DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico

Polynomial Regression

Reconocimiento de patrones

Integrante	LU	Correo electrónico
Rodrigo Oscar Kapobel	695/12	rokapobel35@gmail.com

Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Ciudad Universitaria - (Pabellón I/Planta Baja)
Intendente Güiraldes 2160 - C1428EGA
Ciudad Autónoma de Buenos Aires - Rep. Argentina
Tel/Fax: (54 11) 4576-3359
http://www.fcen.uba.ar

Índice

1. Introducción **3**

2. Implementación **3**

3. Casos de estudio **4**
3.1. Hiperparámetro M 4
3.2. Hiperparámetro λ 5

4. Conclusiones **7**

1. Introducción

En este trabajo práctico se implementa un algorítmo de regresión polinomial.

La idea es, a partir de un set de datos $\mathcal{D} = \{(x_i\ ,\ t_i)\}$, i=1,...,N, $t \in$ [0, 1] uniformemente distribuidos que provienen de la función $t = sin(2\pi x)$ más un ruido independiente con distribución $\mathcal{N}(0, \sigma^2)$, predecir valores desconocidos a partir del cálculo de un vector de pesos w^*, que es el que configura el polinomio predictor:

$$y(x_i, w^*) = w_0 + w_1 x_i^1 + w_2 x_i^2 + ... + w_M x_i^M$$

Aplicando el método de cuadrados mínimos lineales a éste polinomio se podrá obtener el vector de pesos w^*. Para ello debe calcularse $(XX^t)^{-1}X^t t$, con X la matriz que en la fila $i - esima$ tiene como vector a $< x_i^1, x_i^2 ... x_i^M >$

En el caso de utilizar término de regularización la fórmula será $(XX^t + \lambda I)^{-1}X^t t$.

Las fórmulas mencionadas provienen de minimizar el error cuadrático (derivar e igualar a cero) $\frac{1}{2}\sum_{n=1}^{N}(t_n - y(x_n, w))^2$ donde en el caso de usar término de regularización se suma $\frac{\lambda}{2}\sum_{j=0}^{M}|w_j|$

Se realizan dos estudios para los hiperparámetros M y λ, donde el último forma parte del término de regularización que se suma a la ecuación de cuadrados mínimos lineales para disminuir el overfitting.

2. Implementación

Para correr el test que grafica la influencia del grado del polinomio M se debe correr desde el directorio TP1 el siguiente comando:

```
python PolynomialRegressionTest.py -t a
```

Para correr el test que grafica la influencia del parámetro de regularización λ:

```
python PolynomialRegressionTest.py -t b
```

para más opciones:

```
python PolynomialRegressionTest.py -h
```

Podrá elegirse la cantidad de datos, o hacer regresión sobre otra función ingresando los comandos indicados por help.

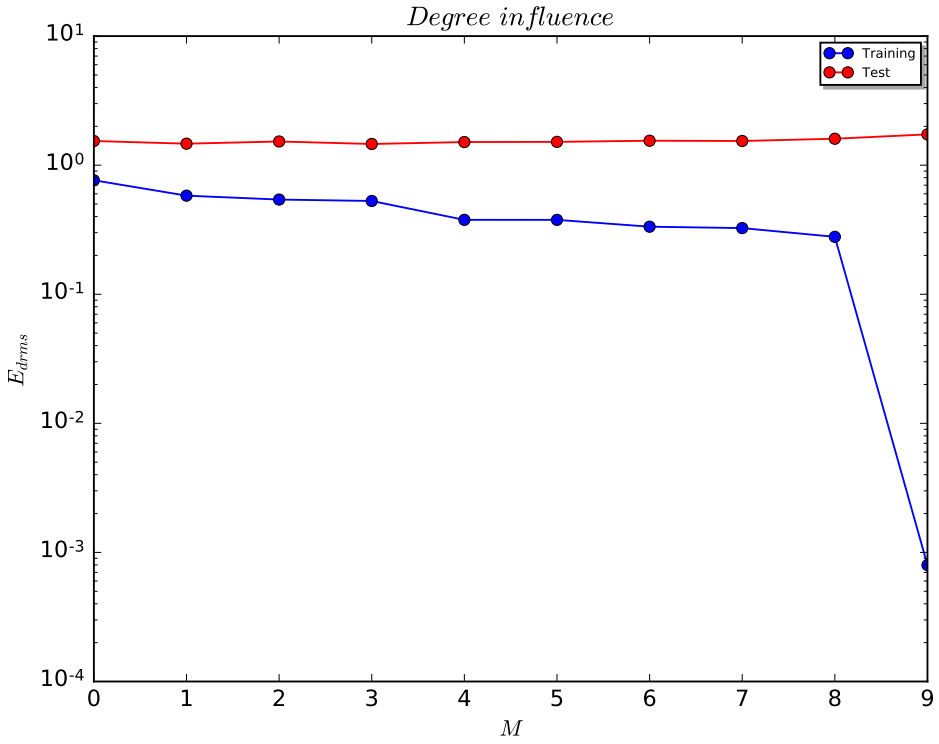
```
-t TEST a: Influence of the degree hyperparameter.
        b: Influence of the lambda hyperparameter.
-n N Number of data to generate.
Test A will run for degree M: 0..N-1.
-f FUNCTION  Function to test:['sin', 'log', 'pol']
```

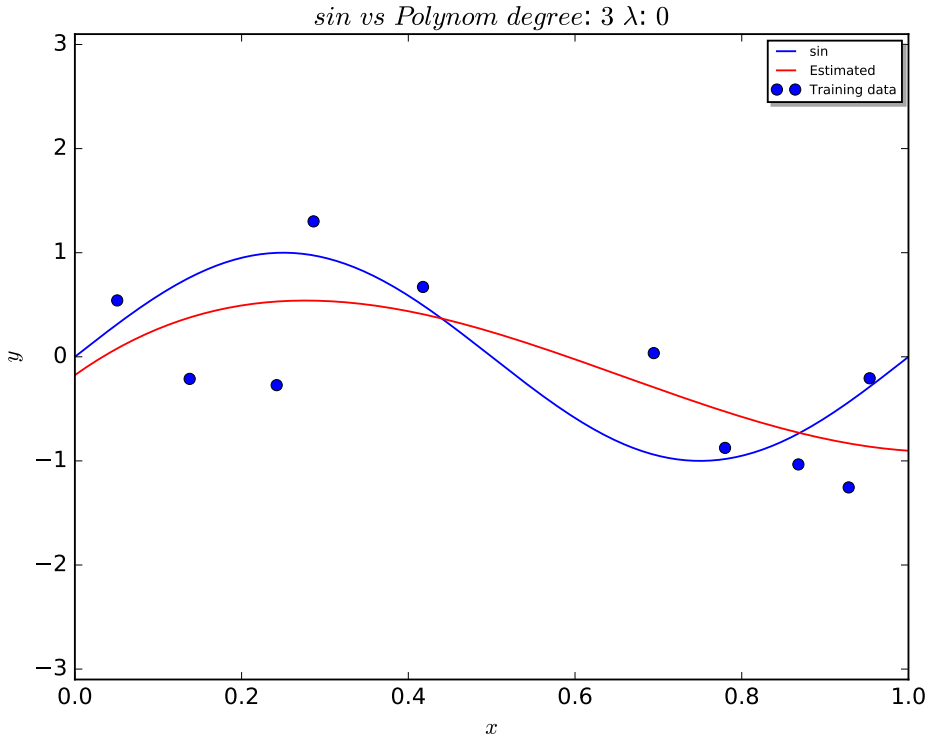
3. Casos de estudio

3.1. Hiperparámetro M

Puede verse que a medida que el grado del polinomio aumenta, la precisión para los datos de entrenamiento aumenta, es decir, el error disminuye. Pero también crece el error para los datos de test, es decir, el polinomio se ajusta muy bien para los datos de entrenamiento pero tiene un peor desempeño a la hora de predecir. Esto se conoce como overfitting.

Al ser M muy grande, los valores de w son muy grandes, lo que produce que en los sectores donde el polinomio no fué entrenado, haya mayores fluctuaciones y peores predicciones, ya que estas fluctuaciones muy probablemente no se ajusten a la función buscada.

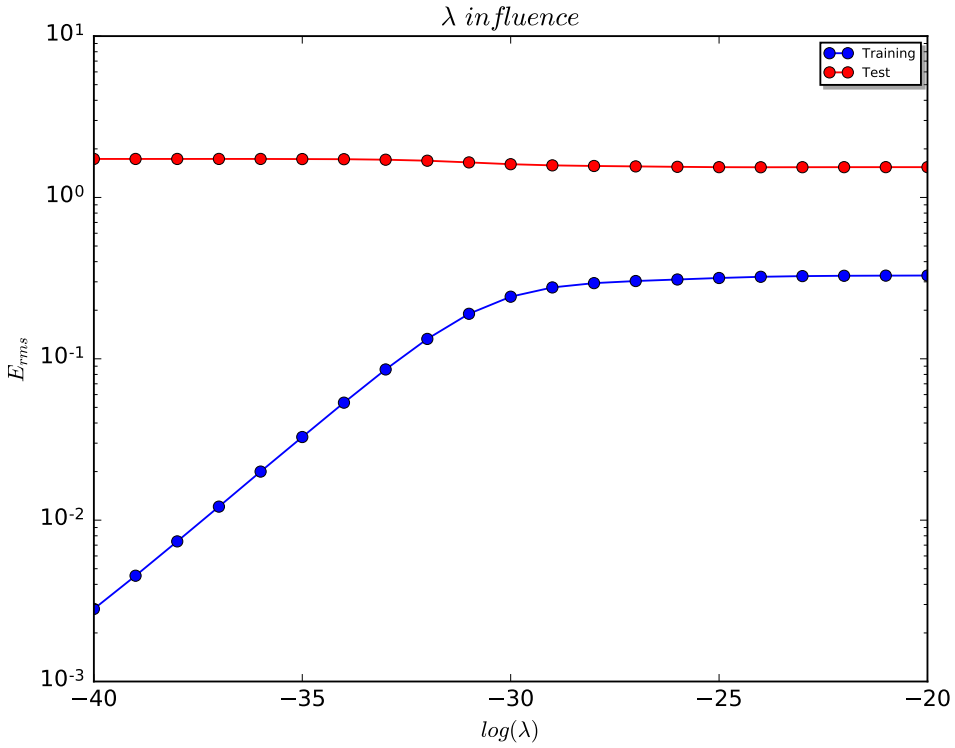


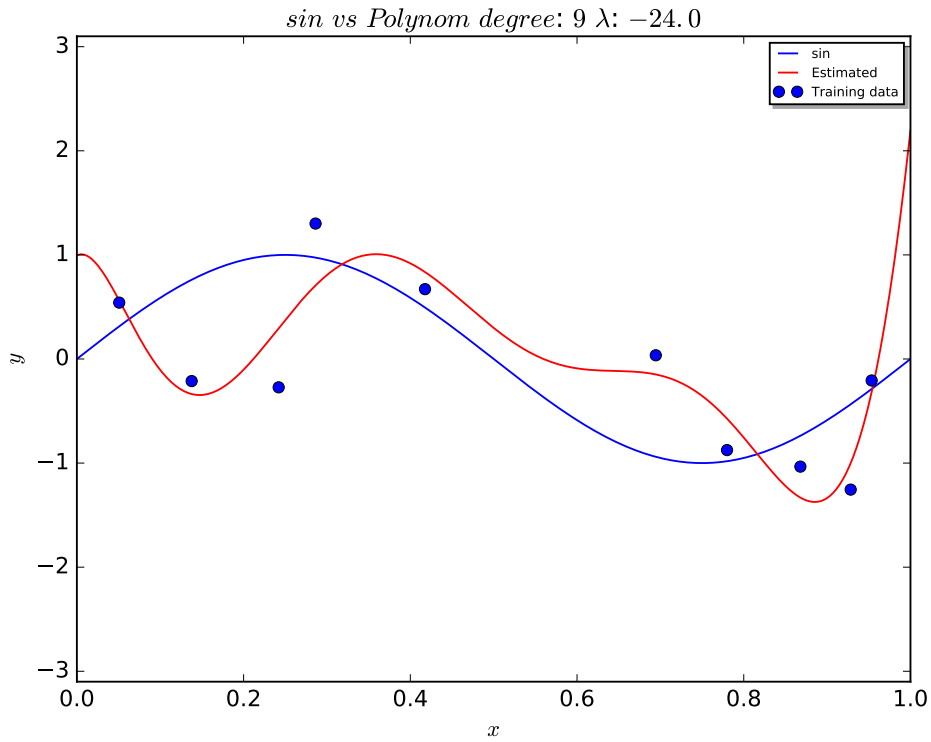


3.2. Hiperparámetro λ

Para intentar corregir los errores producidos por un valor del hiperparámetro M muy grande, se utiliza un término de regularización en el entrenamiento de w. Este témino se ajusta mediante el hiperparámetro λ. Este término hará que disminuyan las fluctuaciones y como consecuencia hará que los valores de w no sean tan grandes.

Para este caso de estudio se elige un polinomio de grado 9. Puede verse que para un un valor de $log(\lambda)$ = -24 se obtiene el mejor ajuste.





Vale aclarar que los valores elegidos para los hiperparámetros siempre dependen del polinomio elegido.

4. Conclusiones

1. Los hiperparámetros son muy importantes a la hora de configurar los algorítmos de regresión. Su elección depende de la función que se desea predecir y por lo tanto requieren pre entrenamientos de prueba.

2. Algunas de las aplicaciones comerciales de los algoritmos de regresión en la industria se encuentran en la investigación de las ciencias sociales, análisis de comportamiento e incluso en la industria de seguros para determinar la validez de las reclamaciones. El correcto ajuste de los hiperparámetros es escencial para su éxito comercial.