Mínimos cuadrados ponderados (introducción a su uso en control)

*Caso 1: exceso de salidas (vbles. controladas)

Presentación en vídeo: http://personales.upv.es/asala/YT/V/mcpy.html

© 2021, Antonio Sala Piqueras. Universitat Politècnica de València. Todos los derechos reservados.

*Este código ejecutó correctamente en Matlab **R2020b**

Objetivo: Cuando se quiere controlar "todo un sistema" con pocas VM (más salidas que entradas) en principio ello no es posible. El enfoque básico es mínimos cuadrados (minimizar $\sum_{i=1}^{N} e_i^2$) pero cuando cada variable tiene unas unidades físicas y unos objetivos de precisión debe cambiarse a mínimos cuadrados **ponderados**. El objetivo es entender el significado de los pesos y de las medidas de error ponderadas.

Tabla de Contenidos

Mínimos cuadrados ponderados (introducción a su uso en control)........ 1
*Caso 1: exceso de salidas (vbles. controladas)........ 1
1.-Exceso de salidas (variables controladas)........ 1
1.1- Mínimos cuadrados no ponderados........ 2
1.2- Pesos de salida........ 3
1.3- Escalado basado en umbrales de errores (equivalente)........ 5
Conclusiones........ 8

1.-Exceso de salidas (variables controladas)

Consideremos un sistema (estático, por simplicidad) $y = Gu$ que, por ejemplo, modele el efecto en régimen permanente de cuatro elementos calefactores sobre la temperatura de una pieza larga.

```
N=12; centros=[4 5.5 7.5 9]; Nu=length(centros);
```

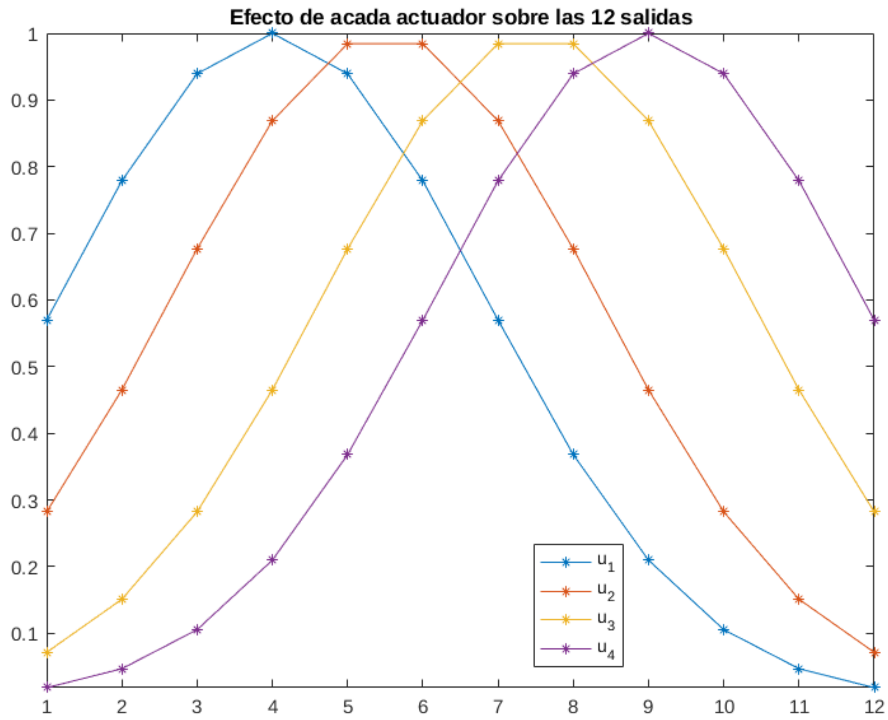
Se dispone de 12 sensores, con una matriz de ganancia dada por:

```
G=zeros(N,Nu);sig=4;
for i=1:Nu
    G(:,i)=exp(-((1:N)-centros(i)).^2/sig^2)';
end
G
```

```
G = 12×4
    0.5698    0.2821    0.0713    0.0183
    0.7788    0.4650    0.1510    0.0468
    0.9394    0.6766    0.2821    0.1054
    1.0000    0.8688    0.4650    0.2096
    0.9394    0.9845    0.6766    0.3679
    0.7788    0.9845    0.8688    0.5698
    0.5698    0.8688    0.9845    0.7788
    0.3679    0.6766    0.9845    0.9394
```

```
    0.2096    0.4650    0.8688    1.0000
    0.1054    0.2821    0.6766    0.9394
      ⋮
```

```
plot(G,'*-'), axis tight,
title("Efecto de acada actuador sobre las 12 salidas")
legend("u_1","u_2","u_3","u_4","location","best")
```



1.1- Mínimos cuadrados no ponderados

Si deseamos (objetivo de control) cierto perfil de temperaturas (arbitrario, en principio):

```
ref=[4.5 6 7 7 7 7.2 7.2 7 7 7 6 4.5]';
```

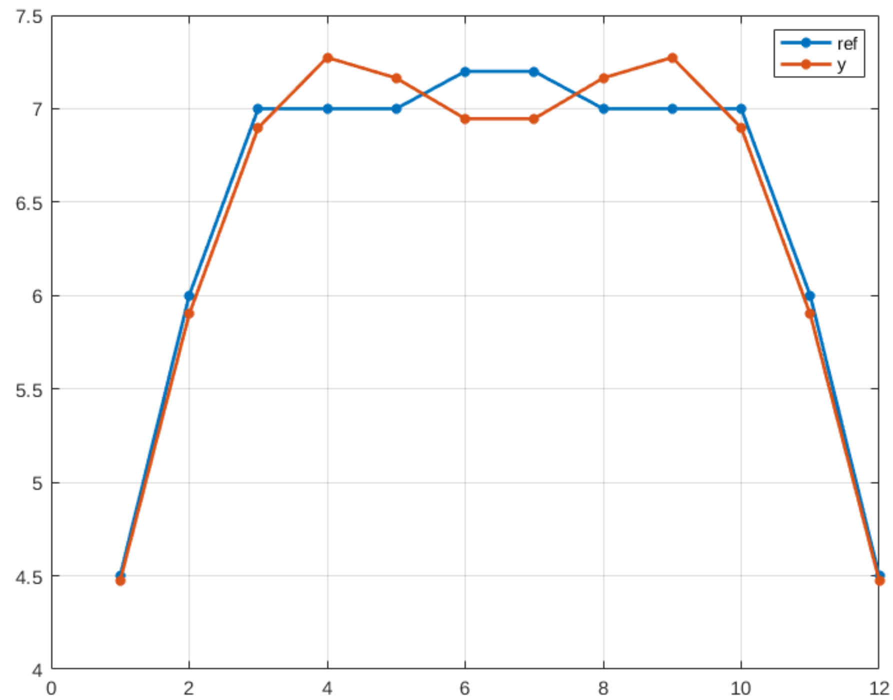
la solución mínimo cuadrática es:

```
uopt=pinv(G)*ref
```

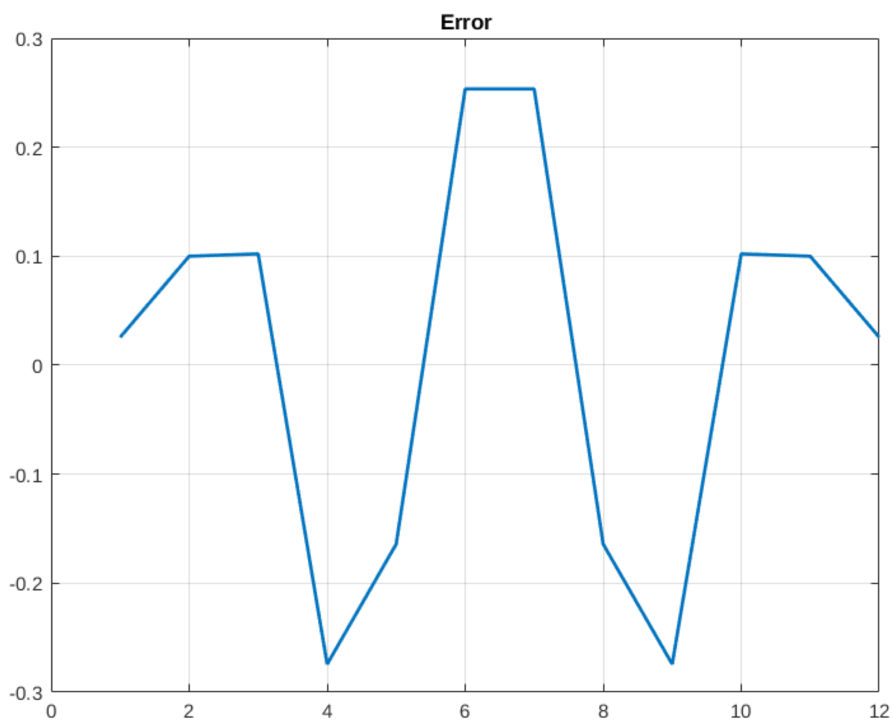
```
uopt = 4×1
    9.5172
   -3.1772
   -3.1772
    9.5172
```

y el perfil alcanzado es:

```
y=G*uopt;
plot([ref y],'*-',"LineWidth",2), grid on
legend("ref","y")
```




```
plot(ref-y,"LineWidth",2), grid on, title("Error")
```



1.2- Pesos de salida

Deseamos mayor precisión en las temperaturas centrales:

```
Peso_y=diag([1 2 4 8 50 100 100 50 8 4 2 1]);
Gesc=Peso_y*G; ref_esc=Peso_y*ref;
uopt2=pinv(Gesc)*ref_esc
```

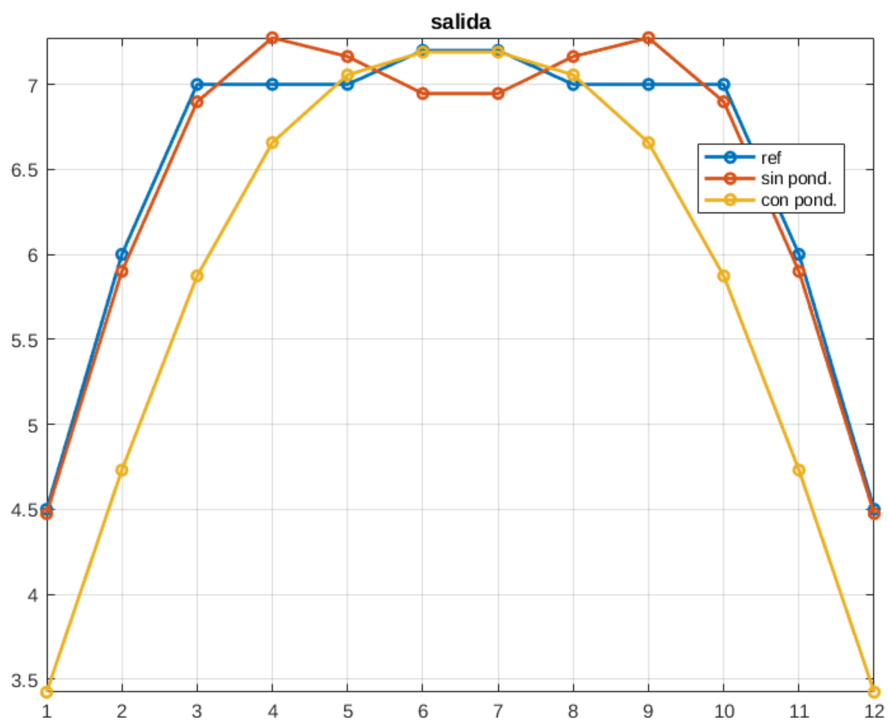
```
uopt2 = 4×1
    6.2030
   -0.6340
   -0.6340
    6.2030
```

*Escalar primero y mínimos cuadrados (pseudoinversa) clásicos después coincide con fórmulas en libros de mínimos cuadrados ponderados (usando el cuadrado del peso):

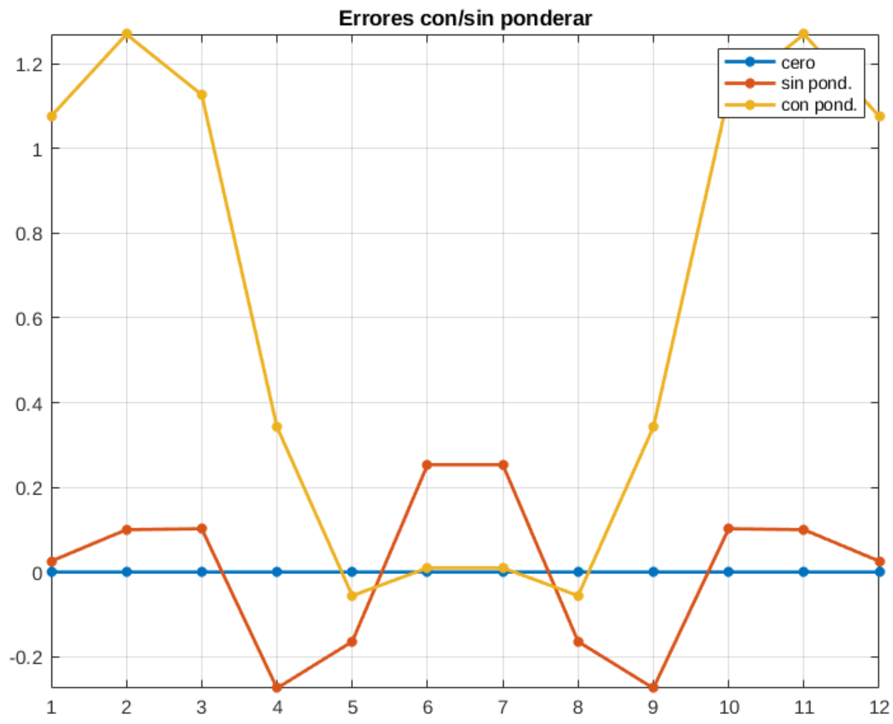
```
W=Peso_y^2;
uopt2_otravez=inv(G'*W*G)*G'*W*ref
```

```
uopt2_otravez = 4×1
    6.2030
   -0.6340
   -0.6340
    6.2030
```

```
y2=G*uopt2;
plot([ref y y2],'o-',"LineWidth",2), grid on, axis tight,
legend("ref","sin pond.","con pond.","location","best"), title("salida")
```




```
plot([zeros(N,1) ref-y ref-y2],"*-","LineWidth",2), grid on,  axis tight, legend("cero'
title("Errores con/sin ponderar")
```



1.3- Escalado basado en umbrales de errores (equivalente)

```
Umbral_Error_aceptable=[4 4 3 2 .2 .1 .1 .2 2 3 4 4]*0.5;%reducir si >1 criterio1 abajo
Matriz_esc_e=diag(Umbral_Error_aceptable);
```

Proponemos que "peso" de los mínimos cuadrados ponderados sea la inversa de las cotas de error...

```
Gesc=inv(Matriz_esc_e)*G;
ref_esc=inv(Matriz_esc_e)*ref;
%implícitamente, hacemos cambio de variable
%error=Matriz_esc_e*error_esc
%error_esc=inv(Matriz_esc_e)*error
```

La solución del problema de mínimos cuadrados con ese peso:

```
uopt2=pinv(Gesc)*ref_esc;
y2=G*uopt2;
```

tiene un error:

```
e2=ref-y2;
```

que si dividimos por su "umbral" tenemos

```
error_escalado=Matriz_esc_e\e2;
```

El error es idéntico al error con las matrices escaladas:

```
error_escalado_igual=ref_esc-Gesc*uopt2;
norm(error_escalado-error_escalado_igual) %coinciden
```

```
ans = 2.1045e-14
```

Si cada elemento (abs) es menor de 1, está por debajo del umbral, y la solución sería válida.

```
max(abs(error_escalado)) %si realmente nos interesa esta geometría, deberíamos plantear
```

```
ans = 0.7697
```

Si $(\sum_{i=1}^{N} e_{esc,i}^2)^{1/2} < 1$ entonces también todos los elementos están por debajo de su umbral:

```
crit1=norm(error_escalado)
```

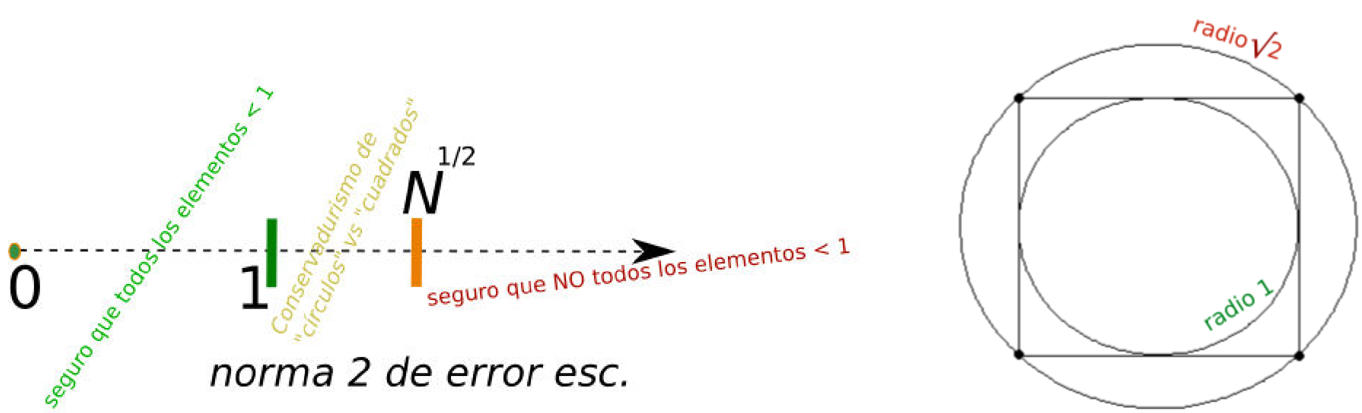
```
crit1 = 1.8539
```

aunque sumar $\sqrt{1+1+\ldots+1} = \sqrt{N}$ puede tener valor mayor a 1 sin superar individualmente el límite.

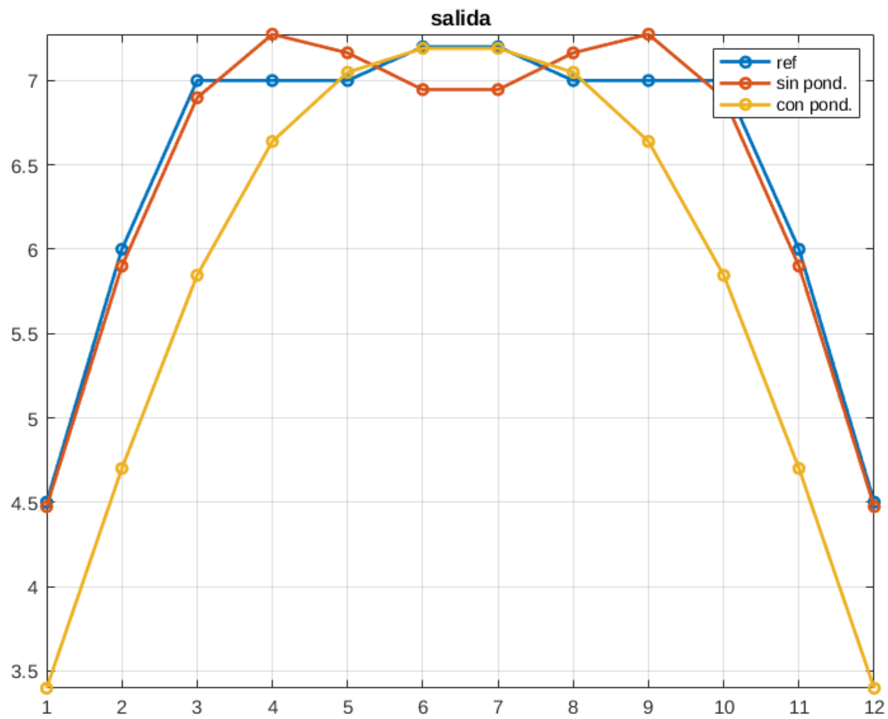
```
crit2=norm(error_escalado)/sqrt(N)
```

```
crit2 = 0.5352
```

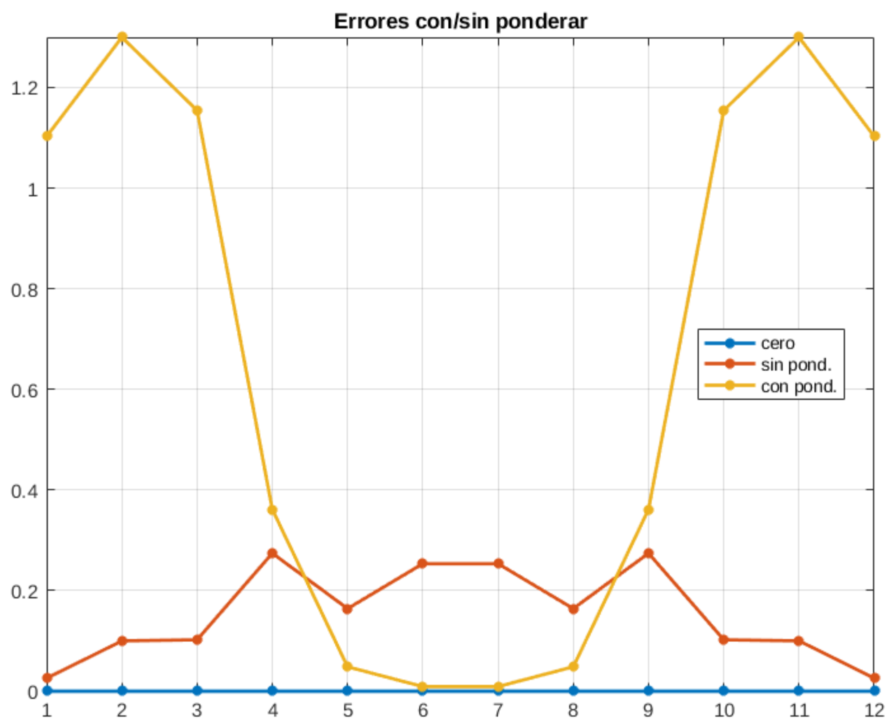
- Si $crit1 < 1$ entonces *todos* los errores por *debajo* del umbral ($crit1 < 1$ es cond. **suficiente** para asegurar que todo **SÍ** está **por debajo** del umbral)
- Si $crit2 > 1$ (o sea, $crit1 > \sqrt{N}$) entonces *seguro* que alguno de los errores *sobrepasa* el umbral ($crit1 > \sqrt{N}$ es cond. **suficiente** para certificar que **NO** se cumplen los umbrales; $crit1 \leq \sqrt{N}$ es condición **necesaria** para que todos los errores estén **por debajo** del umbral aceptable).



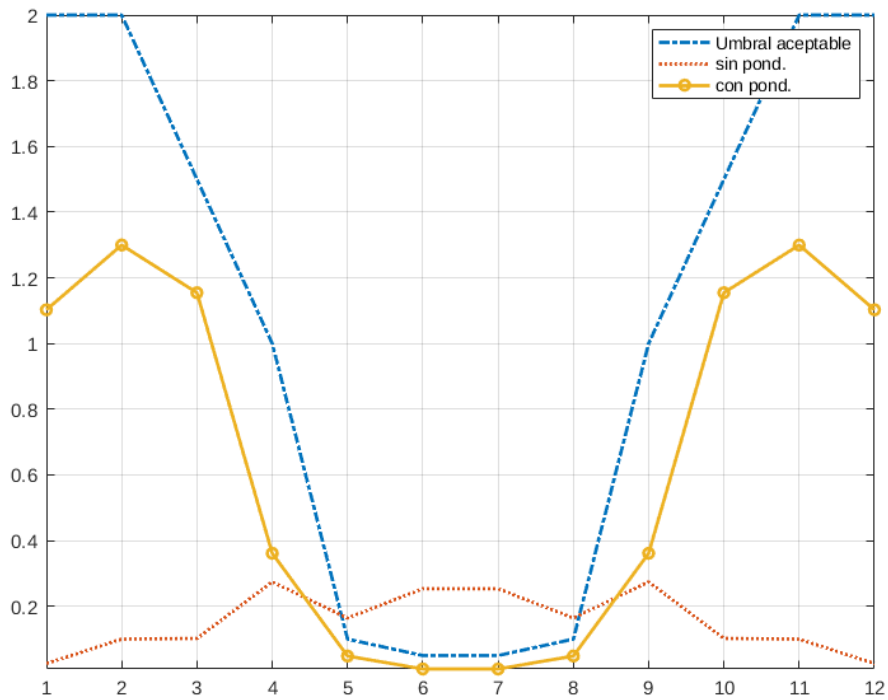

```
plot([ref y y2],'o-',"LineWidth",2), grid on, axis tight, legend("ref","sin pond.","cor
```




```
plot(abs([zeros(N,1) ref-y ref-y2]),"*-","LineWidth",2), grid on,  axis tight, legend('
title("Errores con/sin ponderar")
```




```
plot(1:N,Umbral_Error_aceptable','-.',1:N, abs(ref-y),":",1:N,abs(ref-y2),"o-","LineWid
```



Conclusiones

Cuando se quiere controlar "todo" con pocas VM (más salidas que entradas) en principio ello no es posible. El enfoque básico es mínimos cuadrados (minimizar $\sum_{i=1}^{N} e_i^2$) pero cuando cada variable tiene unas unidades físicas y unos objetivos de precisión debe cambiarse a mínimos cuadrados **ponderados**.

En un primer enfoque, se trata de minimizar $\sum_{i=1}^{N} (w_i e_i)^2$, dejando a criterio del usuario el valor del peso w_i. Cuanto mayor sea cada peso w_i , más cercano a cero se desea ese error (a expensas de otros).

En un segundo enfoque, lo que se deja a criterio del usuario es especificar un cierto umbral máximo de error (aproximado).

Las ventajas del segundo enfoque (umbrales, peso = inversa de umbrales) son:

- que trabajamos en unidades "físicas" en dichos umbrales, y por lo tanto, resulta trivial que unas salidas sean voltios, otras milivoltios, y otras litros... con su especificación de precisión (en sus unidades), se forma el peso.
- que de forma rápida (aproximada por un posible error en un factor $\sqrt{N}$) podemos asegurar que se cumplen las especificaciones (error<umbral) si la norma del vector de "error escalado" es menor a 1.