

# Análisis numérico

## Clase 16: Interpolación y aproximación (parte 2)

Joaquín Cavieres

Instituto de Estadística, Universidad de Valparaíso



# Outline

## 1 Interpolación

# Introducción

Generalmente, en problemas estadísticos o matemáticos, nosotros queremos evaluar una función en uno o más puntos. Sin embargo esto no es tan sencillo ya que surgen algunos inconvenientes tales como:

- Tiempos de ejecución del proceso de estimación demasiado costoso.
- Evaluación de funciones complejas
- Puede suceder que solo tengamos el valor de una función en un conjunto finito de puntos

# Introducción

Una estrategia adecuada y conveniente para este tipo de problemas podría ser reemplazar esa función en forma parcial por otra función más simple y que pueda ser evaluada eficientemente.

# Introducción

Una estrategia adecuada y conveniente para este tipo de problemas podría ser reemplazar esa función en forma parcial por otra función más simple y que pueda ser evaluada eficientemente.

Estas funciones "simples" casi siempre son elegidas entre **polinomiales**, **funciones trigonométricas**, **racionales**, etc.

# Interpolación

## Definición

La interpolación de una función  $f$ , mediante otra función  $g$ , consiste en, dado los siguientes puntos de datos:

- $n + 1$  puntos distintos  $x_0, \dots, x_n$
- $n + 1$  valores en aquellos puntos,  $f(x_0) = \omega_0, f(x_1) = \omega_1, \dots, f(x_n) = \omega_n$ ,

encontrar una función  $g$  tal que  $g(x_i) = \omega_i$  para un  $i = 0, 1, \dots, n$ .

Los puntos  $x_0, \dots, x_n$  son llamados **nodos** (o "knots") de interpolación y la función  $g$  es llamada **interpolante de  $f$**  en los puntos  $x_0, \dots, x_n$ .

# Interpolación

En primera instancia, vamos a considerar dos tipos de interpolantes:

- Interpolante polinomial

$$g(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n = \sum_{k=0}^n a_k x^k$$

- Interpolante polinomial por partes ("Piecewise polynomial")

$$g(x) = \begin{cases} p_1(x) & \text{si } x \in (x_0^*, x_1^*) \\ p_2(x) & \text{si } x \in (x_1^*, x_2^*) \\ \dots & \\ p_m(x) & \text{si } x \in (x_{m-1}^*, x_m^*) \end{cases}$$

donde  $x_0^*, x_m^*$  forman una partición del intervalo que contiene a los nodos de interpolación  $(x_0, x_n)$  y  $p_i(x)$  son los polinomiales.

# Interpolación polinomial

## 1. Interpolación Lineal

En este caso sencillo vamos a representar a dos puntos, por ejemplo el crecimiento de un niño, la cantidad de azucar en una bebida gaseosa o el número de computadores en las casas durante un año. Para este tipo reales, realizar dos mediciones es más factible que realizar mediciones continuas. Por lo tanto, las dos mediciones se toman en diferentes puntos en el tiempo, o para diferentes tamaños de bebida, o en diferentes valores de lo que sea que estemos midiendo.



# Interpolación polinomial

## 1. Interpolación Lineal

Así:

El punto en cual se determina nuestra medida es  $x$  y el valor observado es nuestra variable  $y$ . Por tanto, mediante un cálculo básico de álgebra, podemos hacer:

$$y = mx + b$$

que no es nada más que la ecuación de la recta con su intercepto y pendiente. Para encontrar  $m$  (pendiente) hacemos:

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

y su intercepto  $b = y_2 - m * x_2$

Ver ejemplo 1 en R

# Interpolación polinomial

## 2. Interpolación de Lagrange

Ya sabemos por definición que un interpolante polinomial tiene la siguiente expresión:

$$g(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

la cual satisface que  $(g(x_0) = \omega_0, \dots, g(x_n) = \omega_n)$ . Considerando los nodos de interpolación e igualándolos a  $\omega_i$ , entonces obtenemos una solución de un sistema lineal como:

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \omega_0 \\ \omega_1 \\ \vdots \\ \omega_n \end{bmatrix}$$

# Interpolación polinomial

## 2. Interpolación de Lagrange

La matriz de coeficientes:

$$\mathbb{A} = \begin{bmatrix} 1 & x_0 & x_0^2 & \dots & x_0^n \\ 1 & x_1 & x_1^2 & \dots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \dots & x_n^n \end{bmatrix}$$

es del tipo “*Vandermonde*” con determinante:

$$\det(A) = \prod_{0 \leq l < k \leq n} (x_k - x_l)$$

Como los nodos del interpolante son distintos, la  $\det(A) \neq 0$ , por tanto el sistema tiene solución única ( $g(x)$ ). Este polinomial se llamado el interpolante de Lagrange en los punos  $x_0, \dots, x_n$  realtivo a los valores  $\omega_0, \dots, \omega_n$ .

# Interpolación polinomial

## 2. Interpolación de Lagrange

La solución del sistema anterior puede volverse costoso computacionalmente ya que, si tenemos " $n$ " observaciones, necesitamos encontrar " $n$ " coeficientes.

Por lo anterior es que existen alternativas para calcular los interpolantes de Lagrange, entre ellos: [Los polinomiales fundamentales de Lagrange](#) y el método de [diferencias divididas](#)

# Interpolación polinomial

## Polinomios fundamentales de Lagrange

### Definición

Para un conjunto de datos  $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$ , sin  $x$  duplicados, existe una función  $g$  que evalúa esos puntos para así encontrar un único polinomial  $P(x)$  de grado  $\leq n$ . El polinomial esta dado por:

$$P(x) = f(x_0)L_{n,0}(x) + \dots + f(x_n)L_{n,n}(x) = \sum_{k=0}^n f(x_k)L_{n,k}(x)$$

donde  $k = 0, 1, \dots, n$  está dado por:

$$L_{n,k} = \frac{(x - x_0)(x - x_1) \cdots (x - x_{k-1})(x - x_{k+1}) \cdots (x - x_n)}{(x_k - x_0)(x_k - x_1) \cdots (x_k - x_{k-1})(x_k - x_{k+1}) \cdots (x_k - x_n)} = \prod_{\substack{i=0 \\ i \neq k}}^n \frac{(x - x_i)}{(x_k - x_i)}$$

Ver ejemplo 2 en R

# Interpolación polinomial por partes

# Interpolación polinomial por partes

Como vimos anteriormente, cuando el número de nodos del interpolante de Lagrange aumenta, puede suceder que a) se forman oscilaciones a partir el aumento del grado del polinomial y b) la aproximación no necesariamente es mejor, esto es, que todas las derivadas de la función interpolada deben estar uniformemente acotadas.

# Interpolación polinomial por partes

Como vimos anteriormente, cuando el número de nodos del interpolante de Lagrange aumenta, puede suceder que a) se forman oscilaciones a partir el aumento del grado del polinomial y b) la aproximación no necesariamente es mejor, esto es, que todas las derivadas de la función interpolada deben estar uniformemente acotadas.

Una manera de evitar esto es mediante **funciones polinomiales por partes**. Aunque algunas regularidades se pierden con esta técnica, nos aseguramos de que el error disminuya a medida que aumenta el número de nodos de interpolación.



# Interpolación polinomial por partes

El grado de polinomiales de una interpolación por partes (*linearwise polynomials*) es tal que los polinomiales, en este caso las líneas rectas, están determinadas por dos nodos consecutivos:

$$g(x) = \omega_i + (\omega_{i+1} - \omega_i) \frac{x - x_i}{x_{i+1} - x_i}, \quad \text{si } x \in [x_i, x_{i+1}]$$

para cada  $i = 0, \dots, n - 1$ . En este caso  $g$  es continua pero su derivada, en general, es discontinua en los nodos. Así, la interpolación mediante polinomiales por partes de orden 3 (**splines cúbicos**), son las más importantes en esta familia de interpolantes.

Ver ejemplo en R

# Splines

Un spline lineal con nodos en  $\nu_k$ ,  $k = 1, \dots, K$  es un polinomial linear por partes continuo en cada nodo. Además, un conjunto de splines lineales con nodos fijos no es un espacio vectorial. Un spline lineal puede ser descompuesto en basis de  $K + 2$  funciones de base como:

$$y = \sum_{m=1}^{K+2} \beta_m h_m(x) + \epsilon$$

Las funciones de base pueden elegirse como:

$$h_1(x) = 1$$

$$h_2(x) = x$$

$$h_{k+2}(x) = (x - \nu_k)_+, \quad k = 1, \dots, K$$

donde  $(\cdot)$  denota la parte positiva, por ejemplo,  $(x - \nu_k)_+ = x - \nu_k$  si  $x > \nu_k$  y  $(x - \nu_k)_+ = 0$  en otro caso.

# Cubic splines

Un cubic spline con nodos en  $\nu_k$ ,  $k = 1, \dots, K$  es un polinomial cúbico por partes con derivadas continuas de orden 2 en cada nodo. Además, un conjunto de splines cúbicos con nodos fijos no es un espacio vectorial. Un spline lineal puede ser descompuesto en basis de  $K + 4$

$$y = \sum_{m=1}^{K+4} \beta_m h_m(x) + \epsilon$$

# Cubic splines

Las funciones de base de poder (*power basis functions*) truncadas pueden elegirse como:

$$\begin{aligned}h_k(x) &= x^{k-1}, \quad k = 1, \dots, 4 \\h_{k+4}(x) &= (x - \nu_k)_+^4, \quad k = 1, \dots, K\end{aligned}$$

Ver ejemplo en R.

# Aproximación de mínimos cuadrados

# Aproximación de mínimos cuadrados

En algunas ocasiones la solución para  $Ax = b$  no tiene solución y la principal razón es que el sistema en  $A$  tiene **demasiadas ecuaciones**, por tanto, para una matriz de dimensión  $m \times n$  las columnas abarcan un pequeño espacio dimensional (de tamaño  $m$ ) por lo que el vector  $b$  está fuera de ese espacio dimensional.

# Aproximación de mínimos cuadrados

En algunas ocasiones la solución para  $Ax = b$  no tiene solución y la principal razón es que el sistema en  $A$  tiene **demasiadas ecuaciones**, por tanto, para una matriz de dimensión  $m \times n$  las columnas abarcan un pequeño espacio dimensional (de tamaño  $m$ ) por lo que el vector  $b$  esta fuera de ese espacio dimensional.

No siempre podemos obtener el error  $e = b - Ax$  llegando a 0.



# Aproximación de mínimos cuadrados

- Si  $e = 0$  entonces  $x$  es una solución exacta del sistema lineal  $Ax = b$ .
- Cuando  $e$  es un valor lo más pequeño posible, entonces  $\hat{x}$  es una **solución por mínimos cuadrados**

# Aproximación de mínimos cuadrados

- Si  $e = 0$  entonces  $x$  es una solución exacta del sistema lineal  $Ax = b$ .
- Cuando  $e$  es un valor lo más pequeño posible, entonces  $\hat{x}$  es una **solución por mínimos cuadrados**

El objetivo de la aproximación por mínimos cuadrados es calcular  $\hat{x}$

# Aproximación de mínimos cuadrados

En primera instancia vamos a mostrar una manera de resolver  $Ax = b$  cuando este no tiene solución mediante:

$$A^T A \hat{x} = A^T b$$

Ejemplo: encontrar la línea que pase por los puntos  $(0, 6)$ ,  $(1,0)$  y  $(2,0)$ .

# Aproximación de mínimos cuadrados

- $t = 0$  el primer punto de la línea  $b = C + Dt \Rightarrow$  si  $C + D * 0 = 6$
- $t = 1$  el segundo punto de la línea  $b = C + Dt \Rightarrow$  si  $C + D * 1 = 0$
- $t = 2$  el tercer punto de la línea  $b = C + Dt \Rightarrow$  si  $C + D * 2 = 0$

No existe una línea  $b = C + Dt$  que pase a través de esos puntos, en donde  $t = 0, 1, 2$  para los valores de  $b = 6, 0, 0$ . Así el sistema no tiene solución ya que  $b = (6, 0, 0)$  no es una combinación de las columnas  $(1, 1, 1)$  y de  $(0, 1, 2)$ .

$$A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \quad x = \begin{bmatrix} C \\ D \end{bmatrix} \quad b = \begin{bmatrix} 6 \\ 0 \\ 0 \end{bmatrix}$$

No tiene solución.

# Aproximación de mínimos cuadrados

## Minimización del error

¿Como podemos entonces hacer que  $e = b - Ax$  se lo más pequeño posible?

El "mejor"  $x$  es llamado  $\hat{x}$  y se puede encontrar mediante:

- Geometría
- Álgebra
- Cálculo

# Aproximación de mínimos cuadrados

## Minimización del error

Por Álgebra

Cada vector  $b$  es dividido en dos partes, una parte en el espacio de columnas  $p$  y otra parte perpendicular en el espacio nulo de  $A^T$  es  $e$ . Así,

$$Ax = b = p + e \Rightarrow \text{es posible}; \quad A\hat{x} = p \Rightarrow \text{tiene solución}$$

en donde la solución de  $A\hat{x} = p$  nos entrega el menor valor de error posible.

Finalmente, el largo para cualquier  $x$  es:  $\|Ax - b\|^2 = \|Ax - p\|^2 + \|e\|^2$ . La solución de mínimos cuadrados ( $\hat{x}$ ) hace que  $E = \|Ax - b\|^2$  sea lo más pequeño posible.

# Aproximación de mínimos cuadrados

## Minimización del error

Por Cálculo

Aquí el error  $E$  es minimizado a través de la suma de cuadrados de los errores  $e_1^2 + e_2^2 + e_3^2$ . Así:

$$E = \|Ax - b\|^2 = (C + D * 0 - 6)^2 + (C + D * 1)^2 + (C + D * 2)^2$$

Los elementos desconocidos son  $C$  y  $D$ , por lo tanto, debemos encontrar dos derivadas (ambas cero como mínimo). Estas son llamadas **derivadas parciales** ya que  $\frac{\partial E}{\partial C}$  trata como constante a  $D$  y  $\frac{\partial E}{\partial D}$  trata como constante a  $C$ .

# Aproximación de mínimos cuadrados

## Minimización del error

Por Cálculo

$$\frac{\partial E}{\partial C} = 2(C + D * 0 - 6) + 2(C + D * 1) + 2(C + D * 2) = 0$$

$$\frac{\partial E}{\partial D} = 2(C + D * 0 - 6)(0) + 2(C + D * 1)(1) + 2(C + D * 2)(2) = 0$$

En  $\frac{\partial E}{\partial D}$  se agregan 0, 1 y 2 por la definición de la regla de la cadena. Luego se cancelan todos los 2 y se ordenan los C's y los Ds:

- La derivada de C (cero):  $3C + 3D = 6$
- La derivada de D (cero):  $3C + 5D = 0$



# Aproximación de mínimos cuadrados

## Minimización del error

Finalmente,

- La derivada de C (cero):  $3C + 3D = 6$
- La derivada de D (cero):  $3C + 5D = 0$

es la matriz  $\begin{bmatrix} 3 & 3 \\ 3 & 5 \end{bmatrix}$  y que a su vez es  $A^T A$ .

# Aproximación de mínimos cuadrados

## Minimización del error

Finalmente,

- La derivada de C (cero):  $3C + 3D = 6$
- La derivada de D (cero):  $3C + 5D = 0$

es la matriz  $\begin{bmatrix} 3 & 3 \\ 3 & 5 \end{bmatrix}$  y que a su vez es  $A^T A$ .





Estas ecuaciones son idénticas a  $A^T A \hat{x} = A^T b$  en donde los "mejores"  $C$  y  $D$  son elementos de  $\hat{x}$ . Por consiguiente, las derivadas parciales de  $\|Ax - b\|^2$  son 0 cuando  $A^T A \hat{x} = A^T b$

# Aproximación de mínimos cuadrados

## Minimización del error

La solución es  $C = 5$  y  $D = -3$ . Además,  $b = 5 - 3t$  es la "mejor línea" que aproxima a los 3 puntos dados previamente.

Ver ejemplo en R

-  Burden, R. L., & Faires, J. D. (2011). Numerical analysis.
-  Howard, J. P. (2017). Computational Methods for Numerical Analysis with R. CRC Press.
-  Banerjee, S., & Roy, A. (2014). Linear algebra and matrix analysis for statistics. Crc Pr
-  Kiusalaas, J. (2013). Numerical methods in engineering with python (2nd ed.). New York: Cambridge University Press.