Transformaciones No-Lineales

M.Sc. William Caicedo Torres

Universidad Tecnológica de Bolívar

caicedo77@gmail.com

26 de octubre de 2016

- Hasta ahora hemos visto una par de modelos de regresión y clasificación, donde las hipótesis hacen parte del conjunto de hipótesis lineales o hiperplanos.
- Ventajas: Los modelos lineales necesitan poca información para ser entrenados (con respecto a otros modelos más complejos).
- Desventajas: Los modelos lineales son limitados en su expresividad.

Modelos lineales de regresión y clasificación

- Hasta ahora hemos visto una par de modelos de regresión y clasificación, donde las hipótesis hacen parte del conjunto de hipótesis lineales o hiperplanos.
- Ventajas: Los modelos lineales necesitan poca información para ser entrenados (con respecto a otros modelos más complejos).
- Desventajas: Los modelos lineales son limitados en su expresividad.
- Entonces qué hacer para atacar problemas de regresión y clasificación en los cuales una linea recta no sea suficiente?

- Hasta ahora hemos visto una par de modelos de regresión y clasificación, donde las hipótesis hacen parte del conjunto de hipótesis lineales o hiperplanos.
- Ventajas: Los modelos lineales necesitan poca información para ser entrenados (con respecto a otros modelos más complejos).
- Desventajas: Los modelos lineales son limitados en su expresividad.
- Entonces qué hacer para atacar problemas de regresión y clasificación en los cuales una linea recta no sea suficiente?
- La respuesta: El uso de trasformaciones no lineales.

Regresión lineal con transformaciones no lineales

- El algoritmo de regresión lineal que vimos buscaba en el espacio de hipótesis lineales, el mejor hiperplano para ajustar un conjunto de puntos de entrenamiento.
- Recordemos la clase de hipótesis que usamos en la regresión lineal:

$$h_\theta(x) = \theta^t x = \theta_0 + \theta_1 x_1 + \ldots + \theta_n x_n$$

- Un error cuadrado medio (MSE) muy alto nos indicaría que la hipótesis lineal es insuficiente para describir los datos.
- Qué sucedería si considerásemos hipótesis de regresión no lineales?

Universidad Tecnológica de Bolívar
CARTAGENA DE INDIAS

Regresión lineal con transformaciones no lineales

- El algoritmo de regresión lineal que vimos buscaba en el espacio de hipótesis lineales, el mejor hiperplano para ajustar un conjunto de puntos de entrenamiento.
- Recordemos la clase de hipótesis que usamos en la regresión lineal:

$$h_\theta(x) = \theta^t x = \theta_0 + \theta_1 x_1 + ... + \theta_n x_n$$

- Un error cuadrado medio (MSE) muy alto nos indicaría que la hipótesis lineal es insuficiente para describir los datos.
- Qué sucedería si considerásemos hipótesis de regresión no lineales?
- Respuesta: En vez de un hiperplano, tendríamos una hipersuperficie genérica (cuadrática, cúbica, grado-n) de regresión.

Universidad Tecnológica de Bolívar
CARTAGENA DE INDIAS

Ejemplo hipótesis no lineal

- Tenemos los siguientes datos de entrenamiento:

x	$f(x)$
0,86	2,49
0,09	0,83
-0,85	-0,25
0,87	3,10
-0,44	0,87
-0,43	0,02
-1,10	-0,12
0,40	1,81
-0,96	-0,83
0,17	0,43

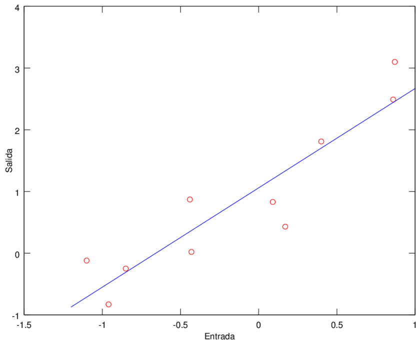
Universidad Tecnológica de Bolívar
CARTAGENA DE INDIAS

Ejemplo hipótesis no lineal

- Ahora, aplicando el algoritmo de regresión lineal obtenemos la siguiente hipótesis:

$$h(x) = 1,058 + 1,610x$$

$$\text{MSE} = 0,112$$

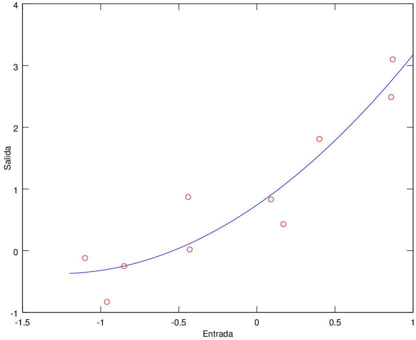


Ejemplo hipótesis no lineal

- Probemos con una hipótesis cuadrática:

$$h(x) = 0,739 + 1,747x + 0,686x^2$$

$$\text{MSE} = 0,078$$

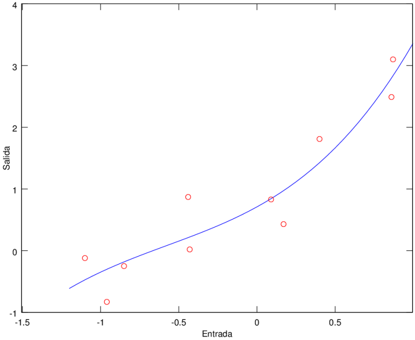


Ejemplo hipótesis no lineal

- Por qué no una hipótesis cúbica? :

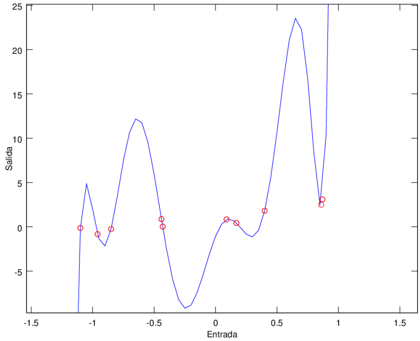
$$h(x) = 0,710 + 1,394x + 0,80x^2 + 0,464x^3$$

$$\mathrm{MSE} = 0,074$$



- Por qué no una hipótesis de noveno grado? :

$$MSE = 3,7502 * 10^{-25}$$



Entrenamiento Regresión con features polinomiales

- Vemos que entre más compleja sea la hipótesis, menor el MSE y más curva la gráfica.
- Sin embargo, no todo lo que brilla es oro. La disminución en el MSE puede ser a expensas de la habilidad de generalización del modelo...Una pregunta que aparece es, cual es la complejidad óptima? Eso lo responderemos más adelante en el curso.
- Por ahora concentrémonos en $\vec{\theta}$. Cómo entrenamos nuestros modelos no lineales?
- La respuesta es sorprendentemente simple: De la misma forma en que entrenamos nuestro modelo de regresión "lineal".
- Recordemos nuestra función de error (MSE):

$$J(\theta) = \frac{1}{N}\sum_{i=1}^{N}(h(x_i) - y_i)^2$$

- Recordemos además que el entrenamiento consiste en encontrar el vector θ que minimice $J(\theta)$.

- Observemos que la hipótesis

$$h(x) = 0,739 + 1,747x + 0,686x^2$$

a pesar de ser cuadrática en x, sigue siendo LINEAL en θ!

- Y la función de error depende de θ, por lo que la derivación de la solución en un solo paso, lleva al mismo resultado:

$$\theta = (X^T X)^{-1} X^T y$$

- Por la misma razón, el algoritmo de gradiente descendente funciona sin ningún cambio, más allá del cálculo de $h(x)$:

$$\theta j = \theta_j - \alpha \frac{1}{N} \sum_{i=1}^{N} (h(x_i) - y_i) x_{ij}$$

Universidad Tecnológica de Bolívar
CARTAGENA DE INDIAS

Entrenamiento Regresión con features polinomiales

- Para poder aplicar un modelo de regresión polinomial, debemos complementar nuestro conjunto de entrenamiento con los features del caso. Por ejemplo, si queremos usar el modelo cuadrático, nuestro conjunto de entrenamiento se convierte en:

x	x^2	$f(x)$
0,86	0,7396	2,49
0,09	0,0081	0,83
-0,85	0,7225	-0,25
0,87	0,7569	3,1
-0,44	0,1936	0,87
-0,43	0,1849	0,02
-1,1	1,21	-0,12
0,4	0,16	1,81
-0,96	0,9216	-0,83
0,17	0,0289	0,43

Universidad Tecnológica de Bolívar
CARTAGENA DE INDIAS

Entrenamiento Regresión con features polinomiales

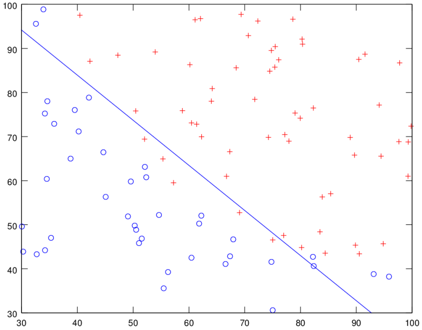
- Vemos entonces que no solo usamos los datos originales, sino transformaciones no lineales de las entradas, tambien llamadas features no lineales.
- El uso de dichas transformaciones incrementa sustancialemente la expresividad de nuestros modelos de regresión.
- De hecho, el uso de features no lineales es la clave de modelos de machine learning tales como las Redes Neuronales y las Máquinas de Vectores de Soporte (SVMs).

- De igual manera, es posible utilizar clasificadores con fronteras de decisión no lineales.
- La buena noticia sigue siendo que el algoritmo de entrenamiento es prácticamente igual, con la diferencia de que $h_\theta(x)$ se calcula de forma diferente.
- La regla de actualización del gradiente descendente aplicado a la regresión logística con features polinomiales sigue siendo:

$$\theta j = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h(x_i) - y_i) x_{ij}$$

Entrenamiento Regresión Logística con features polinomiales

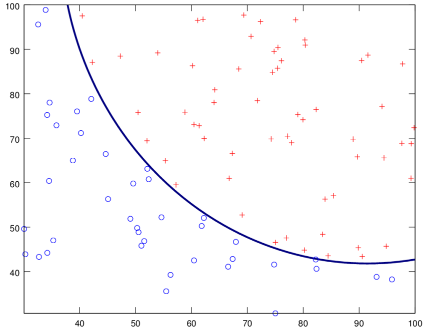
- Recordemos nuestro ejemplo acerca de la predicción de la probabilidad de admisión de un estudiante a la universidad.
- Habiendo entrenado el modelo, la frontera de decisión se ve de la siguiente manera:



$$\theta^t x = -25,1612 + 0,2062 x_1 + 0,2014 x_2$$

Entrenamiento Regresión Logística con features polinomiales

- Si utilizamos un modelo de segundo grado, este sería el resultado:

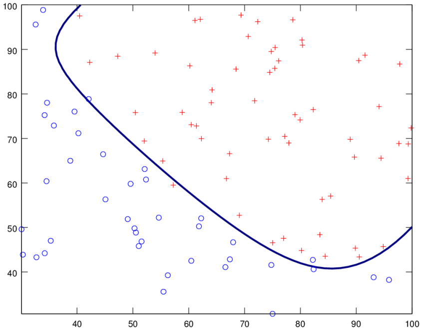


$$\theta^t x = -176,72 + 2,53x_1 + 1,768x_2 - 0,013x_1^2 - 0,007x_2^2$$

Universidad Tecnológica de Bolívar
CARTAGENA DE INDIAS

Entrenamiento Regresión Logística con features polinomiales

- Si utilizamos un modelo de tercer grado, este sería el resultado:



$$\theta^t x = -0,206 - 3,037x_1 - 1,368x_2 + 0,071x_1^2 + 0,0043x_2^2 + 0,00042x_1^3 + 0,00026x_3^3$$

Entrenamiento Regresión Logística con features polinomiales

- De igual manera que en la Regresión Lineal, entre más compleja sea la hipótesis de clasificación a utilizar, más complicada puede ser la frontera de decisión del clasificador.
- En este caso, un modelo cuadrático lleva el error de clasificación prácticamente a cero.
- La mala noticia es que la complejidad adicional viene con un costo de generalización.
- Existen resultados probabilísticos que señalan que a más complejidad del clasificador, mayor error de generalización puede ser esperado.

Muchas gracias!

Preguntas?

Universidad Tecnológica de Bolívar
CARTAGENA DE INDIAS