***TAREA ACADEMICA 2***



ALUMNO: DIEGO IGNACIO URRUTIA LEY

CODIGO: U201515749

CARRERA: CIENCIAS DE LA COMPUTACION

CURSO: PROGRAMACION CONCURRENTE Y DISTRIBUIDA

PROFESOR: CARLOS ALBERTO JARA GARCIA

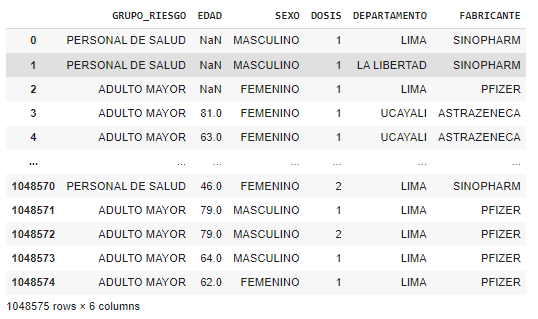
Explicación del dataset seleccionado:

El dataset fue obtenido del portal de datos abiertos del Perú ((<https://www.datosabiertos.gob.pe/>)

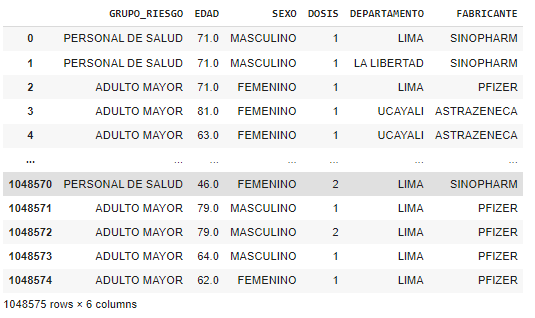
Se selecciono el dataset de Vacunación el cual contenía las vacunas con la cual los peruanos fueron vacunados.

El covid actualmente es un tema de investigación actual, la data me daba información suficiente para aprovechar el algoritmo KNN y descubrir qué tipo de vacunación recibió a partir de los atributos de la data.

De la data se tuvo que hacer preprocesamiento ya que contenía valores nulos:



En este caso seleccionamos la media para cubrir los datos faltantes



Por cuestiones del uso del algoritmo knn se reemplazó la data del grupo de riesgo, sexo y departamento con valores numéricos equivalentes.

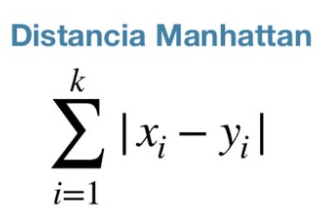
El algoritmo se basó en los siguientes pasos:

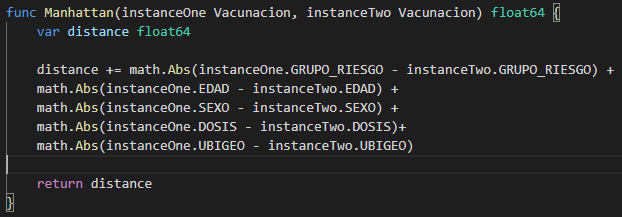
Primero Leer el dataset para ello se leyó desde una URL con el link de github



Segundo Por cada registro sacar la distancia manhattan que resulta ser:

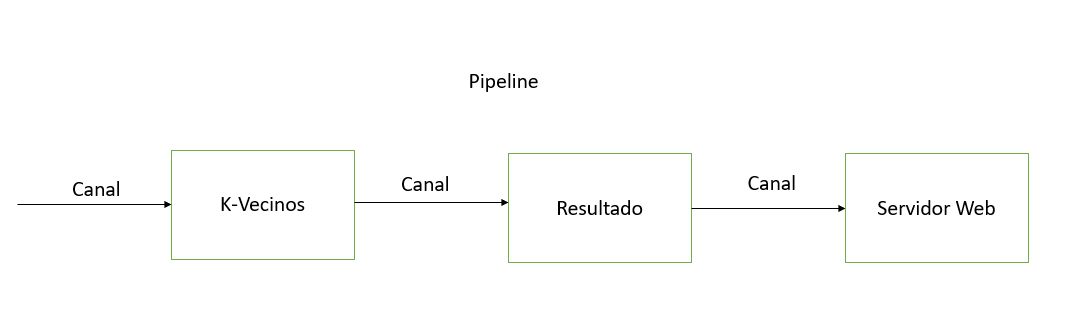
La sumatoria de la diferencia absoluta de los valores del registro y los ingresados por el usuario



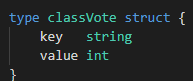


Para los mecanismos de concurrencia se pensó en la arquitectura pipeline que consiste en que la salida de datos será la entrada de la siguiente.

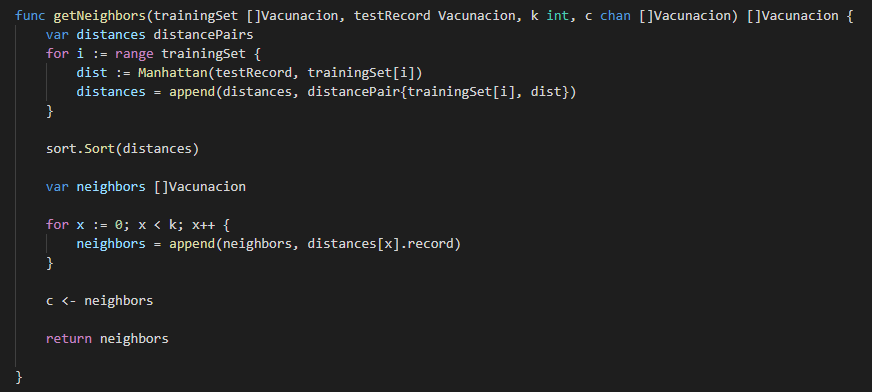
Esta es la arquitectura que se pensó para la tarea 2.



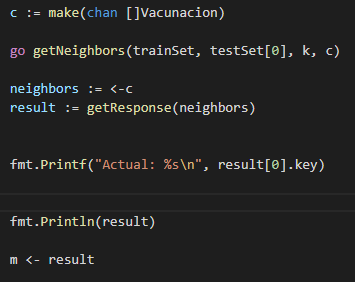
En este caso los datos pasan por los k vecinos y por el canal llega el resultado que viene a ser una estructura que tiene una key y un valor



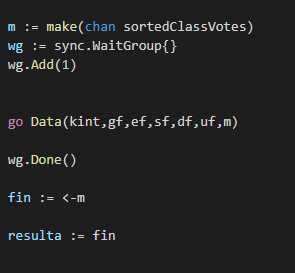
La función getNeighbors recibe como parámetros el training que vendrían ser todos los registros el test que será puesto por el usuario, el número de vecinos K y el canal para recibir los datos

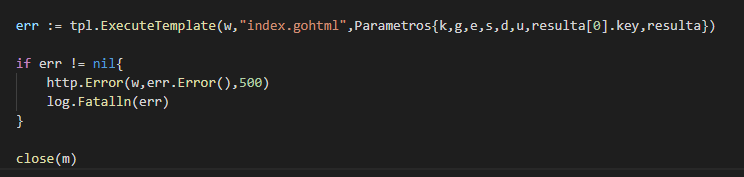


Lo siguiente sería el canal que recibe el resultado



Por último, la función data, donde se va a sincronizar una go rutina y se va cerrar el canal de salida





Se cierra porque ya no se enviarán más datos

Recibe el resultado final para llamarlo en la interfaz web en el handler cerramos con el canal done que es canal booleano

