

## Tarea 6, Fecha de entrega: Antes del 5 de Marzo

Usando las funciones `POLYFIT(X,Y,N)` y `POLYVAL(P,X)` vistas en clase, las cuales ajustan un polinomio de grado  $N$  a pares de puntos definidos en los vectores  $(X, Y)$ , y evalúan un polinomio  $P$  en los valores  $X$ , respectivamente.

Programar una función que:

i) dado un par de vectores  $X, Y$  dados (observaciones de una función desconocida), encuentre el polinomio de menor grado tal que el polinomio evaluado en las  $X$ s sea muy parecido a las  $Y$ s, es decir tal que

$$\frac{1}{n} \sum_i (Y_i - P(x_i))^2 < 0.001$$

Para hacer lo anterior, el algoritmo debe de encontrar el polinomio de grado 1 que intenta ajustar las observaciones, evaluar el error, si este es aún grande, encontrar el polinomio de grado 2, evaluar el error, si este es aún grande, encontrar el polinomio de grado 3, etc.

El prototipo de la función es:

```
function p = ajustaPolyMenorGrado( X, Y)

end
```

donde  $p$  son los coeficientes del polinomio que regresa `POLYFIT(X,Y,N)`, y  $X$  y  $Y$  son las observaciones de la función desconocida.

Probar desde `main` la función con (copiar y pegar en su código)

```
x = -2*pi:0.25*pi;

y = [ 2.3198    2.6070    2.7156    2.5057    2.0800
      1.6988    1.5174    1.5679    1.8447    2.2743
```

2.6360	2.7030	2.4928	2.1853	1.9105	1.7076
1.5717	1.4879	1.4440	1.4332	1.4535	
1.5084	1.6064	1.7610	1.9861	2.2782	
2.5748	2.7183	2.5511	2.1394	1.7396	1.5282
1.5478	1.7963	2.2150	2.6019	2.7133	
2.5303];					