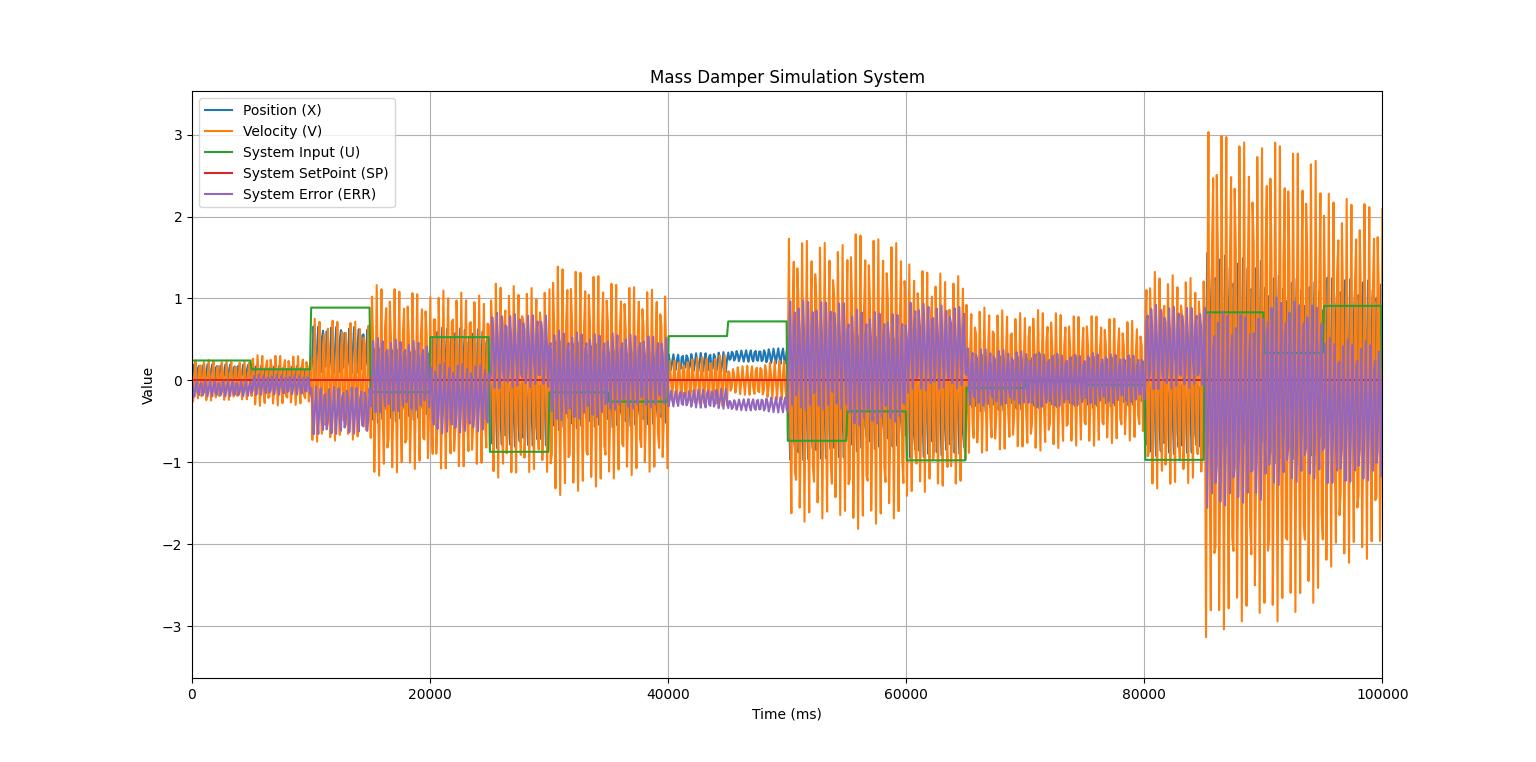
Examen Final Sistemas Inteligentes

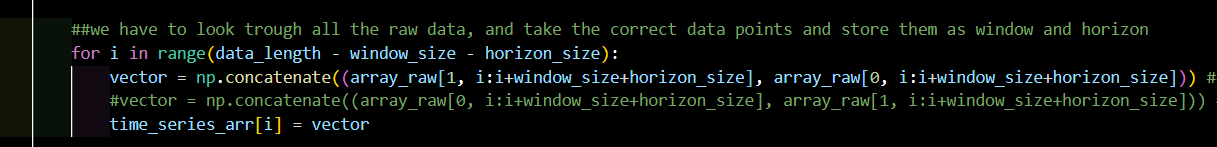
Mauricio Posada Palma

Lo primero que hacemos es ingresar la ecuación de la planta al método de system equation



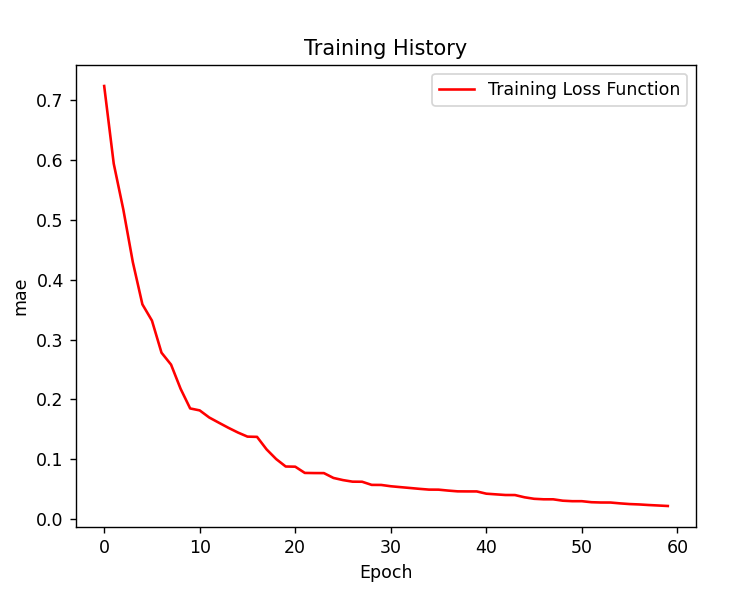
A pesar de ser muy oscilatorio (Puede deberse a que la planta en realidad está en segundos)con las métricas dadas para el profesor se procede a entrenar el modelo para un posible control de PID

Voy a entrenar primero en directa en este pedazo de código



Gráfico

Descripción generada automáticamente



Gráfico

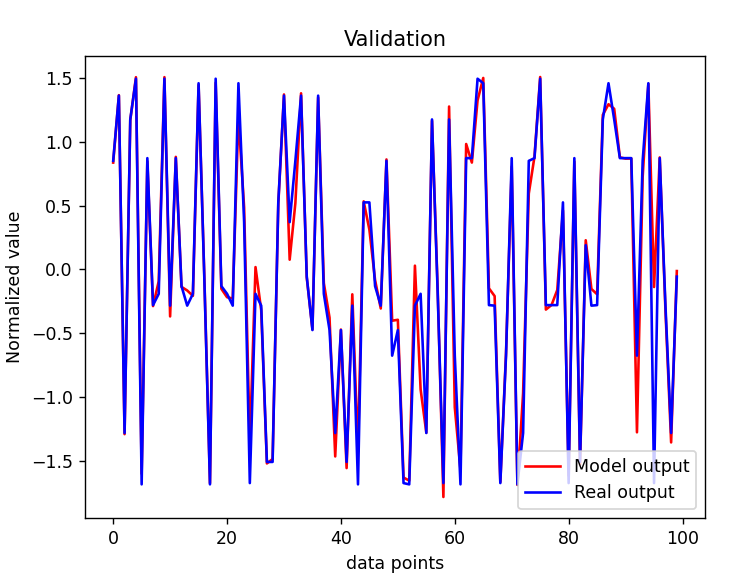
Descripción generada automáticamenteSe evidencia un entrenamiento relativamente bueno , se puede decir que por encima del 90%

Veamos el resultado del accuracy

Accuracy: 92.00%

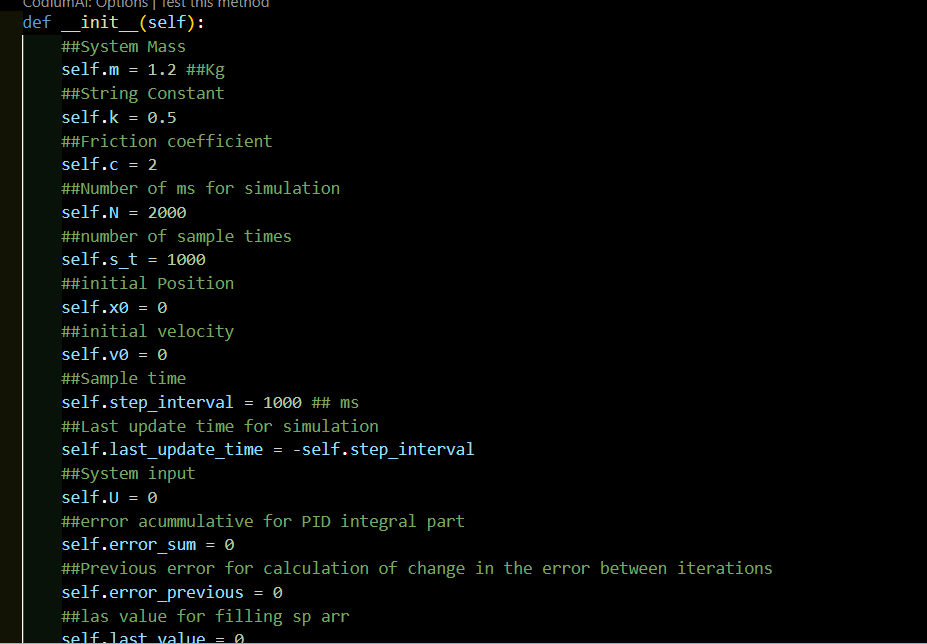
Efectivamente por encima del 90%

Y los resultados de la inversa son los siguiente :



Arrojando un resultado de 96% lo cual nos favorece mas

Ahora con estos valores



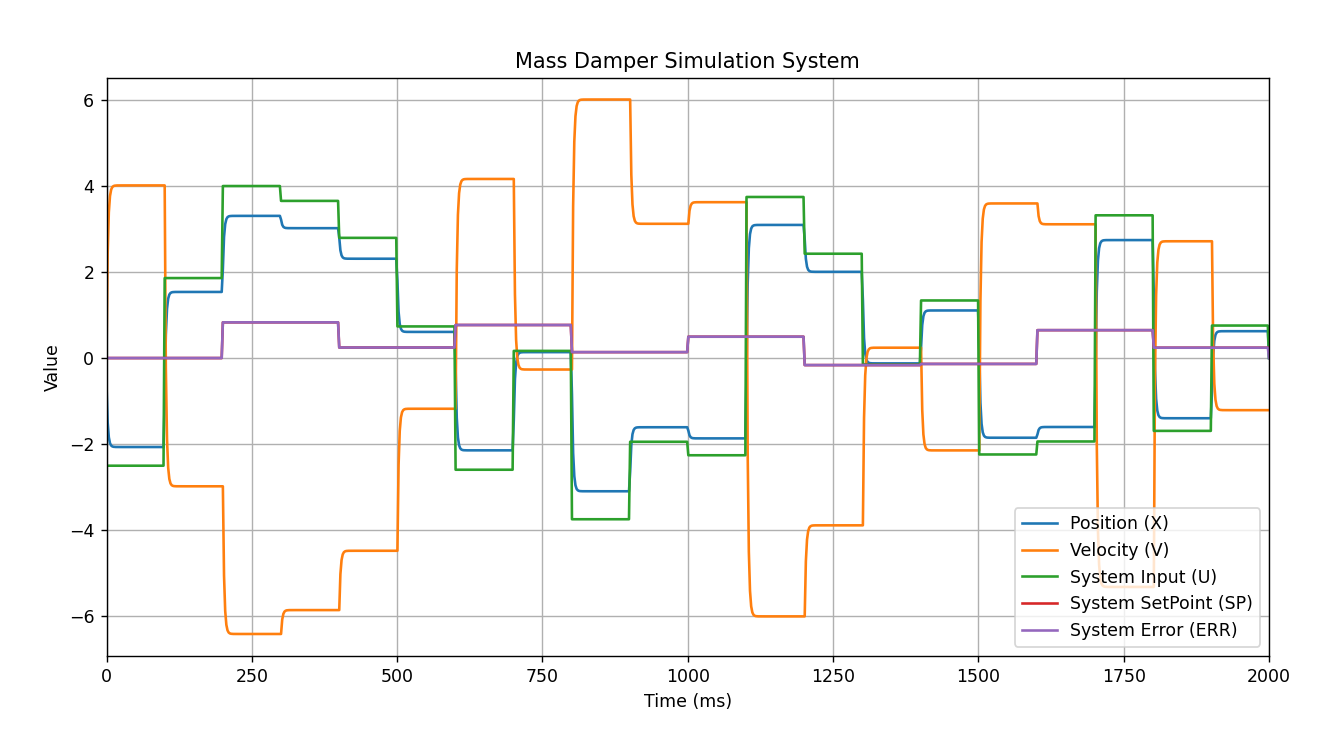
Arrojalo siguiente :

Gráfico, Gráfico de barras

Descripción generada automáticamente

Cambiando los datos para incluir que la variable de posición seaen el instante x[i-1] se hace el intento que no sea en el instante [i]

Quedando de la siguiente manera:



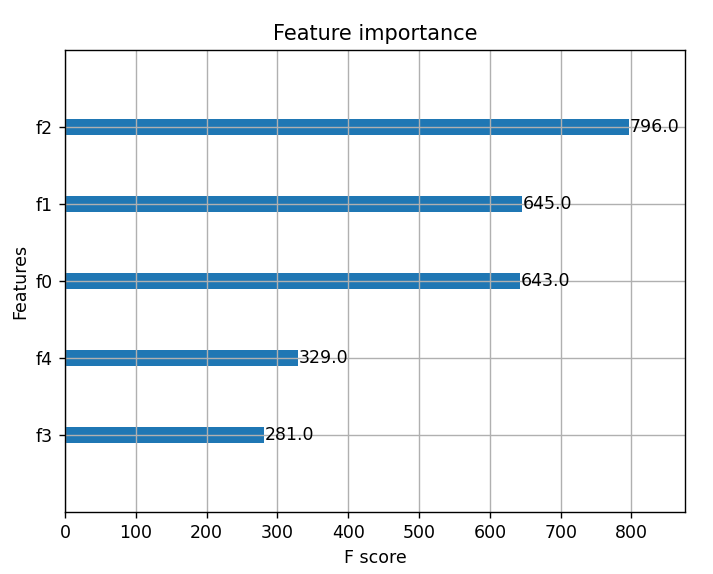
Entrenando de manera inversa saca el siguiente resultado:  


Imagen que contiene Gráfico

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Con un accuracy del 99%

Y por directa :

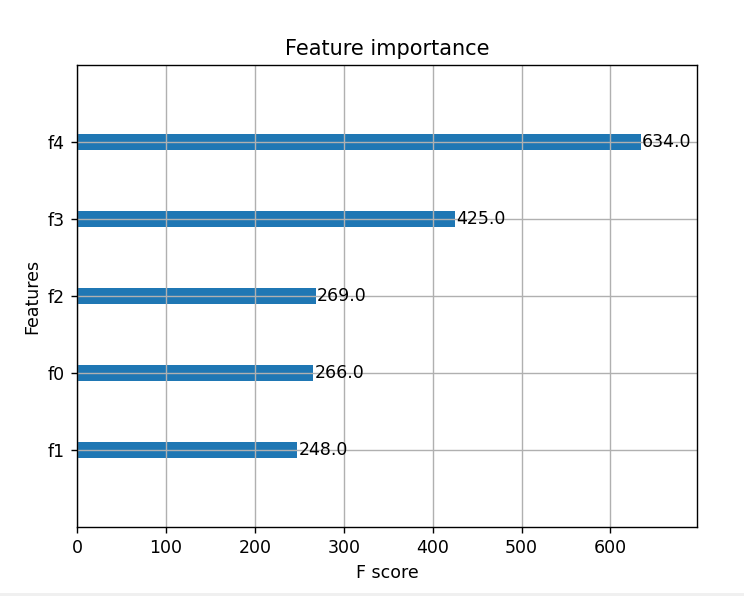


Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Con un accuracy del 100%

Procedi a agrandarle la ventana a 7 y directa para verificar si hay datos que también le pueda servir

Tabla

Descripción generada automáticamente

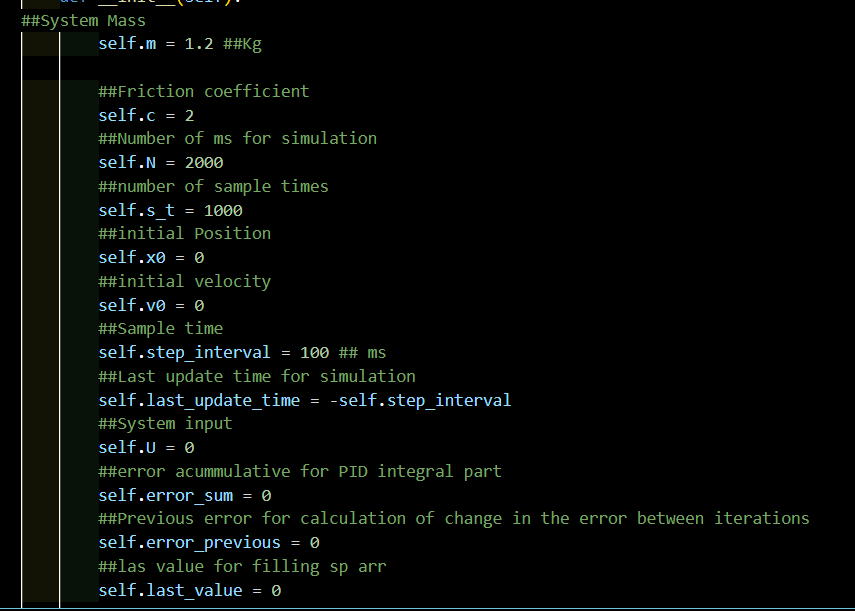
Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Tomando en consideración los errores encontrados se opto por usar la planta de masa resorte con los siguientes parámetros:



Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Dando como resultado lo siguiente para la identificación de la planta

Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

Primer intento ventana 7 en directa

Tabla

Descripción generada automáticamente

Se ve útil todas

Imagen que contiene Forma

Descripción generada automáticamente

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Accuracy: 98.00%

Ahora por inversa

Tabla

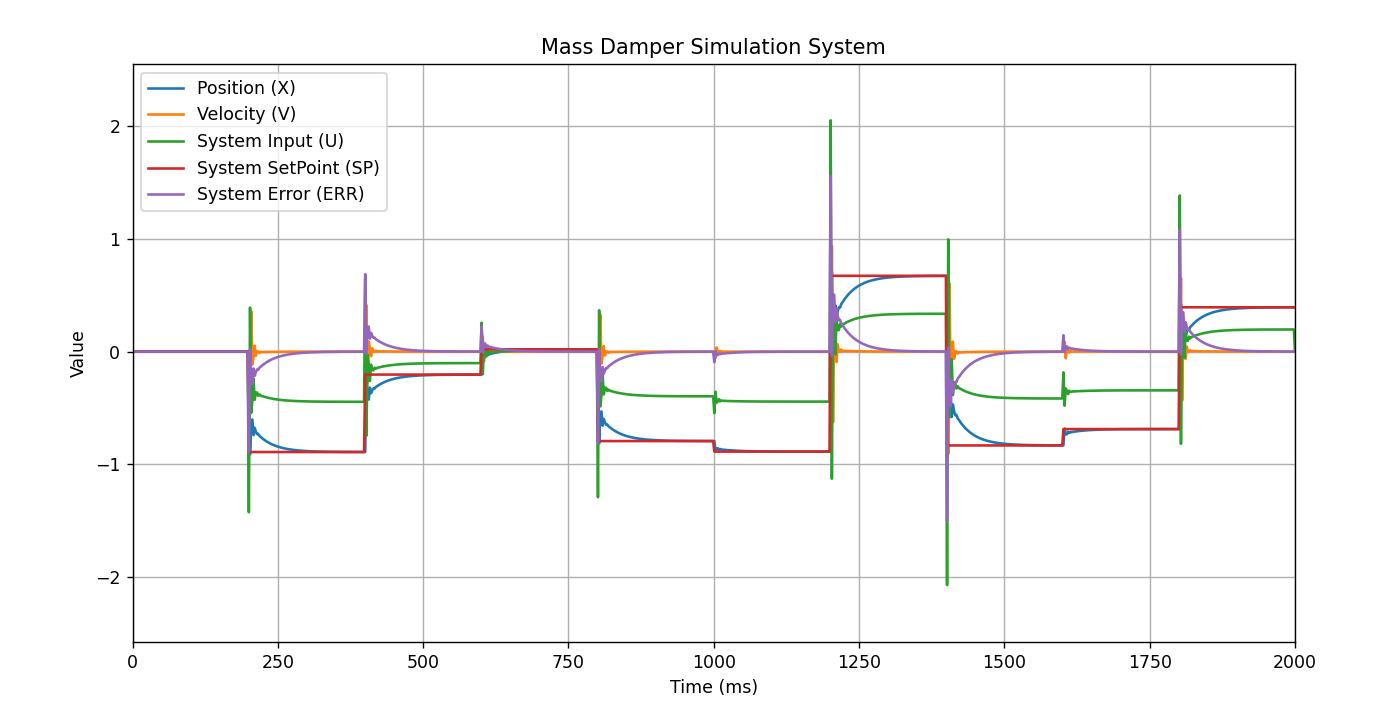
Descripción generada automáticamente

Se evidencia de entrada que no todas son de utilidad

Interfaz de usuario gráfica

Descripción generada automáticamente

Con un acurracy de 96% por lo cual seguiremos con el directo

el PID da de la siguiente manera   


con las siguientes constantes :

        Kp = 1

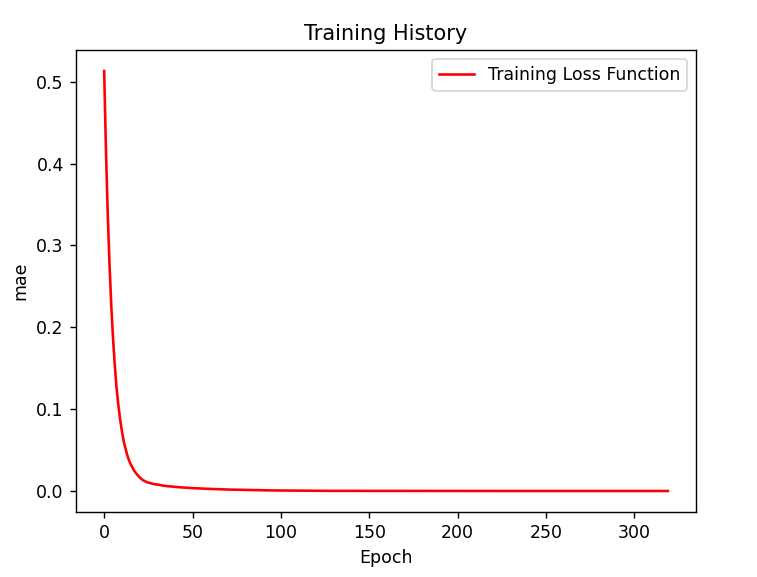
        Ki = 0.1

        Kd = 0.5

Continuando con 7 ventanas para el PID se evidencia que hay mucha información que no es relevante

Gráfico

Descripción generada automáticamente



Interfaz de usuario gráfica

Descripción generada automáticamente

Accuracy del 98% pero igual le reducir ventanas

Al reducir ventanas a 1

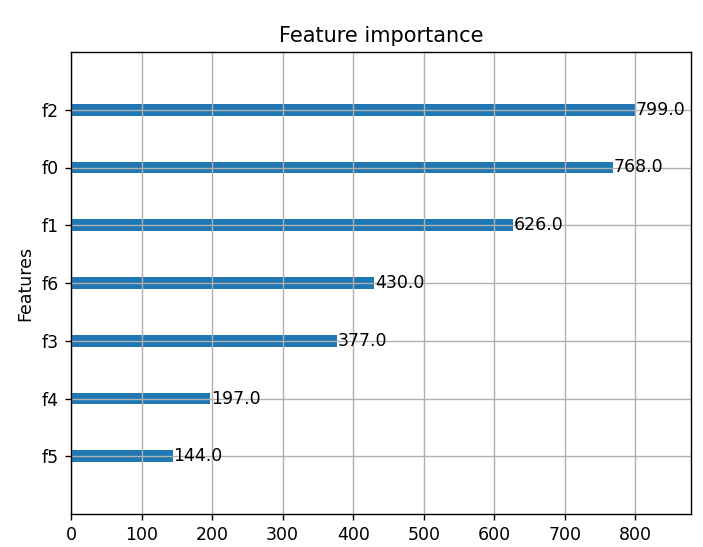


Imagen que contiene Gráfico

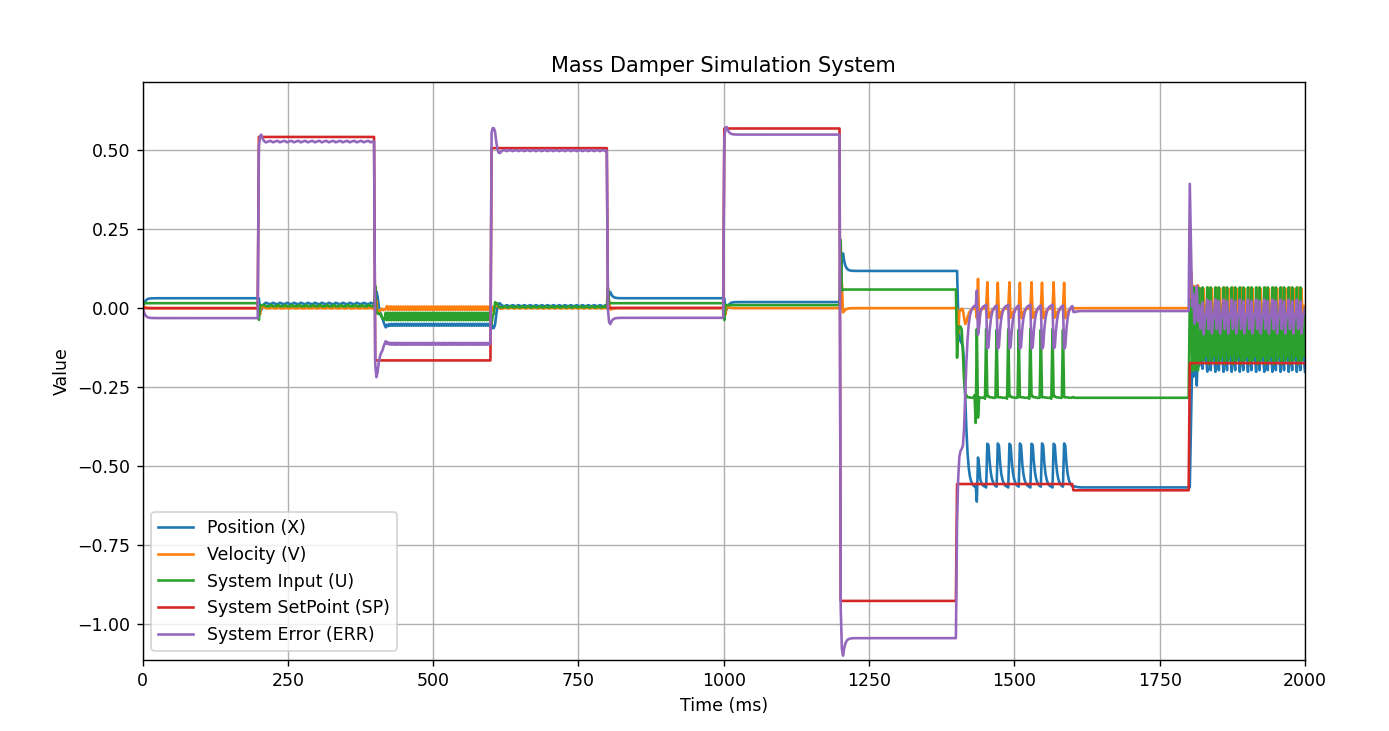
Descripción generada automáticamente

Gráfico

Descripción generada automáticamente

Accuracy de 99% y venatana 1

Ya aplicando el control neuronal

Se evidencia al final que a pesar de tener sus oscilaciones logra controlar, tiene ventana 1

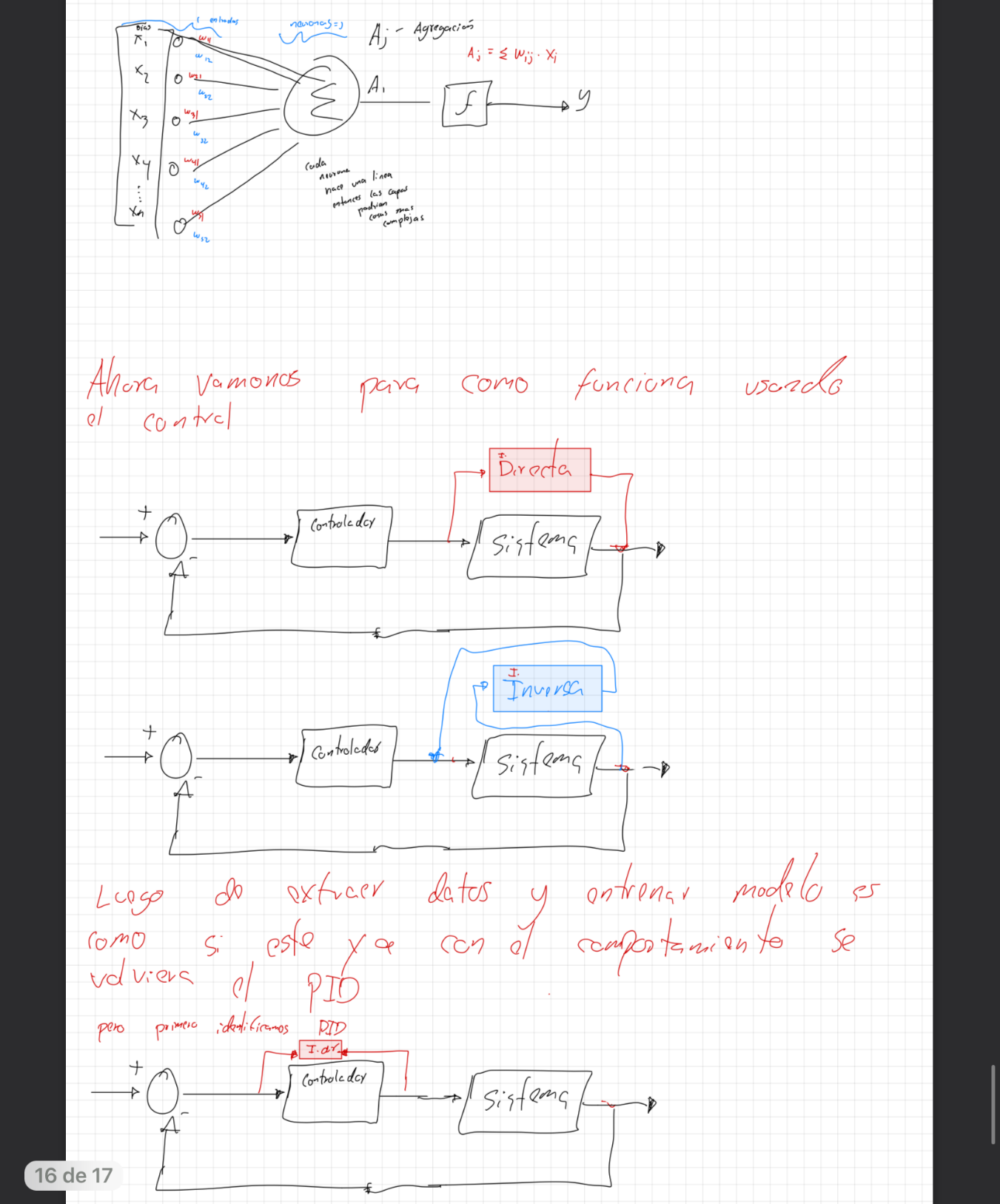
Para el punto 3

Para un modelo por ejemplo primero necesitamos tener en cuenta la cantidad de entradas que vamos a tener, no necesariamente necesitamos saber la cantidad de salidas que vamos a tener, a esto se le llama no supervisado, , con esto tenemos también los pesos de tamaño en este caso nx13

Ahora ya se explicó por encima un perceptrón, ahora vamos a la identificación inversa. Esto quiere decir que a partir de unas salidas conocidas vamos a calcular unas entradas.

Para la formula exacta esto depende mucho de la función de activación que se esté utilizando y también del modelo de red neuronal.

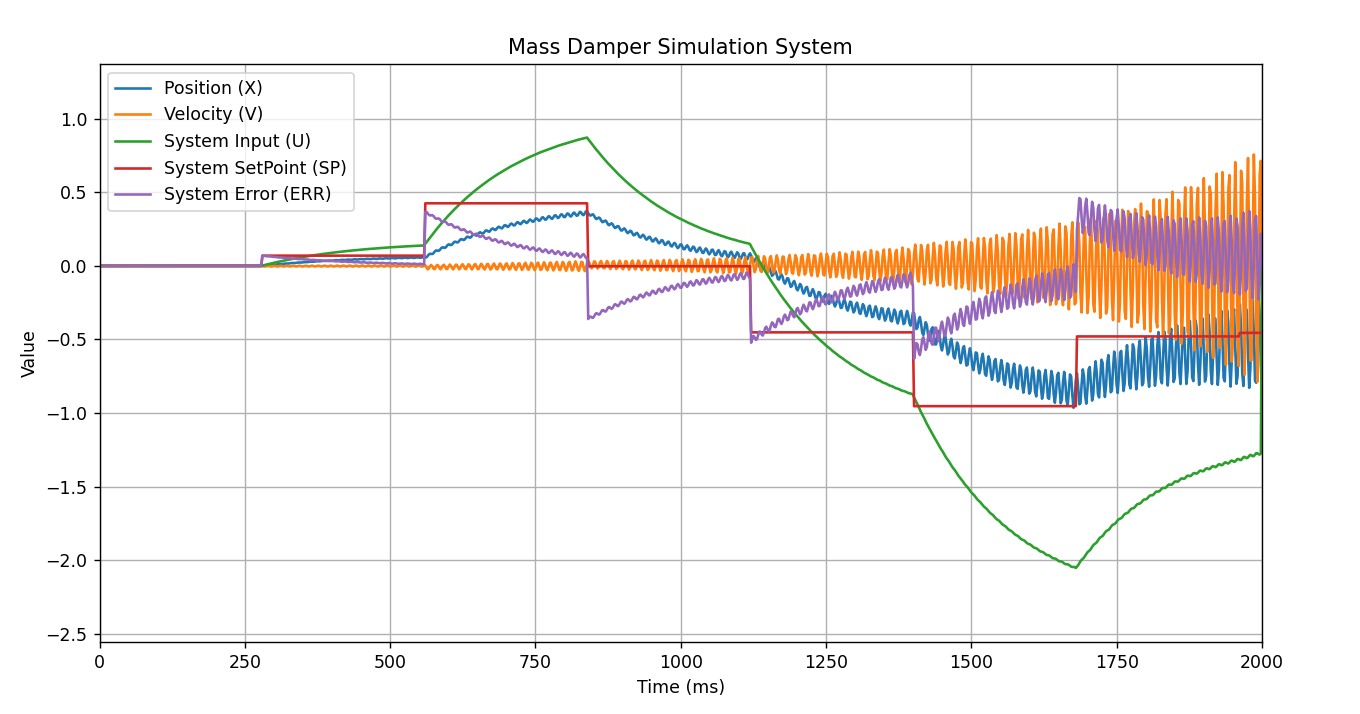
Ahora para el controlador, se necesita o bien las salidas con un setpoint o bien las entradas y un esfuerzo de control , en ambos casos , nos sirve para un controlador ya que este a partir de un setpoint va a realizar un control para que siga dicho voltaje o variable que sea.

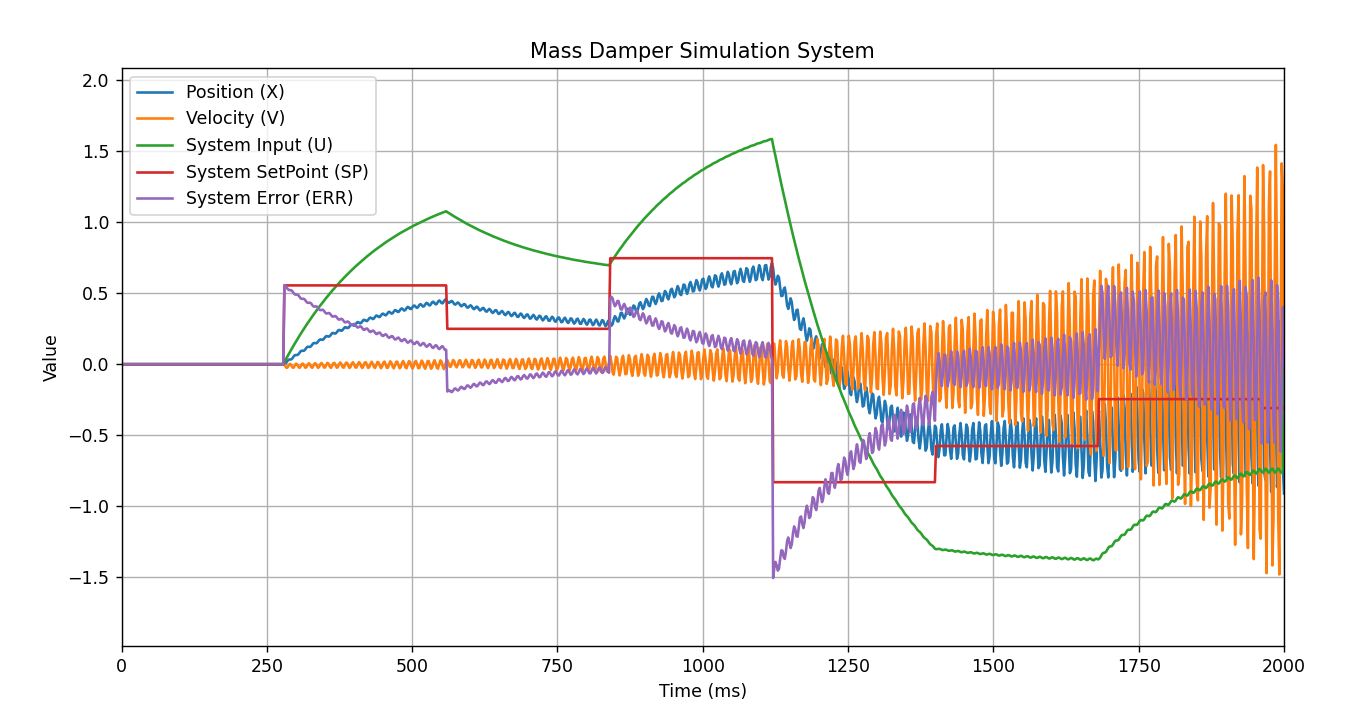


Diagrama, Esquemático

Descripción generada automáticamente

Bueno por ultimo hice un ensayo con la primera planta del péndulo con unasconstantes en el PID y puededecirse que esta bien





Con estas constantes :

    Kp = 0.0001

       Ki = 0.028

       Kd = 0.002

    def system\_equations(self, t, y):

        x, v = y ##actual position and velocity of system

        dxdt = v

        # dvdt = -(self.k/self.m) \* x - (self.c/self.m) \* v + self.U/self.m

        dvdt = -(self.g/self.l) \* x + self.U\*(self.m+self.l)

        return [dxdt, dvdt]

    def \_\_init\_\_(self):

##System Mass

        self.m = 1.2 ##Kg

        ##Friction coefficient

        self.c = 2

        ##Number of ms for simulation

        self.N = 2000

        ##number of sample times

        self.s\_t = 1000

        ##initial Position

        self.x0 = 0

        ##initial velocity

        self.v0 = 0

        ##Sample time

        self.step\_interval = 1000 ## ms

        ##Last update time for simulation

        self.last\_update\_time = -self.step\_interval

        ##System input

        self.U = 0

        ##error acummulative for PID integral part

        self.error\_sum = 0

        ##Previous error for calculation of change in the error between iterations

        self.error\_previous = 0

        ##las value for filling sp arr

        self.last\_value = 0

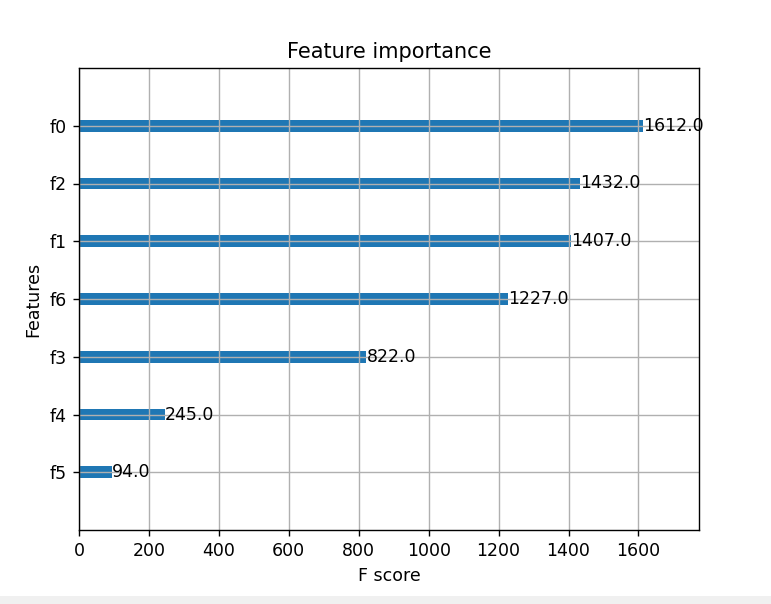
        ##String Constant

        self.g=9.8

        self.l=1.5

        self.k = 0.5

Por temas de tiempo es un ensayo :



Con acurracy de 93%

Entreno el modelo y aplico control neuronal

