Compte rendu projet intégrateur Big Data

1. Expliquez pourquoi a-t-on recours aux technologies Big Data dans le réalisation de ce type de système ?

Traitons des 5 V du Big Data :

Le **volume** peut rapidement devenir conséquent. Si des capteurs traite sur un hôpital de manière quotidienne en streaming, le volume pourra devenir rapidement grand.

La **vélocité**, il s'agit de données qui seront transmises en streaming par des capteurs ce qui pourra données des nombreuses données très rapidement pour chaque patient.

La **variété,** de nombreux capteurs pourront être placés pour vérifier la température de la pièce par exemple, sur le patient ou même sur les appareils médicaux pour vérifier leur bon fonctionnement.

La **véracité** des données ici est très importante, les capteurs seront maintenus et diagnostiqués eux même. Cette véracité donnera la précision sur l'état des malades.

La **valeur !** Ici cette valeur est d'une importance cruciale**,** elle pourrait potentiellement sauver des vies. Le nombre de données volumineuses améliorera le diagnostic.

2. Quelle est la partie qui relève du mode 'batch' et quelle partie relève du mode 'streaming' ?

La partie streaming est la partie des données transférée par les capteurs en temps réel. Ces données devront être surveillées par les urgences. Par exemple, les indicateurs anormaux d'un patient alertent les urgences, Des examens d'un patient tout juste établi seront transférés automatiquement au service adéquat.

Pour la partie batch, ces données produites seront conservées dans la base de données.

Le valeur de ces données sera analysés par des médecins, des data analystes ou data scientist qui pourront développer des modèles de prédictions, analysés les paternes des malades et améliorer le diagnostic.

Ici le machine learning sera parfaitement efficace pour la prédiction.

3. Expliquez dans le contexte de ce système e-health le rôle de chaque couche et proposez, pour chaque couche, la technologie Big Data à utiliser en justifiant votre choix.

Ici on a un cluster Kafka-Mongo créé sous docker. L'utilisation de container est particulièrement efficace ici car on pourra facilement répliquer l'architecture dans un autre service ou un autre hôpital.

Dans notre cluster nous avons :

Kafka qui sert de ‘Message Broker’, divise les données sur plusieurs topics MongoDB qui sera une base de données NoSQL. Ici les données seront par capteur, ‘en vrac’, Non relationnel dans des tables. MongoDB et son architecture ‘document based’ est parfaitement adéquat à ce genre de problème. Python sera utilisé dans l’analyse de données, le machine learning et comme langage le mieux adapté aujourd’hui à toute technologie data.

4. Le ‘Message Broker’ est un composant très utile dans ce type de système. Justifier son utilité ?

Le message Broker permet de séparer les données en Topic. Filtrer les données en temps réel et savoir quoi en faire. Un exemple serait de diviser les données reçues et de les transférer à différents services. En aval des topics, on pourra avoir des applications à différents buts. Un de ces applications consomme les données du topic qui lui sera assigné et aura diverses fonctions en fonction du topic et du service associé.

5. Quels sont les ‘topics’ qui vous paraissent pertinents pour ce système au niveau du ‘Message Broker’.

Ici plusieurs façons de voir les choses sont pertinentes. La façon classique serait de considérer les patients malades et de leur assigner l’urgence en cas de valeur critique ou de la transférer à un service adéquat.
Ici, j’ai décidé de choisir une idée non intuitive. Me concentrer sur les patients diagnostiqués non malades.
En effet, ne voulant pas remettre en cause le diagnostic d’un médecin quand à un patient malade, on peut se dire quand est il d’un patient diagnostiqué non malade ? Et si la maladie n’est pas facilement détectable et la batterie de tests effectuées n’était pas suffisante ?
Donc ici, je vais faire un EDA du dataset pour comprendre les corrélations entre les features et la maladie. Si une personne non-malade a des symptômes similaires à une personne diagnostiquée malade, on la diagnostique à mettre sous surveillance et on ne la laisse pas rentrer chez elle sans tests supplémentaires.

D’où l’utilité d’urgence du message broker pour empêcher les personnes à risques de partir.

L'exploratory data analysis à été effectué dans le fichier Heath EDA.ipynb
De plus, je me suis aider de cet article scientifique qui fait une observation similaire:
Recommendation of Attributes for Heart Disease Prediction using Correlation Measure
S.Chellammal, R. Sharmila
https://www.ijrte.org/wp-content/uploads/papers/v8i2S3/B11630782S319.pdf

Lors de l'EDA, on observe la corrélation suivante entre les features et la détection de la maladie :

oldpeak -0.438441
exang -0.438029
ca -0.382085
slope 0.345512
thalach 0.422895
cp 0.434854

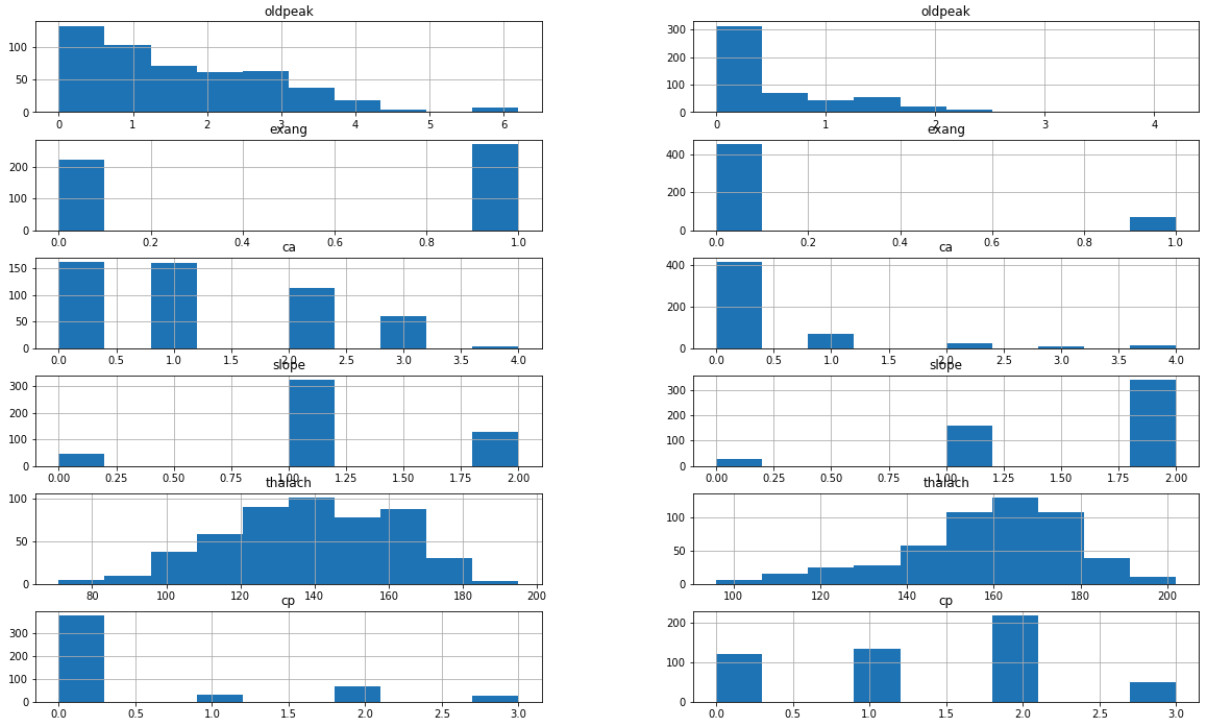
Ces features sont celles avec un score de corrélation les plus élevés en valeur absolue.

La corrélation négative veut dire que plus la valeur est basse, plus le patient à de chance d'être malade.

La corrélation positive veut dire que plus la valeur est haute, plus le patient à de chance d'être malade.

En comparant les histogrammes des Top 6 features des diagnostiqués Non Malades VS Malades :

Histogrammes Non Malade VS Malade (Top 6 features)



- oldspeak, exang et ca = 0 : les malades ont tendance d'avoir une valeur 0 pour ces features comparé aux non malades.
- slope = 2 : Les malades ont majoritaire une valeur 2 ce qui peut être inquiétant pour un non malade

- thalach >= 160 , les non-malades avec un maximum heart rate supérieur à 160 sont rares chez les non-malades. A surveiller.
- La distribution de CP est trop uniforme pour pouvoir décider d'une valeur.

Conclusion :

On voit que sur 1000 entrées on a 119 patients non malades à surveiller en utilisant ce filtre.

On aura donc 3 topics :

- Un pour les diagnostiqués malades qui devront être surveillés par les urgences.

- Un pour les non malades à surveiller qui auront besoin de tests supplémentaires.

- Un pour les non malades qui peuvent juste rentrer chez eux.