

Part I

Interpolation polynomiale

1 Introduction

Objectif :

Étant donné $n+1$ points de coordonnées $x_i \in \mathbb{R}^2, i \in \{0, \dots, n\}$
déterminer P un (le ?) polynôme vérifiant

$$P(x_i) = y_i, \quad \forall i \in \{0, \dots, n\} \text{ où } x_i = (x_i, y_i)$$

1.1 Définition

On appelle problème d'interpolation de Lagrange le
problème suivant : trouver P un polynôme vérifiant

$$P(x_i) = y_i, \quad \forall i \in \{0, \dots, n\}$$

Considérons $P \in \mathbb{P}_n$ l'espace des polynômes de degré n et
notons par $(\varphi_j)_{j=0, \dots, n}$ une base. On a alors :

$$P(x_i) = y_i \Leftrightarrow \sum_{j=0}^n \alpha_j \varphi_j(x_i) = y_i$$

où les inconnues sont les coefficients α_j .

$$P(x_i) = y_i \Leftrightarrow \begin{bmatrix} \varphi_0(x_0) & \varphi_1(x_0) & \cdots & \varphi_n(x_0) \\ \varphi_0(x_1) & \varphi_1(x_1) & \cdots & \varphi_n(x_1) \\ \vdots & \vdots & & \vdots \\ \varphi_0(x_n) & \varphi_1(x_n) & \cdots & \varphi_n(x_n) \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

Remarque :

On peut dorénavant noter que la matrice n'est pas
inversible si on a $x_i = x_j, j \neq i$

1.2 Proposition

Le problème d'interpolation de Lagrange admet une unique solutionssi la matrice de Gram G définie par $G_{ij} = \varphi_j(x_i)$ est inversible.

1.3 Théorème

Si les $(\varphi_j)_{j=0,\dots,n}$ forment une base de \mathbb{P}_n , et $x_i \neq x_j, \forall j \neq i$ alors la matrice de Gram G est inversible, et donc le problème d'interpolation admet une unique solution.

Preuve

Prouvons que $\text{Ker}(G) = \{0\}$.

Soit $\alpha \in \text{Ker}(G)$ alors

$$\begin{aligned} G\alpha = 0 &\iff \begin{bmatrix} \varphi_0(x_0) & \dots & \varphi_n(x_0) \\ \vdots & \ddots & \vdots \\ \varphi_0(x_n) & \dots & \varphi_n(x_n) \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \\ &\iff \forall i \in \{0 \dots n\} \sum_{j=0}^n \varphi_j(x_i) \alpha_j = 0 \end{aligned}$$

Posons $Q(x) = \sum_{j=0}^n \varphi_j(x) \alpha_j$

On a $Q(x_i) = 0 \forall i \in \{0 \dots n\}$ Ce qui implique que Q possède $n+1$ racines.

Or $Q \in P[X^n]$, on déduit que $Q = 0$.

D'où $\sum_{j=0}^n \varphi_j(x) \alpha_j = 0 \forall x$

Comme les φ_j forment une famille libre, on déduit que $\alpha_j = 0 \forall j \in \{0 \dots n\} \iff \text{Ker}(G) = 0$

Remarque autant d'équation que d'inconnues n'implique pas forcément que G soit inversible.

② Base canonique : $\mathbb{P}_n = \text{vect}\{1, x, x^2, \dots, x^n\}$

③ Base de Lagrange :

Idée : Choisir les $(\varphi_j)_{j=0, \dots, n}$ t.q. La matrice de Gram

$$G = \begin{bmatrix} \varphi_0(x_0) & \varphi_1(x_0) & \cdots & \varphi_n(x_0) \\ \varphi_0(x_1) & \varphi_1(x_1) & \cdots & \varphi_n(x_1) \\ \vdots & \vdots & & \vdots \\ \varphi_0(x_n) & \varphi_1(x_n) & \cdots & \varphi_n(x_n) \end{bmatrix}$$

soit la matrice identité.

Autrement dit, on cherche φ_j t.q. $\varphi_j(x_i) = \delta_{ij}$.

1.4 Proposition

La famille de fonction $(L_i^n)_{i=0, \dots, n}$ définie par

$$L_i^n(x) := \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)}$$

forme une base de \mathbb{P}_n et vérifie $L_j^n(x_i) = \delta_{ij}$.

Preuve

$$\text{On a } L_i^n(x_k) = \frac{\prod_{j \neq i} (x_k - x_j)}{\prod_{j \neq i} (x_i - x_j)}$$

- Si $k = i$

$$L_i^n(x_i) = \frac{\prod_{j \neq i} (x_i - x_j)}{\prod_{j \neq i} (x_i - x_j)} = 1$$

- Si $k \neq i$

$$L_i^n(x_k) = \frac{\overbrace{\prod_{j \neq i} (x_k - x_j)}^{0 \text{ quand } j=k}}{\prod_{j \neq i} (x_i - x_j)} = 0$$

Montrons que $(L_i^n)_{i=0..n}$ est une famille libre.

Posons $Q(x) = \sum_{j=0}^n \alpha_j L_j^n(x)$

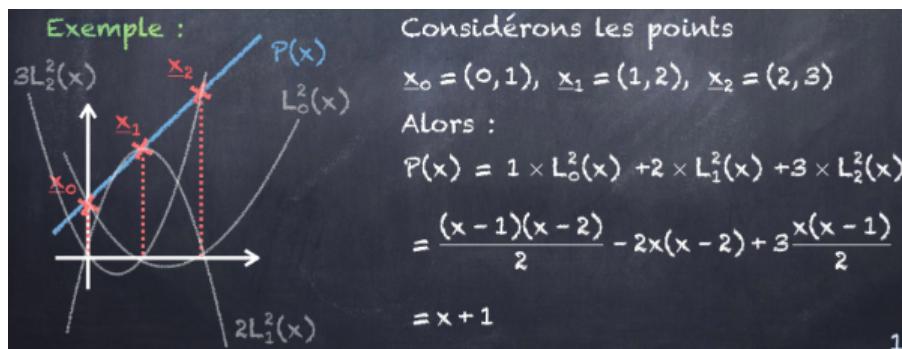
Alors en $x = x_i$, $\sum_{j=0}^n \alpha_j L_j^n(x_i) = 0$

Donc $\alpha_j = 0 \forall j \in \{0..n\}$

$(L_i^n)_i$ est donc une famille libre, c'est une base de $P[X^n]$

Remarque Dans la base de Lagrange, l'expression du polynôme d'interpolation est direct

$$P(x) = \sum_{j=0}^n L_j^n(x) y_j$$



2 Évaluation du polynôme d'interpolation

2.1 Schéma de Neville-Aiken

La base de Lagrange nous permet d'obtenir l'expression du polynôme d'interpolation facilement :

$$P(x) = \sum_{j=0}^n y_j L_j^n(x) \quad \text{où} \quad L_i^n(x) := \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)}$$

Cependant, cette expression n'est pas pratique pour évaluer P en un point $x \neq x_i$ lorsque n devient grand.

Notons par $T_k^i \in \mathbb{P}_k$ le polynôme d'interpolation vérifiant :

$$T_k^i(x_j) = y_j, \quad \forall j \in \{i, \dots, i+k\}$$

Remarque : On aura en particulier $T_0^i(x) = y_j$

Notons par $T_k^i \in \mathbb{P}_k$ le polynôme d'interpolation vérifiant :

$$T_k^i(x_j) = y_j, \quad \forall j \in \{i, \dots, i+k\}$$

2.1 Proposition

Les $T_k^i \in \mathbb{P}_k$ vérifient la relation suivante :

$$T_{k+1}^i(x) = \frac{T_k^i(x)(x_{i+k+1} - x) - T_k^{i+1}(x)(x_i - x)}{(x_{i+k+1} - x_i)}$$

Preuve

Par définition, on a $T_{k+1}^i \in P[X^{k+1}]$ qui vérifie $T_{k+1}^i(x_j) = y_j \quad \forall j \in \{i, \dots, i+k+1\}$

$$\text{Posons } Q(x) = \overbrace{T_k^i(x)}^{\deg k} \overbrace{(x_{i+k+1} - x)}^{\deg 1} - \overbrace{T_k^{i+1}(x)}^{\deg k} \overbrace{(x_i - x)}^{\deg 1}$$

Donc $\deg Q = k+1$

Prouvons que $\forall j \in \{i, \dots, i+k+1\}$ On a $Q(x_j) = y_j$ et on aura ainsi

$$Q(x) = T_{k+1}^i(x)$$

- Pour $x = x_i$

$$Q(x_i) = \frac{\overbrace{T_k^i(x_i)(x_{i+k+1} - x_i)}^{y_i} - T_k^{i+1}(x_i) \overbrace{(x_i - x_i)}^0}{(x_{i+k+1} - x_i)} = y_i$$

- Pour $x = x_{i+k+1}$

$$Q(x_{i+k+1}) = \frac{T_k^i(x_{i+k+1}) \overbrace{(x_{i+k+1} - x_{i+k+1})}^0 - T_k^{i+1}(x_{i+k+1}) \overbrace{(x_i - x_{i+k+1})}^{y_{i+k+1}}}{(x_{i+k+1} - x_i)} = y_{i+k+1}$$

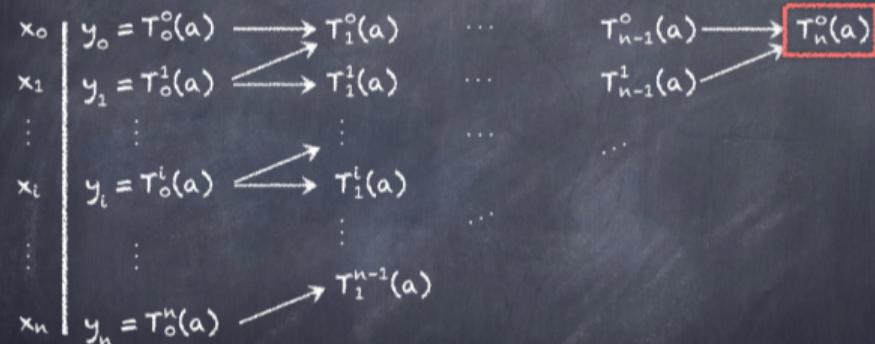
- Pour $x = x_j$

$$Q(x_j) = \frac{\overbrace{T_k^i(x_j)(x_{i+k+1} - x_j)}^{y_j} - T_k^{i+1}(x_j)(x_i - x_j)}{(x_{i+k+1} - x_i)} = y_j$$

Remarque La relation indique que pour évaluer T_{k+1}^i en x , on a besoin de T_k^i et T_k^{i+1}

De plus, on notera que $T_n^0(x) = P(x)$

Algorithme d'évaluation de P en $x=a$ (à la main) :



Remarque :

L'ordre des points x_i ne compte pas.

Algorithme d'évaluation de P en $x=a$:

```

Pour k = 0 à n-1
  Pour i = 0 à n-k-1
    T_{k+1}[i] ←  $\frac{T_k[i](x[i+k+1]-a) - T_k[i+1](x[i]-a)}{(x[i+k+1] - x[i])}$ 
    T_k ← T_{k+1}
  
```

Complexité : ~~$O(n^2/2)$~~ $O(5n^2/2)$

À chaque itération i , on effectue 4 différences, 2 produits et une division, soit au total 7 opérations. Au total, on a donc

$$\sum_{k=0}^{n-1} \sum_{i=0}^{n-k} 7 = 7 \sum_{k=0}^{n-1} (n-k) = 7 \frac{n(n+1)}{2} \text{ opérations}$$

19

2.2 Différence divisé et Horner

Base de Newton :

Idée : Choisir les $(\varphi_j)_{j=0, \dots, n}$ t.q. la matrice de Gram

$$G = \begin{bmatrix} \varphi_0(x_0) & \varphi_1(x_0) & \cdots & \varphi_n(x_0) \\ \varphi_0(x_1) & \varphi_1(x_1) & \cdots & \varphi_n(x_1) \\ \vdots & \vdots & & \vdots \\ \varphi_0(x_n) & \varphi_1(x_n) & \cdots & \varphi_n(x_n) \end{bmatrix}$$

triangulaire (inférieur), i.e. $\varphi_j(x_i) = 0 \quad \forall j > i$

2.2 Proposition

La famille de fonction $(N_i)_{i=\{0, \dots, n\}}$ définie par

$$N_0(x) = 1 \text{ et } N_i(x) = \prod_{j=0}^{i-1} (x - x_j)$$

forme une base de P_n et vérifie $N_j(x_i) = 0 \quad \forall j > i$.

Preuve

Montrons que

1. $N_i(x_k) = 0 \quad \forall k \leq i - 1$
2. La famille des $(N_i)_i$ forme une base de $P[X^n]$

On a $\forall k \leq i - 1$

$$N_i(x_k) = \prod_{j=0}^{i-1} \underbrace{(x_k - x_j)}_{0 \text{ quand } j=k} = 0$$

De plus la famille de fonction $(N_i)_i$ possède $n+1$ éléments et $\dim P[X^n] = n+1$. Montrons que c'est une famille libre.

Posons $Q(x) = \sum_{i=0}^n N_i(x)\alpha_i$

Montrons que $Q(x) = 0 \quad \forall x \iff \alpha_i = 0 \quad \forall i$

Le sens indirecte est évident, prouvons le sens direct.

Si on a $\sum_{i=0}^n N_i(x)\alpha_i = 0 \quad \forall x$ alors en $x = x_0$, $\sum_{i=0}^n N_i(x_0)\alpha_i = 0 \iff \alpha_0 = 0$

Ensuite en $x = x_1$, $\sum_{i=0}^n N_i(x_1)\alpha_i = \alpha_1 \underbrace{N_1(x_1)}_{\neq 0} = 0 \iff \alpha_1 = 0$

Et on continue ainsi pour déduire que $\alpha_i = 0 \forall i$

Base de Newton : $P_n = \text{vect}\{N_0, N_1, \dots, N_n\}$

La décomposition du polynôme dans cette base pourrait être obtenu par résolution du système linéaire $O(8n^2)$.

$$\begin{bmatrix} N_0(x_0) & 0 & \cdots & 0 \\ N_0(x_1) & N_1(x_1) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ N_0(x_n) & N_1(x_n) & \cdots & N_n(x_n) \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

Mais on peut faire mieux ! Posons :

$$f[x_i, \dots, x_{i+j+1}] = \frac{f[x_{i+1}, \dots, x_{i+j+1}] - f[x_i, \dots, x_{i+j}]}{x_{i+j+1} - x_i}, \quad f[x_i] = f(x_i)$$

2.2 Proposition

Le polynôme d'interpolation est donné par

$$P(x) = f[x_0]N_0(x) + f[x_0, x_1]N_1(x) + \dots + f[x_0, x_1, \dots, x_n]N_n(x)$$

Preuve

Posons l'hypothèse de récurrence :

$$(H_k) : T_k^i = f[x_i]N_0^i(x) + f[x_i, x_{i+1}]N_1^i(x) + \dots + f[x_i, \dots, x_{i+k}]N_k^i(x)$$

où $N_0^i(x) = 1$ et $N_j^i(x) = \prod_{l=0}^{j-1} (x - x_{l+i}) \forall i \in \{0..n-k\}$

Pour $k = 0$

On doit vérifier que $\forall i \in \{0..n\} T_0^i(x) = f[x_i]N_0^i(x) = f[x_i] = y_i$ vrai au rang 0

Supposons (H_k) vrai, montrons le rang $k+1$

Soit $i \in \{0, \dots, n-k-1\}$

On veut prouver que

$$T_k^i = f[x_i]N_0^i(x) + f[x_i, x_{i+1}]N_1^i(x) + \dots + f[x_i, \dots, x_{i+k+1}]N_{k+1}^i(x)$$

$$\text{Posons } T_{k+1}^i(x) = \sum_{j=0}^{k+1} \alpha_j N_j^i(x)$$

$$T_k^i(x) = \sum_{j=0}^k f[x_i, \dots, x_{i+j}]N_j^i(x)$$

en évaluant en $x = x_i$ on obtient

$$T_k^i(x_i) = f[x_i]N_0^i(x_i) = y_i$$

$$\text{De plus } T_{k+1}^i(x_i) = \alpha_0 N_0^i(x_i) = y_i$$

On déduit que $\alpha_0 = f[x_i]$

De même en $x = x_{i+1}$ on a

$$T_k^i(x_{i+1}) = f[x_i]N_0^i(x_{i+1}) + f[x_i, x_{i+1}]N_1^i(x_{i+1}) = y_{i+1}$$

$$\text{De plus } T_{k+1}^i(x_{i+1}) = f[x_i]N_0^i(x_i) + \alpha_1 N_1^i(x_{i+1}) = y_{i+1}$$

$$\text{Donc } \alpha_1 = f[x_i, x_{i+1}]$$

Et on continue ainsi pour les autres α_j jusqu'à $j = i + k$

Il reste à prouver que $\alpha_{k+1} = f[x_i, \dots, x_{i+k+1}]$

Or

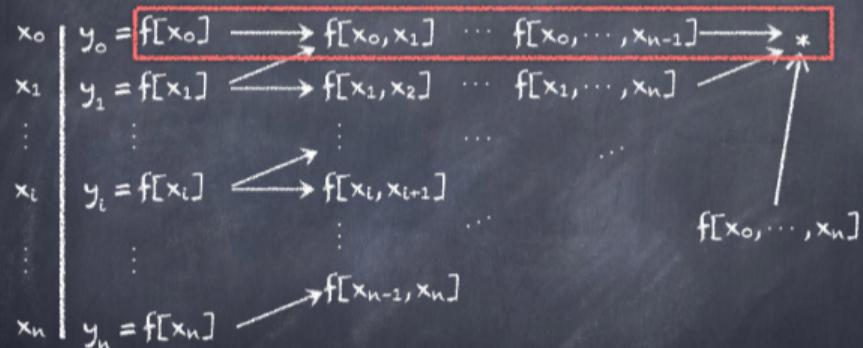
$$T_{k+1}^i(x) = \frac{T_k^i(x)(x_{i+k+1} - x) - T_k^{i+1}(x)(x_i - x)}{(x_{i+k+1} - x_i)}$$

$$\begin{aligned} &\iff \alpha_{k+1} N_{k+1}^i(x) + T_k^i(x) \\ &= \frac{\sum_{j=0}^k f[x_i, \dots, x_{i+j}] N_j^i(x)(x_{i+k+1} - x) - \sum_{j=0}^k f[x_i, \dots, x_{i+j+1}] N_j^{i+1}(x)(x_i - x)}{(x_{i+k+1} - x_i)} \end{aligned}$$

En identifiant les termes de degrés $k+1$ on a

$$\begin{aligned} \alpha_{k+1} &= \frac{f[x_i, \dots, x_{i+k}] \times (-1) - f[x_i, \dots, x_{i+k+1}] \times (-1)}{x_{i+k+1} - x_i} \\ &= f[x_i, \dots, x_{i+k+1}] \end{aligned}$$

Algorithme des différences divisées :



Remarque :

L'ordre des points x_i ne compte pas.

Algorithme des différences divisées :

```

Pour k = 0 à n-1
  Pour i = 0 à n-k-1
    F_{k+1}[i] ←  $\frac{F_k[i+1] - F_k[i]}{(x[i+k+1] - x[i])}$ 
    a_{k+1} ← F_{k+1}[0]
    F_k ← F_{k+1}
  
```

Complexité : $O(3n^2/2)$

À chaque itération i , on effectue 2 différences et une division, soit au total 3 opérations. Au total, on a donc

$$\sum_{k=0}^{n-1} \sum_{i=0}^{n-k} 3 = 3 \sum_{k=0}^{n-1} (n-k) = 3 \frac{n(n+1)}{2} \text{ opérations}$$

24

Voyons maintenant comment évaluer notre polynôme

$$P(x) = f[x_0]N_0(x) + f[x_0, x_1]N_1(x) + \dots + f[x_0, x_1, \dots, x_n]N_n(x)$$

exprimé dans la base de Newton.

2.3 Proposition (Algorithme de Horner)

En posant $q_{i-1} = f[x_0, \dots, x_i] + (a - x_i)q_i \quad \forall i \in \{0, n\}$ et $q_n = 0$,
on a $P(a) = q_0$.

Algorithme de Horner :

Pour $i = 0$ à n

$$q_{i-1} \leftarrow F[i] + (a - x_i)q_i$$

Complexité : $O(3n)$

Avec cette approche, pour évaluer P en $x=a$, il faut donc :

- ➊ calculer la décomposition dans la base de Newton
(Différences divisées) $O(3n^2/2)$
 - ➋ appliquer l'algorithme de Horner $O(3n)$
- $\left. \begin{matrix} O(3n^2/2) \\ O(3n) \end{matrix} \right\} O(3n^2/2)$

Remarque :

Changer le point d'évaluation nécessite uniquement
d'appliquer de nouveau l'algorithme d'Horner !

2

Remarque sur Horner

$$\begin{aligned} P_n(x) &= \sum_{i=0}^n f[x_0, \dots, x_i] N_i(x) \\ &= f[x_0] + f[x_0, x_1](x - x_0) + \dots + f[x_0, \dots, x_n](x - x_0) \dots (x - x_{n-1}) \\ &= f[x_0] + (x - x_0)[f[x_0, x_1] + f[x_0, x_1, x_2](x - x_1) + \dots + f[x_0, \dots, x_n](x - x_1) \dots (x - x_{n-1})] \\ &= f[x_0] + (x - x_0)[f[x_0, x_1] + (x - x_1)[f[x_0, x_1, x_2] + \dots + f[x_0, \dots, x_n](x - x_2) \dots (x - x_{n-1})]] \end{aligned}$$

On commencera donc l'évaluation par

$$f[x_0, \dots, x_{n-1}] + f[x_0, \dots, x_n](x - x_{n-1})$$

Au final :

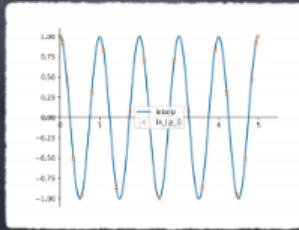
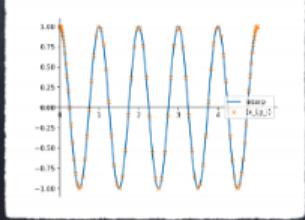
Neville Aitken

- Complexité : $O(5n^2/2)$
- Evaluation en un nouveau point : $O(5n^2/2)$

DD + Horner

- + Complexité : $O(3n^2/2)$
- + Evaluation en un nouveau point : $O(3n)$

MAIS...



3 Interpolation d'une fonction

3.1 Erreur d'interpolation

Considérons une fonction $f: \mathbb{R} \rightarrow \mathbb{R}$ continue.

3.1 Théorème de Weierstrass (admis)

Pour toute fonction f continue, il existe une suite de polynômes $P_n \in \mathbb{P}_n$ t.q.

$$\lim_{n \rightarrow +\infty} \|f - P_n\|_\infty = 0$$

Ce résultat nous indique que toute fonction continue peut être approchée par un polynôme, donc...

Idée : Approcher f par le polynôme d'interpolation P_n vérifiant : $P_n(x_i) = y_i$, où $y_i = f(x_i)$ $\forall i \in \{0, \dots, n\}$

3.2 Définition

On appelle polynôme d'interpolation de Lagrange associé à f le polynôme $P_n f \in \mathbb{P}_n$ vérifiant

$$P_n f(x_i) = f(x_i), \quad \forall i \in \{0, \dots, n\}$$

3.3 Théorème

Si f est de classe C^{n+1} sur $[a, b]$, avec $a \leq x_0 < \dots < x_n \leq b$ alors $\forall x \in [a, b]$, il existe $\xi \in [\min(x_0, x), \max(x_n, x)]$ t.q.

$$f(x) - P_n f(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} v_n(x) \quad \text{où } v_n(x) = \prod_{i=0}^n (x - x_i)$$

Preuve

Posons $Q(t)$ le polynôme d'interpolation associé aux points x_i et x , c'est à dire

$$Q(x_i) = f(x_i) \quad \forall i \in \{0, \dots, n\}$$

$$Q(x) = f(x)$$

remarque : Q est de degré $n+1$

On a vu en TD que

$$Q(t) = \underbrace{\sum_{i=0}^n f[x_0, \dots, x_i] N_i(t)}_{P(t)} + f[x_0, \dots, x_n, x] v_n(t)$$

(décomposition de Q dans la base de Newton)

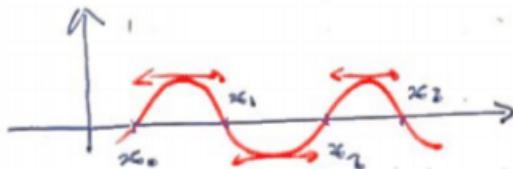
en $t = x$, on a

$$Q(x) = f(x) = P(x) + f[x_0, \dots, x_n, x] v_n(x)$$

Posons $E(t) = f(t) - Q(t)$

On a $E(x_i) = E(x) = 0 \forall i \in \{0, \dots, n\}$

Donc E s'annule $n+2$ fois



D'après le théorème de Rolle, E' s'annule donc au moins $n+1$ points différents.

Ainsi, $E^{(2)}$ s'annule en au moins n points

et $E^{(n+1)}$ s'annulera en au moins 1 point qu'on notera ξ

Alors $E^{(n+1)}(\xi) = 0 \iff f^{(n+1)}(\xi) - Q^{(n+1)}(\xi) = 0$

Or $Q^{(n+1)}(t) = f[x_0, \dots, x_n, x](n+1)!$

On obtient finalement

$$\begin{aligned} f^{(n+1)}(\xi) - f[x_0, \dots, x_n, x](n+1)! &= 0 \\ \iff f[x_0, \dots, x_n, x] &= \frac{f^{(n+1)}(\xi)}{(n+1)!} \end{aligned}$$

D'où

$$f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} v_n(x)$$

3.4 Corollaire

$$\text{On en déduit : } \|f - P_n f\|_\infty \leq \frac{\|f^{(n+1)}\|_\infty}{(n+1)!} \|v_n\|_\infty$$

Preuve

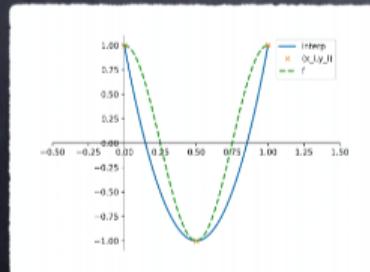
$$\|f - P_n(f)\|_\infty = \frac{1}{(n+1)!} \|f^{(n+1)}(\xi)v_n\|_\infty \leq \frac{\|f^{(n+1)}\| \|v_n\|_\infty}{(n+1)!}$$

Peut-on déduire la convergence ?

$$\|f - P_n f\|_\infty \leq \frac{\|f^{(n+1)}\|_\infty}{(n+1)!} \|v_n\|_\infty \xrightarrow[n \rightarrow +\infty]{} 0?$$

Exemple :

Prenons par exemple les points équirépartis $x_i = i \frac{b-a}{n} + a$ et la fonction $f(x) = \cos(2\pi x)$ sur $[0,1]$:



Sur cet exemple, on observe bien la convergence, ce qui est logique car :

$$\frac{(2\pi)^{n+1}}{(n+1)!} \xrightarrow[n \rightarrow +\infty]{} 0$$

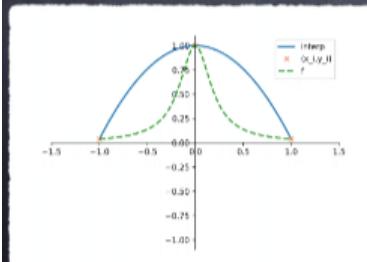
(détails au (vrai) tableau)

Peut-on déduire la convergence ? NON !

$$\|f - P_n f\|_\infty \leq \frac{\|f^{(n+1)}\|_\infty}{(n+1)!} \|v_n\|_\infty \xrightarrow[n \rightarrow +\infty]{} 0?$$

Exemple 2 (Runge) :

Prenons par exemple les points équirépartis $x_i = i \frac{b-a}{n} + a$ et la fonction $f(x) = \frac{1}{1+25x^2}$ sur $[-1,1]$:



On observe ici le phénomène de Runge !

Peut-on mieux choisir les points d'interpolation ?

3.2 Meilleurs points d'interpolations

Revenons à notre majoration :

$$\|f - P_n f\|_{\infty} \leq \frac{\|f^{(n+1)}\|_{\infty}}{(n+1)!} \|v_n\|_{\infty}$$

où on rappelle que $v_n(x) = \prod_{i=0}^n (x - x_i)$

Problème :

Déterminer les coordonnées x_i minimisant $\|v_n\|_{\infty}$.

Pour simplifier l'étude, on se ramènera au cas où

$$a = -1 \leq x_0 < \dots < x_n \leq b$$

ce qui est toujours possible en utilisant le changement de variable:

$$x = (\hat{x} \frac{b-a}{2}) + \frac{b+a}{2}, \text{ où } \hat{x} \in [-1, 1]$$

3.

3.5 Définition

On appelle **polynôme de Chebychev de première espèce** et de degré k le polynôme défini par

$$T_k(x) = \cos(k \arccos(x)) \quad \forall x \in [-1, 1]$$

3.6 Théorème

Les polynômes de Chebychev vérifient la relation de récurrence suivante :

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x), \quad \forall k \geq 1$$

Preuve

Rappel $\cos(a+b) = \cos(a)\cos(b) - \sin(a)\sin(b)$

$$\begin{aligned}
T_{k+1} &= \cos((k+1)\arccos(x)) \\
&= \cos(k\arccos(x) + \arccos(x)) \\
&= \cos(k\arccos(x))\cos(\arccos(x)) - \sin(k\arccos(x))\sin(\arccos(x)) \\
&= 2xT_k(x) - \cos(k\arccos(x))\cos(\arccos(x)) - \sin(k\arccos(x))\sin(\arccos(x)) \\
&= 2xT_k(x) - [-\cos(k\arccos(x))\cos(\arccos(x)) + \sin(k\arccos(x))\sin(\arccos(x))] \\
&= 2xT_k(x) - T_{k+1}(x)
\end{aligned}$$

3.5 Définition

On appelle **polynôme de Chebychev de première espèce et de degré k** le polynôme définie par

$$T_k(x) = \cos(k \arccos(x)) \quad \forall x \in [-1, 1]$$

3.6 Théorème

Les polynômes de Chebychev vérifient la relation de récurrence suivante :

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x), \quad \forall k \geq 1$$

Remarque :

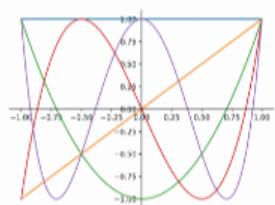
Ce théorème nous « rassure » sur le fait que le polynôme $T_k \in \mathbb{P}_k$.

3.5 Définition

On appelle **polynôme de Chebychev de première espèce et de degré k** le polynôme définie par

$$T_k(x) = \cos(k \arccos(x)) \quad \forall x \in [-1, 1]$$

Illustration :



T_0, T_1, T_2, T_3 et T_4

Remarque :

À l'aide de la définition ci-dessus, on remarque facilement que $T_k(x) \in [-1, 1], \quad \forall x \in [-1, 1]$

Problème :

Déterminer les x_i minimisant $\|v_n\|_\infty$ où $v_n(x) = \prod_{i=0}^n (x - x_i)$

Posons Γ_n l'ensemble des polynômes de degré n :

- ① unitaire, c'est à dire t.q. $P(x) = x^n + \dots$
- ② et possédant n racines réelles distinctes.

Notre problème peut alors se reformuler ainsi :

Trouver $P \in \Gamma_{n+1}$ minimisant $\|P\|_\infty$

Remarque :

Le polynôme $\frac{T_{k+1}}{2^k} \in \Gamma_{k+1}$ (détails au (vrai) tableau)

$$\frac{T_{k+1}}{2^k} \in \Gamma_{k+1}$$

En effet, on a bien $n+1$ racines réelles et de plus $T_{k+1} = 2^k x^{k+1} + \dots$

Prouvons le par récurrence

Si $T_k = 2^{k-1} x^k$

Alors

$$\begin{aligned} T_{k+1} &= 2xT_k - T_{k+1} \\ &= 2^k x^{k+1} + \dots \end{aligned}$$

Ce qui prouve l'hérédité

Et pour $k = 0$, on a $T_0 = \cos(\arccos(x)) = x = 2^0 x$

Intéressons nous maintenant aux racines

$$\begin{aligned} \frac{T_{k+1}}{2^k} = 0 &\iff \cos((k+1)\arccos(x)) = 0 \\ &\iff (k+1)\arccos(x) = \frac{\pi}{2} + n\pi \\ &\iff \underbrace{\arccos(x)}_{\in [0, \pi]} = \frac{(2n+1)\pi}{2(k+1)} \quad n \in [0, \dots, k] \end{aligned}$$

On a alors $k+1$ racines

$$x_n = \cos \left[\frac{(2n+1)\pi}{2(k+1)} \right]$$

3.7 Théorème

Une solution du problème

Trouver $P \in \Gamma_{n+1}$ minimisant $\|P\|_\infty$

est donnée par $\frac{T_{n+1}}{2^n}$.

Preuve

On cherche un candidat qui minimise $\|P\|_\infty$ avec $P \in \Gamma_{n+1}$

Plus précisément, on va prouver que $\frac{T_{n+1}}{2^n} \in \Gamma_{n+1}$ minimise $\|P\|_\infty$.

Supposons qu'il existe $P \in \Gamma_{n+1}$ tq $\|P\|_\infty \leq \|\frac{T_{n+1}}{2^n}\|_\infty$

$$\frac{T_{n+1}}{2^n} = \frac{\cos((n+1)\arccos(x))}{2^n} \in \left[\frac{-1}{2^n}, \frac{1}{2^n} \right]$$

Posons $v = \frac{T_{n+1}}{2^n} - P$

$\frac{T_{n+1}}{2^n}$ atteint son maximum lorsque $(n+1)\arccos(x) = 2k\pi$

$$\iff \underbrace{\arccos(x)}_{\in [0, \pi]} = \frac{2k\pi}{n+1} \quad k \in [0, \dots, \overbrace{\frac{n+1}{2}}^{\text{partie entière}}]$$

Notons $x_k^M = \cos(\frac{2k\pi}{n+1})$ les points où $\frac{T_{n+1}}{2^n}$ est maximal

De même, $\frac{T_{n+1}}{2^n}$ sera minimal en $x_k^m = \cos(\frac{(2k+1)\pi}{n+1})$ $k \in [0, \lfloor \frac{n}{2} \rfloor]$

On déduit que

$$x_0^M > x_0^m > x_1^M > x_1^m > \dots$$

De plus pour $v(x_k^M) > 0$ et $v(x_k^m) < 0$ car $\|P\|_\infty \leq \|\frac{T_{n+1}}{2^n}\|_\infty$

Par le TVI, on obtient que v doit s'annuler au moins $n+1$ fois (dans chaque intervalle $[x_k^M, x_k^m]$ et $[x_k^m, x_{k+1}^m]$)

Or $v = \frac{T_{n+1}}{2^n} - P$ donc, comme $\frac{T_{n+1}}{2^n} \in \Gamma_{n+1}$ et $P \in \Gamma_{n+1}$, on doit avoir

$$v = c_n x^n + \dots$$

donc $\deg(v) = n$, ce qui implique $v = 0$

ce qui contredit que $\|P\|_\infty \leq \|\frac{T_{n+1}}{2^n}\|_\infty$

Remarque

Sur $[-1, 1]$, les points de Tchebychev sont les $\hat{x}_k = \cos\left(\frac{(2k+1)\pi}{n+1}\right)$ $k \in [0..n]$

Sur un intervalle $[a, b]$, on aura

$$x_k = \frac{(\hat{x}_k + 1)}{2}(b - a) + a \in [a, b]$$

Les points de Chebychev résolvent-ils tous nos soucis ?

$$\|f - P_n f\|_{\infty} \leq \frac{\|f^{(n+1)}\|_{\infty}}{(n+1)!} \|v_n\|_{\infty}$$

En choisissant $v_n = \frac{T_{n+1}}{2^n}$, on a $\|v_n\|_{\infty} = \frac{1}{2^n}$.

La question est alors :

Existe-t-il des fonctions dont la norme infinie des dérivées croît plus vite que $(n+1)!2^n$?

et la réponse est... oui ! Donc les points de Chebychev ne fonctionnent pas pour toute fonction.

Les points de Chebychev résolvent-ils tous nos soucis ?

$$\|f - P_n f\|_{\infty} \leq \frac{\|f^{(n+1)}\|_{\infty}}{(n+1)!} \|v_n\|_{\infty}$$

3.8 Théorème (admis)

Pour chaque séquence de points $x_i^{(n)}$, il existe au moins une fonction f continue pour laquelle $\lim_{n \rightarrow +\infty} \|f - P_n f\|_{\infty} > 0$

3.9 Théorème (admis)

Pour chaque fonction f continue, il existe une séquence de points $x_i^{(n)}$ t.q. $\lim_{n \rightarrow +\infty} \|f - P_n f\|_{\infty} = 0$

4 Quelques extensions

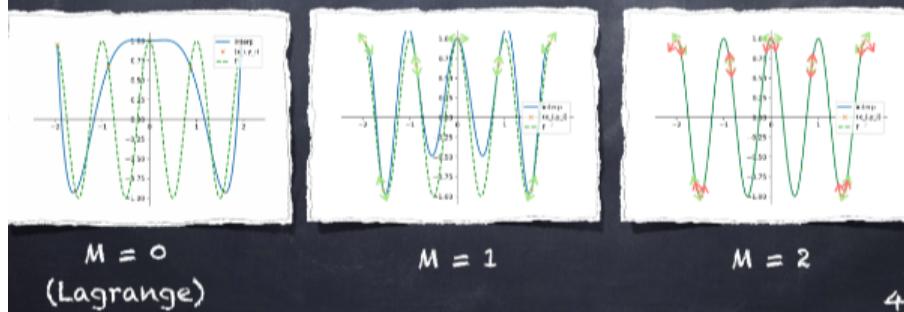
4.1 Interpolation d'Hermite

4.1 Définition

Soit $f \in C^M$, $M \in \mathbb{N}^*$. Le problème d'interpolation d'Hermite consiste à déterminer $P_n f \in \mathbb{P}_n$ t.q.

$$P^{(j)}(x_i) = f^{(j)}(x_i), \quad \forall i \in \{0, \dots, n\}, \forall j \in \{0, \dots, M\}$$

Illustration :



4.2 Théorème

Le problème d'interpolation d'Hermite admet une unique solution $P_n f \in \mathbb{P}_n$ où $N = (M+1) \times (n+1) - 1$

Preuve

Problème d'Hermite, trouver $P \in P[X^N]$ tel que $P^{(j)}(x_i) = f^{(j)}(x_i) \quad \forall j \in \{0, \dots, n\} \quad \forall j \in \{0, \dots, M\}$

On a ici $N = (n+1)(M+1) - 1$.

Prouvons que s'il existe une solution, alors celle-ci est unique.

Supposons que P_1 et P_2 soient deux solutions dans $P[X^N]$ alors $v = P_2 - P_1$ vérifie $v \in P[X^N]$ et $v^{(j)}(x_i) = 0$. Ainsi $\underbrace{v(x)}_{deg=N=(n+1)(M+1)-1} =$

$$\prod_{i=0}^n \underbrace{(x - x_i)^{M+1}}_{deg=(n+1)(M+1)} Q(x). \quad$$
 Ce qui implique $Q(x) = 0$. Donc s'il existe une solution ,celle-ci est unique.

Pour prouver qu'il existe une solution, on va montrer que le problème est

linéaire et revient à résoudre un système linéaire de dimension $N + 1$ avec $N + 1$ inconnues.

Comme la matrice associée est injective (par unicité de la solution) on déduira par le théorème du rang qu'il existe une unique solution.

On cherche $P \in P[X^N]$, donc cela revient à chercher les coefficients α_i tel que

$$\begin{aligned} \left(\sum_{k=0}^N \alpha_k x^k \right)^{(j)}(x_i) &= f^{(j)}(x_i) \\ \iff \alpha_k \underbrace{\sum_{k=0}^N (x^k)^{(j)}(x_i)}_{M_{lk}} &= f^{(j)}(x_i) \text{ avec } l = i + (n+1)j \\ \alpha_0 & \\ \iff M \begin{matrix} \vdots \\ \alpha_N \end{matrix} &= F \end{aligned}$$

Il s'agit bien d'un système linéaire.

4.3 Théorème (admis)

$$\forall x \in [a, b], \exists \xi \in [a, b] \text{ t.q. } f(x) - P_n f(x) = \frac{f^{(N+1)}(\xi)}{(N+1)!} \prod_{i=0}^n (x - x_i)^M$$

Évaluation du polynôme d'interpolation d'Hermite ?

Autrement dit, comment généraliser les algorithmes de Neville-Aitken, Différences Divisées et Horner.

Idée : « Dé doubler » les points d'interpolation !

Pour $M = 1$ par exemple, on pose :

$$y_0 = y_1 = x_0 < y_2 = y_3 = x_1 < \dots < y_{2n} = y_{2n+1} = x_n$$

Notons par $T_k^i \in \mathbb{P}_k$ le polynôme d'interpolation vérifiant

$$(T_k^i)^{(j)}(y_l) = f^{(j)}(y_l) \quad \forall l \in \{i, \dots, i+k\}, \forall j \in \{0, M_l = (l-i) \bmod 2\}$$

où $T_0^i(x) = f(y_i)$

4.4 Théorème

On a la relation suivante entre les T_k^i

$$T_{k+1}^i(x) = \begin{cases} \frac{(y_{i+k+1} - x)T_k^i(x) - (y_i - x)T_{k+1}^{i+1}(x)}{(y_{i+k+1} - y_i)} & \text{si } y_{i+k+1} \neq y_i \\ T_k^i(x) + f'(y_i)(x - y_i) & \text{sinon} \end{cases}$$

Remarque : De cette relation, on peut déduire que les Différences divisées s'appliquent comme dans l'interpolation de Lagrange sauf si $y_i = y_{i+1}$, dans ce cas, on aura $f[y_i, y_{i+1}] = f'[y_i]$

Preuve

Pour $M = 1$

prouvons le résultat par récurrence.

Initialisation :

$$y_0 = y_1 = x_0$$

$$P(x) = \underbrace{f(y_0)}_{f[y_0]N_0(x)} + \underbrace{f'(y_0)(x - y_0)}_{f[y_0, y_1]N_1(x)}$$

Pour $k = 0$, on a

$$T_1^i(x) = \begin{cases} T_0^i(x) + f'(y_i)(x - y_i) & \text{si } y_i = y_{i+1} \\ \frac{T_0^i(x)(y_{i+1} - x) - T_0^{i+1}(x)(y_i - x)}{y_{i+1} - y_i} & \text{sinon} \end{cases}$$

où $T_0^i(x) = f(y_i)$.

Si $y_i = y_{i+1}$ on a par définition T_1^i le polynôme de degré 1 qui vérifie $T_1^i(y_i) = f(y_i)$ et $(T_1^i)'(y_i) = f'(y_i) \iff T_1^i(x) = f(y_i) + f'(y_i)(x - y_i)$ (dl d'ordre 1)

Si $y_i \neq y_{i+1}$, on a par définition

$$\begin{aligned} T_1^i(x) &= f(y_i) + \frac{f(y_{i+1}) - f(y_i)}{y_{i+1} - y_i}(x - y_i) \\ &= T_0^i(x) + \frac{T_0^{i+1}(x) - T_0^i(x)}{y_{i+1} - y_i}(x - y_i) \\ &= \frac{T_0^i(x)(y_{i+1} - y_i) - T_0^{i+1}(x)(y_i - x) - T_0^i(x)(x - y_i)}{y_{i+1} - y_i} \\ &= \frac{T_0^i(x)(y_{i+1} - x) - T_0^{i+1}(x)(y_i - x)}{y_{i+1} - y_i} \end{aligned}$$

hérité

Maintenant , prouvons que pour tout $k \geq 1$ on a

$$T_{k+1}^i(x) = \frac{T_k^i(x)(y_{i+k+1} - x) - T_k^{i+1}(x)(y_i - x)}{y_{i+k+1} - y_i}$$

Posons $\tilde{T}_{k+1}^i(x) = \frac{T_k^i(x)(y_{i+k+1} - x) - T_k^{i+1}(x)(y_i - x)}{y_{i+k+1} - y_i}$ et prouvons que

- $\tilde{T}_{k+1}^i(y_j) = f(y_j) \forall j \in \{i, \dots, i+k+1\}$
- $(\tilde{T}_{k+1}^i)'(y_j) = f'(y_j) \forall j \in \{i, \dots, i+k\}$ tel que $y_j = y_{j+1}$

On a $\forall j \in \{i+1, \dots, i+k\}$

$$\begin{aligned} \tilde{T}_{k+1}^i(y_j) &= \frac{T_{k+1}^i(y_j)(y_{i+k+1} - y_j) - T_{k+1}^i(y_j)(y_i - y_j)}{y_{i+k+1} - y_i} \\ &= f(y_j) \frac{y_{i+k+1} - y_j + y_j - y_i}{y_{i+k+1} - y_i} = f(y_j) \end{aligned}$$

Pour $j = i$ et $j = i+k+1$ la démonstration est la même que dans le cas de Lagrange.

Deuxièmement, on a

$$\begin{aligned} (\tilde{T}_{k+1}^i)'(x) &= \frac{1}{y_{i+k+1} - y_i} [(T_k^i)'(x)(y_{i+k+1} - x) - (T_k^{i+1})'(x)(y_i - x)] \\ &\quad + \frac{1}{y_{i+k+1} - y_i} [-T_k^i(x) - T_k^{i+1}(x)] \end{aligned}$$

Pour $y_j = y_{j+1}$, on aura

$$-T_k^i(y_j) + T_k^{i+1}(y_j) = -f(y_j) + f(y_j) = 0 \forall j \in \{i, \dots, i+k\}$$

De plus on aura ainsi

$$(\tilde{T}_{k+1}^i)'(y_j) = \frac{(T_k^i)'(y_j)(y_{i+k+1} - x) - (T_k^i)'(y_j)(y_i - x)}{y_{i+k+1} - y_i} = f'(y_j)$$

par les mêmes arguments que dans le cas précédent sans la dérivé.

Algorithme de Neville-Aitken (Hermite) :

$$\begin{array}{l}
 \left| \begin{array}{l} f(y_0) = T_0^0(a) \\ f(y_1) = T_0^1(a) \\ f(y_2) = T_0^2(a) \\ \vdots \\ f(y_{2n+1}) = T_0^{2n+1}(a) \end{array} \right. \rightarrow T_1^0(a) = f(y_0) + f'(y_0)(a - y_0) \\
 \left| \begin{array}{l} f(y_1) = T_0^1(a) \\ f(y_2) = T_0^2(a) \end{array} \right. \rightarrow T_1^1(a) = \frac{(y_2 - a)f(y_1) - (y_1 - a)f(y_2)}{(y_2 - y_1)} \\
 \dots
 \end{array}$$

Remarque : A l'aide de cette relation, on peut aussi généraliser l'algorithme des différences divisées

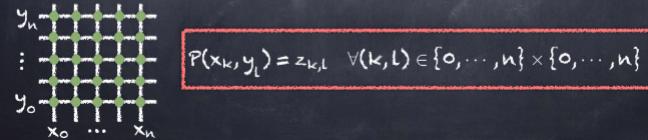
4.2 Interpolation multivariée

Étant donné $(n+1)$ points $(x_i, y_i, z_i)_{i=0, \dots, n} \in \mathbb{R}^3$, déterminer P un polynôme de deux variables vérifiant :

$$P(x_i, y_i) = z_i \quad \forall i \in \{0, \dots, n\}$$

De manière général, ce problème n'est pas évident et il faut en particulier définir la notion de polynôme de degré n en 2D...

Plus simplement, considérons le cas de données sur une grille cartésienne :



4.6 Théorème (admis)

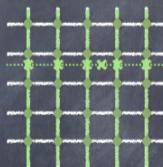
Le problème d'interpolation admet une unique solution dans \mathbb{P}_n .

Évaluation du polynôme d'interpolation ?

On souhaite évaluer P en un point (x^*, y^*) .

Pour ce faire, on commence par évaluer $P(x_i, y')$

(en remarquant que $P(x_i, y) \in \mathbb{P}_n$)



Enfin, connaissant $P(x_i, y')$ et en utilisant le fait que $P(x, y') \in \mathbb{P}_n$ on peut déduire $P(x^*, y')$