Programación: splines básicos

Objetivos. Programar el cálculo de B-splines de cualquier grado.

Requisitos. Función lineal (afín), fórmula recursiva para B-splines.

1. Numeración de los nodos y de los puntos donde queremos evaluar los B-splines. Vamos a suponer que t es el vector de los nodos:

$$t = [t_1, \ldots, t_n]^\top, \qquad t_1 \le t_2 \le \ldots \le t_n,$$

y u es el vector de los puntos en los cuales queremos evaluar las funciones:

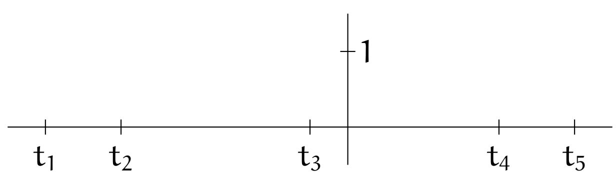
$$u = [u_1, \ldots, u_m]^\top.$$

Repaso de las fórmulas

2. B-splines de grado 0 (ejemplo). Para un sistema de cinco nodos t_1, t_2, t_3, t_4, t_5, hay cuatro splines básicos de grado 0:

$$B_{0,1},\ B_{0,2},\ B_{0,3},\ B_{0,4}.$$

En particular, $B_{0,2}$ es la función característica del intervalo $\big[\qquad , \qquad\big)$.
Muestre su gráfica:



Además, $B_{0,2}$ es la diferencia de dos funciones características:
de la función característica del rayo $\big[\qquad , +\infty\big)$
hay que restar la función característica del rayo $\big[\qquad , +\infty\big)$.

3. Fórmula para B-splines de grado 0 (repaso). Consideremos un sistema de n nodos: $t_1, \ldots, t_n$, $t_1 \le \ldots \le t_n$. Para cada $k \in \{1, \ldots, \qquad\}$,

$$B_{0,k}(u) = \begin{cases} 1, & u \in \big[\qquad , \qquad\big); \\ 0, & \text{en otro caso.} \end{cases}$$

La función $B_{0,k}$ se puede escribir como la diferencia de dos funciones características:
de la función característica del rayo $\big[\qquad +\infty\big)$
se resta la función característica del rayo $\big[\qquad +\infty\big)$.

4. Ejemplo. Supongamos que

$$t = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} -2 \\ 3 \\ 4 \end{bmatrix}, \qquad u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} -2 \\ 1 \\ 3 \\ 3.5 \end{bmatrix}.$$

Notemos que $B_{0,1}$ es la función característica del intervalo $[\quad,\quad)$,
y $B_{0,2}$ es la función característica del intervalo $[\quad,\quad)$.
Calculemos sus valores en los puntos $u_1, \ldots, u_4$:

$$B_{0,1}(u) = \begin{bmatrix} B_{0,1}(u_1) \\ B_{0,1}(u_2) \\ B_{0,1}(u_3) \\ B_{0,1}(u_4) \end{bmatrix} = \begin{bmatrix} 1 \\ \\ \\ 0 \end{bmatrix}, \qquad B_{0,2}(u) = \begin{bmatrix} B_{0,2}(u_1) \\ B_{0,2}(u_2) \\ B_{0,2}(u_3) \\ B_{0,2}(u_4) \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}.$$

5. Cálculo de los splines básicos usando operaciones matriciales. Con los datos del ejercicio anterior formemos dos matrices, repitiendo las entradas de t y u de la siguiente manera:

$$\begin{bmatrix} u_1 & & u_1 \\ u_2 & & \\ u_3 & & \\ u_4 & & \end{bmatrix}, \qquad \begin{bmatrix} t_1 & t_2 & \\ & & \\ & & \\ t_1 & & \end{bmatrix}.$$

Escribimos lo mismo con números dados y comparamos las matrices entrada por entrada (escribimos *verdadero* como 1, *falso* como 0):

$$\begin{bmatrix} -2 & & \\ 1 & & 1 \\ & & \\ & & \end{bmatrix} \geq \begin{bmatrix} -2 & 3 & \\ -2 & & 4 \\ & & \\ & & \end{bmatrix} = \begin{bmatrix} 1 & & \\ 1 & & 0 \\ & & \\ & & \end{bmatrix}.$$

Denotemos la matriz obtenida por C y restamos sus columnas consecutivas de la siguiente manera:

$$C_{*,1} - C_{*,2} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}, \qquad C_{*,2} - C_{*,3} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}.$$

Compare las respuestas obtenidas con las respuestas del Ejercicio 4.

6. Cálculo de los splines básicos usando operaciones matriciales en el lenguaje Matlab. Ejecute los siguiente comandos en Matlab o en GNU Octave para recordar el sentido de las funciones `repmat` y `diff`:

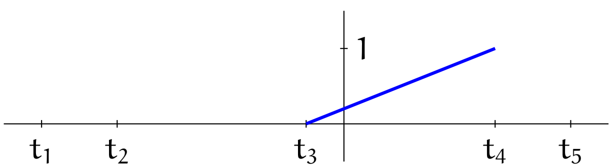
```
v = [1.41; 2.72; 3.14]
repmat(v, 1, 5)
repmat(v', 3, 1)
A = floor(10 * rand(5, 3))
diff(A, 1, 1)
diff(A, 1, 2)
```

Complete los siguiente comandos para repetir los cálculos del Ejercicio 5:

```
t = [-2; 3; 4]
u = [-2; 1; 3; 3.5]
uext = repmat(u, ???, ???)
text = repmat(t', ???, ???)
c = uext >= text
- diff(c, 1, ???)
```

7. Ejemplos de cocientes. Los siguientes dibujos muestran algunos cocientes. Determine en qué puntos estos cocientes toman los valores 0 y 1:

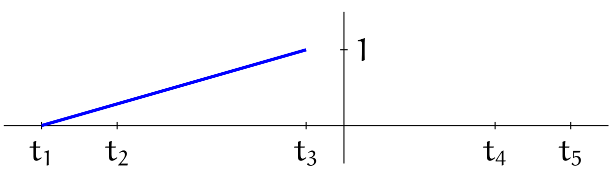
$q_{1,3}$



$$q_{1,3}(t_3) = 0$$

$$q_{1,3}(\qquad) = 1$$

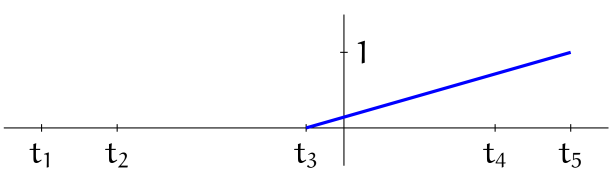
$q_{2,1}$



$$q_{2,1}(\qquad) = 0$$

$$q_{2,1}(\qquad) = 1$$

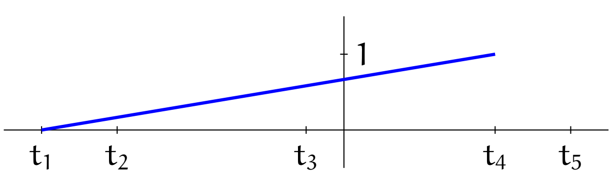
$q_{2,3}$



$$q_{2,3}(\qquad) = 0$$

$$q_{2,3}(\qquad) = 1$$

$q_{3,1}$



$$q_{3,1}(\qquad) = 0$$

$$q_{3,1}(\qquad) = 1$$

8. Fórmula general para los cocientes (repaso). Generalizando los ejemplos del ejercicio anterior, vemos que

$$q_{p,k}(\boxed{}) = 0, \qquad q_{p,k}(\boxed{}) = 1.$$

La función $q_{p,k}$ es un polinomio de grado 1 y se determina de manera única por sus valores en dos puntos.

$$q_{p,k}(u) = \frac{u - \boxed{}}{\boxed{} - \boxed{}}. \tag{1}$$

9. Cálculo de los cocientes con operaciones matriciales ($n = 5,\ p = 2,\ m = 4$). Supongamos que $n = 5$, $p = 2$, y $m = 4$. Tenemos que calcular los valores de los cocientes $q_{2,1}$, $q_{2,2}$, $q_{2,3}$ en los puntos u_1, u_2, u_3, u_4. Escribimos la fórmula (1) para estos casos particulares ($p = \boxed{}$, $1 \leq k \leq 5 - \boxed{}$):

$$q_{2,1}(u) = \frac{u - \boxed{}}{\boxed{} - \boxed{}}, \qquad q_{2,2}(u) = \frac{u - \boxed{}}{\boxed{} - \boxed{}}, \qquad q_{2,3}(u) = \frac{u - \boxed{}}{\boxed{} - \boxed{}}.$$

Sustituimos los puntos u_1, u_2, u_3, u_4 en lugar de u:

$$\begin{bmatrix} q_{2,1}(u_1) & q_{2,2}(u_1) & q_{2,3}(u_1) \\ q_{2,1}(u_2) & q_{2,2}(u_2) & q_{2,3}(u_2) \\ q_{2,1}(u_3) & q_{2,2}(u_3) & q_{2,3}(u_3) \\ q_{2,1}(u_4) & q_{2,2}(u_4) & q_{2,3}(u_4) \end{bmatrix} = \begin{bmatrix} \underline{\hspace{2cm}} & \underline{\hspace{2cm}} & \underline{\hspace{2cm}} \\ \underline{\hspace{2cm}} & \underline{\hspace{2cm}} & \underline{\hspace{2cm}} \\ \underline{\hspace{2cm}} & \underline{\hspace{2cm}} & \underline{\hspace{2cm}} \\ \underline{\hspace{2cm}} & \underline{\hspace{2cm}} & \underline{\hspace{2cm}} \end{bmatrix}.$$

Vamos a calcular aparte los numeradores y los denominadores, luego podremos calcular los cocientes usando la división por componentes:

```
Q = Qnumer ./ Qdenom;
```

10. Los numeradores de los cocientes ($n = 5,\ p = 2,\ m = 4$). Continuamos el Ejercicio 9. Escribimos la matriz de los numeradores y la representamos como cierta diferencia de dos matrices:

$$\begin{bmatrix} u_1 - t_1 & u_1 - t_2 & u_1 - t_3 \\ u_2 - t_1 & \quad & \quad \\ u_3 - t_1 & \quad & \quad \\ u_4 - t_1 & \quad & \quad \end{bmatrix} = \begin{bmatrix} \quad & \quad & \quad \\ \quad & \quad & \quad \\ \quad & \quad & \quad \\ \quad & \quad & \quad \end{bmatrix} - \begin{bmatrix} \quad & \quad & \quad \\ \quad & \quad & \quad \\ \quad & \quad & \quad \\ \quad & \quad & \quad \end{bmatrix}.$$

Las últimas matrices se obtienen al repetir el mismo renglón o la misma columna. Usamos el convenio que los vectores t y u están dados como columnas. Si queremos escribir un vector como renglón, entonces tenemos que transponerlo. Escribimos los comandos correspondientes en el lenguaje Matlab:

```
tfirst = t(1 : 3);
Qnumer = repmat(u, ???, ???) - repmat(tfirst', ???, ???);
```

11. Los denominadores de los cocientes ($n = 5,\ p = 2,\ m = 4$). Ahora ponemos nuestra atención a los denominadores de la matriz que surgió en el Ejercicio 9. Escribimos la matriz de los denominadores:

$$\begin{bmatrix} \quad & \quad & \quad \\ \quad & \quad & \quad \\ \quad & \quad & \quad \\ \quad & \quad & \quad \end{bmatrix}. \tag{2}$$

Se ve que se repite varias veces la misma $\underbrace{\qquad\qquad}_{\text{fila/columna}}$.

La escribimos aparte como un vector-columna y descomponemos en la diferencia de dos columnas:

$$\texttt{tdif} = \begin{bmatrix} \quad - \quad \\ \quad - \quad \\ \quad - \quad \end{bmatrix} = \begin{bmatrix} \quad \\ \quad \\ \quad \end{bmatrix} - \begin{bmatrix} \quad \\ \quad \\ \quad \end{bmatrix}.$$

Las últimas columnas son ciertos fragmentos del vector original t.
Escribimos comandos que construyen la matriz (2):

```
tfirst = t(??? : ???);
tlast = t(??? : ???);
tdif = ??? - ???;
Qdenom = repmat(tdif', ???, ???);
```

12. Matriz de los cocientes, con n, p, m generales. Ahora regresamos a la situación general:

$$t = \begin{bmatrix} t_1 \\ \vdots \\ t_n \end{bmatrix}, \qquad u = \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}, \qquad 1 \le p \le n-2.$$

Las funciones $q_{p,k}$ se definen para cada $k \in \{1, \ldots, \boxed{}\}$.
Recordamos la definición de estas funciones:

$$q_{p,k}(u) = \frac{u - \boxed{}}{\boxed{} - \boxed{}}. \tag{3}$$

Formamos la matriz de los valores de estas funciones en los puntos $u_1, \ldots, u_m$:

$$Q = \begin{bmatrix} q_{p,1}(u_1) & \ldots & \boxed{} \\ \vdots & \ddots & \vdots \\ \boxed{} & \ldots & \boxed{} \end{bmatrix} = \begin{bmatrix} \dfrac{u_1 - t_1}{t_{p+1} - t_1} & \ldots & \dfrac{\boxed{}}{\boxed{}} \\ \vdots & \ddots & \vdots \\ \dfrac{\boxed{}}{\boxed{}} & \ldots & \dfrac{\boxed{}}{\boxed{}} \end{bmatrix}.$$

Vamos a calcular por separado la matriz de los numeradores `Qnumer` y luego la matriz de los denominadores `Qdenom`, entonces la matriz `Q` se podrá calcular con el comando

```
Q = Qnumer ??? Qdenom;
```

13. Matriz de los numeradores, con n, p, m generales. Escribimos aparte la matriz de los numeradores `Qnumer` y la representamos como la diferencia de ciertas dos matrices:

$$Q_{\text{numer}} = \begin{bmatrix} u_1 - t_1 & \ldots & \boxed{} \\ \vdots & \ddots & \vdots \\ \boxed{} & \ldots & \boxed{} \end{bmatrix} = \begin{bmatrix} u_1 & \ldots & \boxed{} \\ \vdots & \ddots & \vdots \\ \boxed{} & \ldots & \boxed{} \end{bmatrix} - \begin{bmatrix} t_1 & \ldots & \boxed{} \\ \vdots & \ddots & \vdots \\ \boxed{} & \ldots & \boxed{} \end{bmatrix}.$$

Una de estas matrices se obtiene al repetir $\boxed{}$ veces la columna u,
y la otra se obtiene al repetir $\boxed{}$ veces un fragmento del vector t, escrito como renglón:

```
tfirst = t(??? : ???);
Qnumer = repmat(u, ???, ???) - repmat(tfirst', ???, ???);
```

14. Matriz de los denominadores, con n, p, m generales. Escribimos aparte la matriz de los denominadores `Qdenom`:

$$Q_{\text{denom}} = \begin{bmatrix} t_{p+1} - t_1 & \ldots & \boxed{} \\ \vdots & \ddots & \vdots \\ \boxed{} & \ldots & \boxed{} \end{bmatrix}.$$

Se ve que todos los renglones de esta matriz son iguales entre si. Escribimos este vector repetido como una columna y la descomponemos en la diferencia de dos columnas:

$$\texttt{tdif} = \begin{bmatrix} t_{p+1} - t_1 \\ \boxed{} \\ \vdots \\ \boxed{} \end{bmatrix} = \begin{bmatrix} t_{p+1} \\ \boxed{} \\ \vdots \\ \boxed{} \end{bmatrix} - \begin{bmatrix} t_1 \\ \boxed{} \\ \vdots \\ \boxed{} \end{bmatrix}.$$

El código correspondiente:

```
tfirst = t(??? : ???);
tlast = t(??? : ???);
tdif = ??? - ???;
Qdenom = repmat(tdif', ???, ???);
```

15. Fórmula recursiva de Cox–de Boor (ejemplo). En la notación que utilizamos,

$$B_{2,k}(u) = q_{2,k}(u)\, B_{1,k}(u) + (1 - q_{p,k+1}(u))\, B_{1,k+1}(u).$$

El subíndice k corre de 1 a $n - 3$, por eso $k + 1$ corre de $\boxed{}$ a $\boxed{}$.

16. Fórmula recursiva de Cox–de Boor (repaso).

$$B_{p,k}(u) = q_{p,k}(u) \boxed{} + (1 - q_{p,k+1}(u)) \boxed{}. \tag{4}$$

El subíndice k en la fórmula (4) corre de $\boxed{}$ a $\boxed{}$,
por eso $k + 1$ corre de $\boxed{}$ a $\boxed{}$.

17. Fórmula recursiva de Cox–de Boor en términos de los productos. Pongamos

$$P_{p,k}(u) = q_{p,k}(u) B_{p-1,k}(u).$$

Entonces la fórmula (4) se puede escribir así:

$$B_{p,k}(u) = \underbrace{\boxed{}}_{P_{???,???}(u)} + B_{p-1,k+1}(u) - \underbrace{\boxed{}}_{P_{???,???}(u)}. \tag{5}$$

18. Un truco para dividir entre cero.

```
a = [3; 4; 5; 6; 7];
b = [-2; -1; 0; 1; 2];
ind = abs(b) > 1.0E-12;
c = zeros(5, 1);
c(ind) = a(ind) ./ b(ind);
```

19. Cálculo de B-splines del grado dado d.

```
function [V] = bsplines(t, u, d),
   m = length(u); n = length(t);
   C = repmat(???, ???, ???) >= repmat(???, ???, ???);
   V = - diff(C, 1, ???);
   for p = 1 : d,
      tfirst = ???;
      tlast = ???;
      tdif = ???;
      Qnumer = ???;
      Qdenom = ???;
      Q = zeros(???, ???);
      ind = abs(Qdenom) > 1.0E-12;
      Q(ind) = Qnumer(ind) ./ Qdenom(ind);
      P = Q .* V;
      V = P(:, 1 : ???) + V(:, 2 : ???) - P(:, 2 : ???);
   end
end
```

20. Prueba.

```
function [] = testsplines(),
   t = [-2; 1; 2; 5; 6; 7; 9];
   u = linspace(min(t), max(t), 50);
   V = bsplines(t, u, 3);
   plot(u, V, '*');
end
```