

# 一元多项式的 Delta 判别式

sun123zxy

2023-10-18<sup>1</sup>

---

<sup>1</sup>最后更新于 2024-10-26.

- ① 1.  $e$ -基、 $m$ -基与  $p$ -基
  - 1.1 整数分拆
  - 1.2 单项对称多项式
  - 1.3 基本对称多项式
  - 1.4 幂和对称多项式
  - 1.5 其它基底
  
- ② 2. Delta 判别式

- 1 1.  $e$ -基、 $m$ -基与  $p$ -基
  - 1.1 整数分拆
  - 1.2 单项对称多项式
  - 1.3 基本对称多项式
  - 1.4 幂和对称多项式
  - 1.5 其它基底
- 2 2. Delta 判别式

- 1 1.  $e$ -基、 $m$ -基与  $p$ -基
  - 1.1 整数分拆
  - 1.2 单项对称多项式
  - 1.3 基本对称多项式
  - 1.4 幂和对称多项式
  - 1.5 其它基底
  
- 2 2. Delta 判别式

# 整数分拆

设非负整数数列  $\lambda := (\lambda_1, \lambda_2, \dots)$  只有有限项非零且（不严格）单调递减. 定义长度  $\mathcal{L}(\lambda)$  为其非零项元素个数; 定义  $\mathcal{S}(\lambda)$  为其非零项元素之和. 此时称  $\lambda$  是整数  $\mathcal{S}(\lambda)$  的一个长度为  $\mathcal{L}(\lambda)$  的分拆.

由于分拆只有有限项非零, 对大于等于  $\mathcal{L}(\lambda)$  的非负整数  $k$ , 我们也常省略从第  $k+1$  项开始的全为 0 的项, 将  $\lambda$  直接记为长度为  $k$  的非负整数数组  $(\lambda_1, \lambda_2, \dots, \lambda_k)$ .

Ferrers diagram 和 Young diagram 是图示分拆的常见方法.

通过沿主对角线翻转分拆的 Ferrers diagram 或 Young diagram, 可以定义分拆的转置. 分拆  $\lambda$  的转置记为  $\lambda^T$ . 转置后分拆的长度变为原分拆的首项, 而首项变为原分拆的长度.

- ① 1.  $e$ -基、 $m$ -基与  $p$ -基
  - 1.1 整数分拆
  - 1.2 单项对称多项式
  - 1.3 基本对称多项式
  - 1.4 幂和对称多项式
  - 1.5 其它基底
  
- ② 2. Delta 判别式

# 单项对称多项式

设  $n$  是正整数,  $K$  是一个域. 记  $\mathbf{x} := (x_1, \dots, x_n)$ . 设  $\lambda := (\lambda_1, \lambda_2, \dots, \lambda_n)$  是长度不超过  $n$  的一个分拆.

定义  $n$  元多项式环  $K[\mathbf{x}]$  上的关于分拆  $\lambda$  的**单项对称多项式** (monomial symmetric polynomial)  $m_\lambda(\mathbf{x})$  为各项系数为 1 的含有单项式  $\mathbf{x}^\lambda := x_1^{\lambda_1} x_2^{\lambda_2} \dots x_n^{\lambda_n}$  的项数最少的对称多项式.

- $m_{(2,1,0)}(x_1, x_2, x_3) = x_1^2 x_2 + x_1^2 x_3 + x_1 x_2^2 + x_1 x_3^2 + x_2^2 x_3 + x_2 x_3^2$
- $m_{(2,2,1)}(x_1, x_2, x_3) = x_1^2 x_2^2 x_3 + x_1^2 x_2 x_3^2 + x_1 x_2^2 x_3^2$

易见  $m_\lambda(\mathbf{x})$  是次数为  $|\mathcal{S}(\lambda)|$  的齐次 (homogeneous) 多项式. 全体单项对称多项式构成  $n$  元对称多项式环  $\Lambda_n \subset K[\mathbf{x}]$  作为  $K$  上线性空间的一组基底.

# 来数数

## 习题

对一给定的长度不超过  $n$  的分拆  $\lambda := (\lambda_1, \lambda_2, \dots, \lambda_n)$ ,  $n$  元单项对称多项式  $m_\lambda(x)$  共有多少项?

在计数时根据分拆中重复项的分布情况进行消序.

## 习题

$n$  元  $d$  次单项对称多项式共有多少种可能的构型? 设  $\Lambda_n^{(d)} \subset \Lambda_n$  由全体至多  $d$  次的  $n$  元对称多项式构成, 其作为  $K$  上线性空间的维数是多少?

该问题等价于求满足  $\mathcal{S}(\lambda) = d$ ,  $\mathcal{L}(\lambda) \leq n$  的所有可能分拆  $\lambda$  的数量, 也即“将  $d$  个无标号球放入  $n$  个可空置的无标号盒”的可行方案数.



- 1 1.  $e$ -基、 $m$ -基与  $p$ -基
  - 1.1 整数分拆
  - 1.2 单项对称多项式
  - 1.3 基本对称多项式
  - 1.4 幂和对称多项式
  - 1.5 其它基底
  
- 2 2. Delta 判别式

# 基本对称多项式

$n$  元多项式环  $K[x]$  上的  $n$  个基本对称多项式 (elementary symmetric polynomial) 定义为

$$e_k(x_1, \dots, x_n) := \sum_{1 \leq i_1 < i_2 < \dots < i_k \leq n} x_{i_1} x_{i_2} \dots x_{i_k}, \quad k = 1, 2, \dots, n$$

使用单项对称多项式的记号, 也可记为

$$e_k(\mathbf{x}) := m_{\lambda_k}(\mathbf{x})$$

其中分拆  $\lambda_k := (1, \dots, 1, 0, \dots)$  的前  $k$  项为 1, 其余项皆为 0.

方便起见, 定义  $e_0 = 1$ , 定义  $k > n$  和  $k < 0$  的  $e_k = 0$ . 设分拆  $\lambda := (\lambda_1, \lambda_2, \dots)$  满足  $\lambda_i \leq n, \quad \forall i \in \mathbb{N}_+$ . 我们记

$$e_{\lambda}(\mathbf{x}) := e_{\lambda_1}(\mathbf{x}) e_{\lambda_2}(\mathbf{x}) \dots e_{\lambda_{\mathcal{L}(\lambda)}}(\mathbf{x}).$$

基本对称多项式有生成函数

$$E(s) := \sum_{k \in \mathbb{Z}} e_k s^k = \prod_{i=1}^m (1 + x_i s)$$

# 对称多项式基本定理

## 定理 (对称多项式基本定理)

设  $f(\mathbf{x})$  是域  $K$  上的  $n$  元对称多项式, 则存在唯一的  $g(\mathbf{x}) \in K[\mathbf{x}]$ , 使得

$$f(\mathbf{x}) = g(e_1(\mathbf{x}), \dots, e_n(\mathbf{x}))$$

该定理对交换环上的对称多项式仍然成立. 这意味着若  $f$  是整系数对称多项式, 则  $g$  也是整系数多项式.

在定理的存在性证明中, 为消去首项对应的单项对称多项式  $m_\lambda(\mathbf{x})$ , 我们构造的若干个基本对称多项式的乘积恰为  $e_\lambda$ .

考察全体满足  $\lambda_i \leq n, \quad \forall i \in \mathbb{N}_+$  的分拆  $\lambda$  对应的  $e_\lambda(\mathbf{x})$ , 它们构成了  $n$  元对称多项式环  $\Lambda_n$  作为  $K$  上线性空间的另一组基底.

- ① 1.  $e$ -基、 $m$ -基与  $p$ -基
  - 1.1 整数分拆
  - 1.2 单项对称多项式
  - 1.3 基本对称多项式
  - 1.4 幂和对称多项式
  - 1.5 其它基底
  
- ② 2. Delta 判别式

# 幂和对称多项式

$n$  元多项式环  $K[x]$  上的**幂和对称多项式** (power sum symmetric polynomial) 定义为

$$p_k(x_1, \dots, x_n) := x_1^k + x_2^k + \cdots + x_n^k, \quad k \in \mathbb{N}_{\geq 0}$$

使用单项对称多项式的记号, 也可记为

$$p_k(\mathbf{x}) := m_{(k,0,0,\dots)}$$

特别的,  $p_0(\mathbf{x}) = n$ . 方便起见, 定义  $k < 0$  的  $p_k = 0$ .

## 定理

设  $\mathbb{Q} \subset K \subset \mathbb{C}$  是数域, 设  $f(\mathbf{x})$  是域  $K$  上的  $n$  元对称多项式, 则存在唯一的  $g(x) \in K[x]$ , 使得  $f(\mathbf{x}) = g(p_1(\mathbf{x}), \dots, p_n(\mathbf{x}))$ .

一般地, 结论对特征为 0 的域  $K$  也成立.

幂和对称多项式有生成函数

$$P(s) := \sum_{k \in \mathbb{Z}} p_k s^k = \sum_{k=0}^{+\infty} s^k \sum_{i=1}^m x_i^k = \sum_{i=1}^m \sum_{k=0}^{+\infty} (x_i s)^k = \sum_{i=1}^m \frac{1}{1 - x_i s}$$

# Newton 公式

以下定理递推地给出了幂和对称多项式  $p_1, \dots, p_n$  与基本对称多项式  $e_1, \dots, e_n$  间的关系. 定理 1.2 的存在性部分可由这一定理给出.

## 定理 (Newton's Identities)

$$p_k = \sum_{i=1}^{k-1} (-1)^{i-1} e_i p_{k-i} + (-1)^{k-1} k e_k \quad k = 1, 2, \dots, n$$

$$p_k = \sum_{i=1}^n (-1)^{i-1} e_i p_{k-i} \quad k > n$$

$$k e_k = \sum_{i=1}^k (-1)^{i-1} p_i e_{k-i} \quad k = 1, 2, \dots, n$$

$$0 = \sum_{i=1}^n (-1)^{i-1} p_i e_{k-i} \quad k > n$$



# 更简单的写法

$$\sum_{i+j=k} (-1)^i e_i p_j = 0, \quad k \in \mathbb{N}_{\geq 0}$$

但在求和中“认为”  $p_0 = k$ .

# 一个基于生成函数的证明

$$\frac{sE'(s)}{E(s)} = s \frac{d}{ds} (\ln E(s)) = \sum_{i=1}^m \frac{x_i s}{1 + x_i s} = m - \sum_{i=1}^m \frac{1}{1 + x_i s} = m - P(-s)$$

即

$$\sum_{k=0}^{\infty} k e_k s^k = sE'(s) = E(s)(m - P(-s)) = E(s) \sum_{k=1}^{\infty} (-1)^{k-1} p_k$$

对比各项即得 Newton 公式.

- 1 1.  $e$ -基、 $m$ -基与  $p$ -基
  - 1.1 整数分拆
  - 1.2 单项对称多项式
  - 1.3 基本对称多项式
  - 1.4 幂和对称多项式
  - 1.5 其它基底
  
- 2 2. Delta 判别式

完全齐次对称多项式 (Complete homogeneous symmetric polynomials)、Schur 多项式.....

本节主要参考 [[Wikf](#); [Wikb](#); [Wike](#); [Wikc](#)].

- ① 1.  $e$ -基、 $m$ -基与  $p$ -基
  - 1.1 整数分拆
  - 1.2 单项对称多项式
  - 1.3 基本对称多项式
  - 1.4 幂和对称多项式
  - 1.5 其它基底
  
- ② 2. Delta 判别式

# Vieta's formulas

## 定理 (Vieta's formulas)

设数域  $K \subset \mathbb{C}$  上  $n$  次首一多项式 (*monic polynomial*)

$$A(x) := x^n + a_{n-1}x^{n-1} + \cdots + a_0 = (x - c_1)(x - c_2) \cdots (x - c_n)$$

其  $n$  个复根分别为  $\mathbf{c} := (c_1, c_2, \dots, c_n)$ , 则  $A(x)$  的系数可由关于根的  $n$  个  $n$  元基本对称多项式表示

$$a_{n-k} = e_k(-\mathbf{c}) = (-1)^k \sum_{1 \leq i_1 < \cdots < i_k \leq n} c_{i_1} c_{i_2} \cdots c_{i_k}$$

其中  $k = 1, 2, \dots, n$ . 特别的

$$a_0 = e_n(-\mathbf{c}) = (-1)^n c_1 c_2 \cdots c_n$$

$$a_{n-1} = e_1(-\mathbf{c}) = -(c_1 + c_2 + \cdots + c_n)$$

# Vieta 定理与对称多项式基本定理

- 即使尚未获知多项式  $n$  个复根  $c_1, \dots, c_n$  的具体取值, 我们也能通过已知的多项式系数  $a_0, \dots, a_{n-1}$  获知  $n$  个  $n$  元基本对称多项式在根处的取值.
- 对称多项式基本定理指出, 任何对称多项式都可被 (唯一) 表示为关于  $n$  个基本对称多项式的一个多项式.
- 仅需知晓多项式的系数, 就可获得任意给定对称多项式在根处的取值.
- 目标: 构造一个 (数域  $K \subset \mathbb{C}$  上的)  $n$  元对称多项式, 使得能通过代入求值的方式, 快速检测  $n$  个复数是否两两不同.

# Vandermonde 行列式

考察作为（数域  $K \subset \mathbb{C}$  上） $n$  元多项式的 Vandermonde 行列式

$$\begin{aligned}\det V &:= \det \left( x_j^{i-1} \right)_{i=1, j=1}^{n, n} \\ &= \det \begin{pmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n-1} & x_2^{n-1} & \dots & x_n^{n-1} \end{pmatrix} = \prod_{1 \leq i < j \leq n} (x_j - x_i)\end{aligned}$$

它是否可用于判定重根？它是对称多项式吗？

## 注记

$\det V$  是一个斜对称多项式。事实上， $\det V$  与所有对称多项式的乘积构成了全体斜对称多项式（alternating polynomials）。



# 判别式

设 (数域  $K \subset \mathbb{C}$  上的)  $n$  元对称多项式

$$D(x_1, \dots, x_n) := (\det V)^2 = \prod_{1 \leq i < j \leq n} (x_j - x_i)^2$$

称其为 (数域  $K$  上) 一元  $n$  次首一多项式的**判别式** (Discriminant). 当代入的  $x := (x_1, \dots, x_n) \subset \mathbb{C}$  互不相同,  $D(x) \neq 0$ ; 否则  $D(x) = 0$ .

根据对称多项式基本定理, 存在唯一数域  $K$  上的  $n$  元多项式  $d$ , 使得  $d(e_1(x), \dots, e_n(x)) = D(x)$ .

## 命题

数域  $K \subset \mathbb{C}$  上的  $n$  次首一多项式  $f(x) := x^n + a_{n-1}x^{n-1} + \dots + a_0$  在复数域中有重根的充分必要条件是  $d(a_{n-1}, \dots, a_0) = 0$ .

这是因为  $f(x)$  的  $n$  个复根  $c := (c_1, \dots, c_n)$  满足

$$D(-c) = d(e_1(-c), \dots, e_n(-c)) = d(a_{n-1}, \dots, a_0)$$

## 习题

写出数域  $K \subset \mathbb{C}$  上一元二次多项式  $x^2 + bx + c$  的判别式.

对次数更高的方程, 直接使用消首项方法求解判别式  $D(x)$  在基本对称多项式下的表示  $d(e_1, \dots, e_n)$  将变得相当繁琐. 下面利用判别式与 Vandermonde 行列式的关系得到另一种分解方法.

# 另一分解方法

$$\begin{aligned} D(\mathbf{x}) &= (\det V)^2 = \det(VV^T) \\ &= \det\left(\begin{pmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{n-1} & x_2^{n-1} & \dots & x_n^{n-1} \end{pmatrix} \begin{pmatrix} 1 & x_1 & \dots & x_1^{n-1} \\ 1 & x_2 & \dots & x_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \dots & x_n^{n-1} \end{pmatrix}\right) \\ &= \det\begin{pmatrix} n & p_1(\mathbf{x}) & \dots & p_{n-1}(\mathbf{x}) \\ p_1(\mathbf{x}) & p_2(\mathbf{x}) & \dots & p_n(\mathbf{x}) \\ \vdots & \vdots & \ddots & \vdots \\ p_{n-1}(\mathbf{x}) & p_n(\mathbf{x}) & \dots & p_{2n-2}(\mathbf{x}) \end{pmatrix} = \det(p_{i+j-2}(\mathbf{x}))_{i=1, j=1}^{n, n} \end{aligned}$$

而由 Newton's Identities, 幂和对称多项式  $p_k(\mathbf{x})$  可较容易地递推分解为基本对称多项式的多项式组合, 故我们找到了分解  $D(\mathbf{x})$  的一种更易操作的方法.

# 另一分解方法

## 习题

写出数域  $K \subset \mathbb{C}$  上不完全三次多项式  $x^3 + bx + c$  的判别式.

本节主要参考 [丘维声 15; 蓝以中 07; Wikd; Wika].

# 参考文献 I

- [Wika] Wikipedia. *Alternating polynomial*. [https://en.wikipedia.org/wiki/Alternating\\_polynomial](https://en.wikipedia.org/wiki/Alternating_polynomial).
- [Wikb] Wikipedia. *Elementary symmetric polynomial*. [https://en.wikipedia.org/wiki/Elementary\\_symmetric\\_polynomial](https://en.wikipedia.org/wiki/Elementary_symmetric_polynomial).
- [Wikc] Wikipedia. *Newton's identities*. [https://en.wikipedia.org/wiki/Newton%27s\\_identities](https://en.wikipedia.org/wiki/Newton%27s_identities).
- [Wikd] Wikipedia. *Partition (number theory)*. [https://en.wikipedia.org/wiki/Partition\\_\(number\\_theory\)](https://en.wikipedia.org/wiki/Partition_(number_theory)).
- [Wike] Wikipedia. *Power sum symmetric polynomial*. [https://en.wikipedia.org/wiki/Power\\_sum\\_symmetric\\_polynomial](https://en.wikipedia.org/wiki/Power_sum_symmetric_polynomial).
- [Wikf] Wikipedia. *Symmetric polynomial*. [https://en.wikipedia.org/wiki/Symmetric\\_polynomial](https://en.wikipedia.org/wiki/Symmetric_polynomial).

- [丘维声 15] 丘维声. “高等代数 下册”. In: 3rd ed. 北京: 高等教育出版社, 2015. Chap. 多项式环, pp. 57–66. ISBN: 978-7-04-042235-1.
- [蓝以中 07] 蓝以中. “高等代数简明教程 (下册)”. In: 2nd ed. 北京: 北京大学出版社, 2007. Chap. 多元多项式环, pp. 213–217. ISBN: 978-7-301-05579-3.