PROJET PYTHON

Objectifs du projet :

- Créer un simulateur de taux d’injections de glycémie avec interface graphique facile à utiliser, basé sur des règles de calculs simples (règles usuelles données aux diabétiques de type 1).

- Vérifier la validité du modèle en comparant sur des données pratiques si les injections simulées sont bien celles qui ont été réalisées en réalité.

- Chercher à améliorer le modèle en faisant varier les paramètres du modèle (bornes dans les règles de calcul, meilleure prise en compte de l’influence du sport ou d’un repas riche) pour avoir des taux d’injections adaptées à l’utilisateur.

1ère étape: coder fonctions élémentaires de base :

à partir de :

|  |  |  |  |
| --- | --- | --- | --- |
|  | Matin | Midi | Soir |
| Jour 1 | Glycémie 1 Injections R/L | Glycémie 2  Injections R | Glycémie 3  Injections R/L |
| Jour 2 | Glycémie 1 Injections R/L | Glycémie 2  Injections R | Glycémie 3  Injections R/L |

Pour rappel :

La glycémie est prise avant chaque repas. Si elle est bonne (comprise entre 50 et 180), et si celle de la veille est bonne, on injecte de l’insuline rapide (action sur 3 heures) et de l’insule lente (action sur 12 heures) aux même taux que la veille pour le même repas. Pas besoin d’injection d’insuline lente à midi.

On modifie l’injection correspondant au repas précédent si la glycémie est mauvaise. Par exemple, hyperglycémie à midi 🡪 modification de l’injection du matin le jour suivant.

Si la glycémie est mauvaise, selon qu’elle est trop basse ou trop élevée et selon la glycémie de la veille est trop élevée ou pas, on modifie les taux selon les règles suivantes :

Comment on modifie les taux d’injections d’insuline en fonction de la glycémie ?

1er cas : hypoglycémie (glycémie<50) : MODIFICATION DU JOUR AU LENDEMAIN (par exemple, si le jour 2 avant le déjeuner, glycémie2<50, alors le jour suivant on applique la modification):

Baisse du taux d’injection du précédent repas (par exemple si glycémie 2 🡪 injections du matin)

* de 0,5 unité si l’injection (rapide ou lente) est inférieure à 7
* de 1 unité si l’injection (rapide ou lente) est comprise entre 7 et 20
* de 2 unités si l’injection (rapide ou lente) est supérieure à 20

2ème cas : hyperglycémie (glycémie>180) : MODIFICATION SI L’HYPERGLYCÉMIE A LIEU DEUX JOURS D’AFFILÉS

Augmentation du taux d’injection du précédent repas :

* de 0,5 unité si l’injection (rapide ou lente) est inférieure à 7
* de 1 unité si l’injection (rapide ou lente) est comprise entre 7 et 20
* de 2 unités si l’injection (rapide ou lente) est supérieure à 20

Donc par exemple, si on a une hyperglycémie au jour 2 au repas du midi mais pas d’hyperglycémie la veille à midi, on ne modifie pas le taux d’injection d’insuline rapide. Si on continue à être en hyperglycémie le jour suivant, alors on modifie le taux d’injection

Comment on modifie les taux d’injections si on fait un bon repas ou si on compte faire du sport ?

Si on compte faire un bon repas, on augmente le taux d’injection d’insuline avant le repas selon les règles ci-dessus. Si on compte faire du sport, on diminue le taux d’injection d’insuline selon ces règles également. On le fait directement, par rapport aux taux d’injection de la veille.

Si on a une hyperglycémie sur deux jours et qu’on compte faire du sport le jour suivant, les effets de modification se compensent : on garde les même taux d’insuline injectés.

Résumé :

Paramètres des fonctions :

Glycémies/taux d’injections sur deux jours/bon repas/sport

2ème étape : coder l’interface graphique (Qt):

initialisation de l’interface : remplir les glycémies/taux d’injections sur deux jours, puis inutile que ces chiffres soient visibles

Type d’interface : le code garde en mémoire les glycémies des deux jours précédents (ou plus) en fonction du moment de la journée. A l’ouverture, il doit savoir à quel moment de la journée on est (en fonction des glycémies enregistrées). Par exemple si on a enregistré du matin au soir sur deux jours, il doit savoir que quand on l’allume, on va lui demander de simuler les injections du matin.

*🡪 réflexion à avoir sur comment on demande à l’utilisateur de rentrer ses chiffres. Les écrire sur un fichier texte tout simple avec des espaces entre les chiffres qu’on charge sur l’interface par exemple (cf tp python de lecture des fichiers sur Qt)*

On doit entrer la glycémie qu’on vient de faire, cliquer sur OUI ou NON au niveau de la question « SPORT », OUI ou NON au niveau « BON REPAS ».

Glycémie : Sport : Repas riche :

**o** Oui **o** Non **o** Oui **o** Non

Simuler

Le simulateur génère une proposition de taux d’injection.

On peut envisager l’ouverture d’une petite fenêtre où apparaîtra la proposition de taux d’injections, avec une question : « Acceptez-vous cette proposition ? » pour savoir si l’utilisateur va effectivement s’injecter les taux proposés. Si « OUI », on enregistre cette valeur, si « NON », nouvelle question : « Combien allez-vous vous injectés ? ». L’utilisateur remplit alors deux cases pour l’insuline rapide et l’insuline lente et on enregistre ces chiffres.

*🡪 possibilité d’utilisation de ces cas où l’injection n’est pas accepté pour une auto-amélioration et rendre le projet un peu plus compliqué. Par exemple, si on est face à un utilisateur qui préfère s’injecter toujours un peu plus que ce qui est simulé, on peut ajouter un facteur qui dépend de ce facteur d’erreur.*

Autre possibilité d’utilisation du code : créer un mode de vérification si on gère bien ou non sa glycémie, une proposition de « diagnostic ». On demande à l’utilisateur de rentrer ses glycémies / insuline sur plusieurs jours et on analyse s’il a fait les bons choix ou pas (a-t-il augmenter / diminuer ses injections aux bons moments en fonctions de ses glycémies ?). On ne prendrait pas en compte le sport/les bons repas.