont beaucoup plus importantes que celles au début des mesures (par exemple le 14 août à 10h30 la luminosité était de 272 lux alors qu’elle est de 728 lux le 25 août à la même heure).

*(ensuite je voulais regrouper les listes obtenues pour les différents capteurs et les comparer pour voir si on obtenait à peu près les mêmes horaires mais je sais pas du tout comment coder ça)*

**Pour le point 3 :** Pour la liste des temps, un « +0200\n » était présent après chaque date. Nous n’avions pas besoin de ces caractères pour notre étude donc nous les avons retirés lors de la création des listes comportant les dates et heures.

**4. Valeurs statistiques.**

Nous avons tout d’abord écrit de façon classique les fonctions calculant les différentes valeurs statistiques d’une liste (maximum, minimum, écart-type, moyenne, variance et médiane). Pour le calcul de la moyenne, nous avons décidé de choisir la moyenne arithmétique classique car elle nous semblait être la plus parlante et la plus pertinente. Pour le calcul de la médiane, nous avons dû au préalable trier la liste par ordre croissant. Nous avons utilisé pour cela un algorithme de tri par insertion car cet algorithme est efficace sur des listes de la taille de celles traitées dans notre projet.

Nous avons ensuite écrit des programmes renvoyant chacun une courbe ou un point légendé(e) correspondant à la valeur statistique associée.

**5. Calcul du coefficient humidex**

Nous avons trouvé après des recherches les formules suivantes pour calculer l’indice humidex :

Où T correspond à la température mesurée et Trosée le point de rosée, calculé comme ceci :

avec où correspond à l’humidité relative, T à la température, a=17,27 et b=237,7.

Nous avons donc transcrit ses formules sur Python pour accéder à l’indice humidex d’un couple humidité relative/température.

**9. Calcul du coefficient de corrélation**

Nous avons trouvé que l’indice de corrélation pouvait être calculé de la façon suivante :

où et correspondent aux écarts-types et cov(X,Y) correspond à la covariance de X et de Y. Cette dernière se calcule de la façon suivante :

avec N le nombre de valeurs des listes X et Y (supposées de même longueur), et et les moyennes des listes X et Y.

Nous avons donc encore une fois écrit ces formules sur Python afin d’accéder au coefficient de corrélation de deux listes.