

# *MACHINE LEARNING*

អត្ថបទសង្ខេបជាកាសាខ្មែរ

លីម ម៉េងសាយ

*Mengsay's NOTES*

## Table of Contents

សេចក្តីផ្តើមអំពី MACHINE LEARNING .....	2
ម៉ូដែលតម្រូវតម្រង់លីនេអ៊ែរ (LINEAR REGRESSION MODEL) .....	3
វិធីសាស្ត្របរមាណូតាមរយៈ SGD .....	11
REGULARIZATION IN MACHINE LEARNING .....	17
បញ្ហាចំណាត់ថ្នាក់២ក្រុម (BINARY CLASSIFICATION PROBLEM) .....	22
បញ្ហាចំណាត់ថ្នាក់ច្រើនក្រុម (MULTICLASS CLASSIFICATION PROBLEM) .....	30
SUPPORT VECTOR MACHINE .....	37
CLUSTERING .....	45
ឯកសារពិគ្រោះ: .....	52

# សេចក្តីផ្តើមអំពី Machine Learning

Machine Learning គឺសំដៅដល់បច្ចេកវិទ្យាដែលមានគោលដៅផ្តល់ឱ្យម៉ាស៊ីន (កុំព្យូទ័រ) នូវសមត្ថភាពដោះស្រាយបញ្ហាបានដោយខ្លួនឯង ពោលគឺមនុស្សមិនផ្តល់នូវបែបបទជាក់លាក់អំពីរបៀបដោះស្រាយណាមួយឡើយ។ ការរៀនពីរបៀបដោះស្រាយបញ្ហាក្នុង Machine Learning គឺធ្វើឡើងតាមរយៈការវិភាគលើទិន្នន័យជាគម្រូដែលផ្តល់ឱ្យដោយមនុស្ស។

បើធ្វើចំណាត់ថ្នាក់ក្រុមដោយផ្អែកលើប្រភេទគម្រូដែលផ្តល់ឱ្យម៉ាស៊ីនដើម្បីសង្កេត នោះគេអាចបែងចែកជា Supervised learning និង Unsupervised learning។ ចំពោះ Supervised learning ទិន្នន័យដែលផ្តល់ឱ្យម៉ាស៊ីនមានធាតុចូល (input) និងចម្លើយនៃបញ្ហាភ្ជាប់ជាមួយ។ ករណីនេះប្រៀបដូចជាករណីគ្រូផ្តល់លំហាត់និងដំណោះស្រាយជាមួយគ្នាឱ្យសិស្សរៀន។ ផ្ទុយទៅវិញ Unsupervised learning ទិន្នន័យដែលផ្តល់ឱ្យម៉ាស៊ីនមានតែធាតុចូល (input)។ ករណីនេះអាចប្រៀបដូចជាករណីគ្រូផ្តល់តែលំហាត់ឱ្យសិស្សរៀនដោះស្រាយ សង្កេតដោយខ្លួនឯង។

## 1. Supervised Learning

ចំពោះ Supervised Learning, គម្រូនៃធាតុចូល  $x$  និង ចម្លើយនៃបញ្ហា  $y$  ជាច្រើន  $\{(x_i, y_i)\}_{i=1}^n$  ត្រូវបានផ្តល់ឱ្យ។ អ្វីដែល Machine Learning ធ្វើគឺចង់កំណត់នូវទំនាក់ទំនងរវាង  $x$  និង  $y$  ។ ដោយផ្អែកលើប្រភេទនៃតម្លៃ  $y$  នោះគេអាចបែងចែកជា

- ចំណោទតម្រូវតម្រង់ Regression problem ( $y$  ជាអថេរជាប់ ចំនួនពិត)
  - Ex: ប៉ាន់ស្មានតម្លៃអចលនទ្រព្យ  $y = 120000, 498302.25 \dots$
- ចំណោទចំណាត់ថ្នាក់ទិន្នន័យ Discrimination problem/ Classification ( $y$  ជាអថេរដាច់)
  - Ex: ចំណាត់ថ្នាក់ប្រភេទសារ spam ( $y = 1$ ), not spam ( $y = 0$ ) (input  $x$ : mail text)
  - Ex: កំណត់លេខសរសេរដៃ  $y = 0, 1, 2, \dots, 9$ . (input  $x$ : រូបភាព)

## 2. Unsupervised Learning

ចំពោះ Unsupervised Learning, គម្រូនៃធាតុចូល  $x$  ជាច្រើន  $\{x_i\}_{i=1}^n$  ត្រូវបានផ្តល់ឱ្យដោយគ្មានគូចម្លើយនៃបញ្ហា  $y$  ។ អ្វីដែល Machine Learning ធ្វើគឺចង់ទាញរកនូវលក្ខណៈឬទម្រង់ពិសេសពីទិន្នន័យ  $x$ ។ បើនិយាយពី Machine learning បែបស្ថិតិវិទ្យា បញ្ហាប្រភេទនេះមានដូចជា

Dimensionality reduction : ការបង្ហាញទិន្នន័យដែលមានវិមាត្រខ្ពស់មកជាវិមាត្រទាបដោយរក្សាលក្ខណៈពិសេសនៃទិន្នន័យ

Feature selection : ការកំណត់នូវធាតុសំខាន់ដែលមានឥទ្ធិពលលើការប៉ាន់ស្មានអ្វីមួយ

Clustering : ការធ្វើចំណាត់ថ្នាក់ក្រុមដោយស្វ័យប្រវត្តិដោយផ្អែកលើលក្ខណៈនៃទិន្នន័យដែលមាន

## ម៉ូដែលតម្រូវតម្រង់លីនេអ៊ែរ (Linear Regression Model)

ការសិក្សាលើម៉ូដែលនៃបាតុភូត អាចឱ្យគេរកឃើញពីមូលហេតុឬកត្តាទាក់ទងនៃបាតុភូតនោះបាន។ លើសពីនេះគេក៏អាចប្រើប្រាស់ម៉ូដែលនោះសម្រាប់ការទស្សន៍ទាយសម្រាប់អនាគតឬការប៉ាន់ស្មានលើទិន្នន័យដែលមិនមានក្នុងដៃ។

ជាទូទៅម៉ូដែលដែលសិក្សាអំពីទំនាក់ទំនងរវាងលទ្ធផលនៃបាតុភូតមួយនិងកត្តាដែលអាចគិតបានថាជាកត្តាជះឥទ្ធិពលលើលទ្ធផលនោះ ហៅថា ម៉ូដែលតម្រូវតម្រង់ (Regression Model) ។ ក្នុងសប្តាហ៍នេះយើងនឹងណែនាំអំពីម៉ូដែលមានទម្រង់ជាលីនេអ៊ែរដែលជាមូលដ្ឋាននៃតម្រូវតម្រង់។

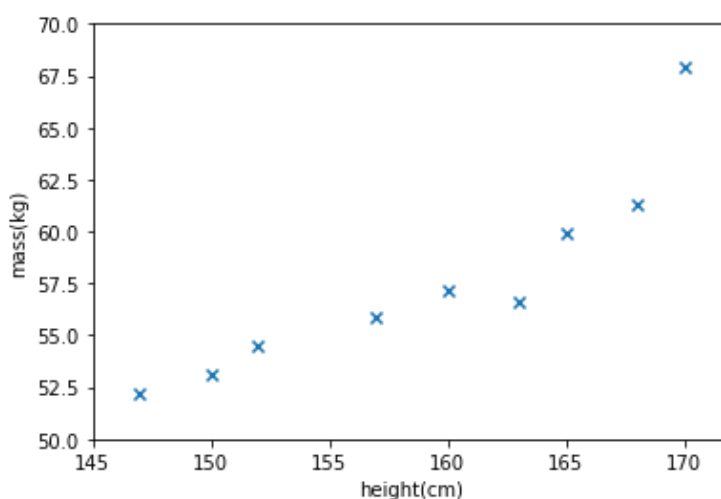
### 1. ការសិក្សាលើទំនាក់ទំនងរវាង២អថេរ

ជាឧទាហរណ៍យើងលើកយកការសិក្សារវាងទំនាក់ទំនងរវាងកម្ពស់ (cm) និងម៉ាស់ (kg) តាមរយៈម៉ូដែលតម្រូវតម្រង់លីនេអ៊ែរ។

#### 1.1. ទិន្នន័យនិងម៉ូដែល

តារាងទី១ ទិន្នន័យកម្ពស់ (cm) និងម៉ាស់ (kg) មនុស្ស៩នាក់

កម្ពស់ (cm)	152	157	160	163	150	147	165	168	178
ម៉ាស់ (kg)	54.48	55.84	57.20	58.57	53.12	52.21	59.93	61.29	69.92



រូបទី១ របាយទិន្នន័យពីតារាងទី១

ក្នុងតារាងទី១ ទិន្នន័យអំពីកម្ពស់គិតជាសង់ទីម៉ែត្រ (cm) និងម៉ាស់គិតជាគីឡូក្រាម (kg) របស់មនុស្ស ៩នាក់។ ពីទិន្នន័យនេះ យើងចង់សិក្សាពីទំនាក់ទំនងរវាងកម្ពស់ ( $x$ ) និងម៉ាស់ ( $y$ ) ។ នៅទីនេះយើងសន្មតថា តម្លៃម៉ាស់ ( $y$ ) គឺជាអនុគមន៍នៃតម្លៃកម្ពស់ ( $x$ ):  $y = f(x)$  ។

ដើម្បីងាយស្រួល ជាដំបូងយើងឧបមាថាទំនាក់ទំនងដើមរវាង  $x, y$  កំណត់ដោយអនុគមន៍ដូចខាងក្រោមដែលយើងហៅថា **ម៉ូដែលលីនេអ៊ែរ** ។

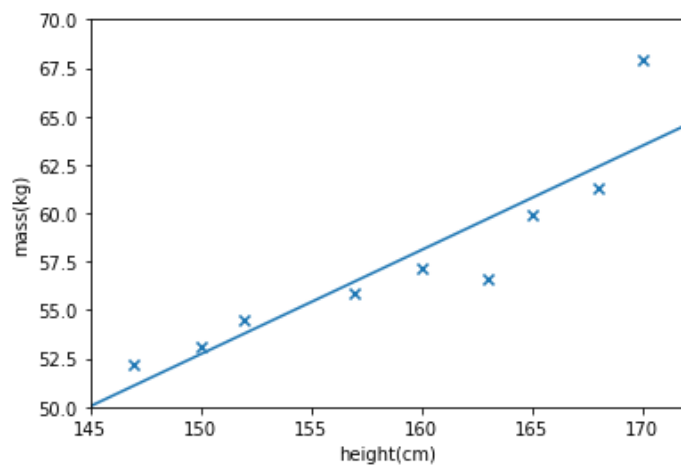
$$y = f(x) = \beta_0 + \beta_1 x$$

ក្រោមឧបមានេះទិន្នន័យដែលមានក្នុងតារាង១ ត្រូវផ្ទៀងផ្ទាត់ទំនាក់ទំនងខាងលើនេះ។ ប៉ុន្តែក្នុងដំណើរការវាស់វែង ឬ ស្រង់ទិន្នន័យលម្អៀងឬគម្លាតរវាងតម្លៃតាមម៉ូដែលដែលឧបមាខាងលើនិងតម្លៃទិន្នន័យជាក់ស្តែងតែងតែកើតឡើង។ ហេតុនេះ យើងសិក្សាករណីដូចខាងក្រោម។

$$\text{ម៉ាស់ } (y) = \beta_0 + \beta_1 \times \text{កម្ពស់ } (x) + \text{លម្អៀង } (\epsilon)$$

ក្នុងពេលនេះទិន្នន័យដែលមានអាចសរសេរជាទំនាក់ទំនងដូចខាងក្រោម។

$$54.58 = \beta_0 + \beta_1 \times 152 + \epsilon, \dots \dots \dots$$



**រូបទី២ របាយទិន្នន័យនិងបន្ទាត់តម្រកម្រងលីនេអ៊ែរ(ម៉ូដែល)**

ជាទូទៅចំពោះទិន្នន័យចំនួន  $N$ :  $\{(x_i, y_i)\}_{i=1}^N$  ទិន្នន័យនីមួយៗអាចបង្ហាញដោយទំនាក់ទំនងដូចខាងក្រោម។

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \quad (i = 1, 2, \dots, N)$$

នៅទីនេះ  $\beta_0, \beta_1$  ហៅថាមេគុណតម្រិតម្រង់ (regression coefficient)  $\epsilon$  គឺតម្លៃលម្អៀងរវាងទិន្នន័យពីការវាស់វែងនិងតម្លៃពិតតាមការសន្មត។ តម្លៃនៃម៉ាស  $y$  ហៅថា អថេរគោលដៅ (subjective variable)  $x$  ហៅថា អថេរពន្យល់ឬអថេរឯករាជ្យ (explanatory variable) ។

គោលដៅរបស់យើងនៅទីនេះគឺការកំណត់តម្លៃនៃមេគុណតម្រិតម្រង់ ដែលធ្វើអោយម៉ូដែលដែលសន្មតខាងលើអាចបង្ហាញទំនាក់ទំនងរវាងអថេរពន្យល់និងអថេរគោលដៅបានល្អប្រសើរ។

មានវិធីជាច្រើនដែលអាចឱ្យយើងកំណត់តម្លៃនៃមេគុណតម្រិតម្រង់ដែលល្អប្រសើរសម្រាប់ទិន្នន័យដែលមាន។ នៅក្នុងអត្ថបទនេះ យើងនឹងណែនាំអំពីវិធីសាស្ត្រងាយនិងពេញនិយម Least Square Error ។

## 1.2. ការប៉ាន់ស្មានតម្លៃប៉ារ៉ាម៉ែត្រដោយLeast Square Error

តាមការសន្មតនៃម៉ូដែលខាងលើ តម្លៃនៃលម្អៀងរវាងទិន្នន័យនិមួយៗនិងតម្លៃពិតតាមម៉ូដែលអាចកំណត់បានដូចខាងក្រោម។

$$\epsilon_i = y_i - (\beta_0 + \beta_1 x_i) , \quad (i = 1, 2, \dots, N)$$

ក្នុងវិធីសាស្ត្រLeast Square Error យើងសិក្សាលើផលបូកនៃការេរបស់តម្លៃលម្អៀងទាំងអស់របស់ទិន្នន័យដែលមាន ពោលគឺ  $\epsilon_1^2 + \epsilon_2^2 + \dots + \epsilon_N^2$  ។ គំនិតក្នុងវិធីសាស្ត្រនេះគឺងាយស្រួល។ ម៉ូដែលដែលអាចពន្យល់ទំនាក់ទំនងរវាងអថេរទាំង២បានល្អប្រសើរ អាចត្រូវបាននិយាយបានថាជាម៉ូដែលដែលមានតម្លៃនៃកម្រិតលម្អៀងតូចបំផុត។ ហេតុនេះ យើងនឹងធ្វើការកំណត់តម្លៃមេគុណតម្រិតម្រង់ (ប៉ារ៉ាម៉ែត្រ) ណាដែលធ្វើឱ្យតម្លៃនៃផលបូកនៃការេរបស់តម្លៃលម្អៀងទាំងអស់របស់ទិន្នន័យ  $E(\beta_0, \beta_1)$  មានតម្លៃតូចបំផុត។

$$E(\beta_0, \beta_1) = \sum_{i=1}^N \epsilon_i^2 = \sum_{i=1}^N \{y_i - (\beta_0 + \beta_1 x_i)\}^2$$

អ្នកដែលបានសិក្សាគណិតវិទ្យា អាចមើលឃើញយ៉ាងងាយថា ពេលនេះវាបានក្លាយជាបញ្ហាបរមា កម្ម លើតម្លៃ  $E(\beta_0, \beta_1)$  ដោយយក  $\beta_0, \beta_1$  ជាអថេរ។ យើងអាចដោះស្រាយបញ្ហានេះបានដោយងាយដោយប្រើចំណេះដឹងផ្នែកវិភាគមូលដ្ឋានដូចជាដេរីវេដោយផ្នែក។

នៅទីនេះ ដើម្បីមានភាពងាយស្រួលក្នុងការសិក្សាលើករណីច្រើនអថេរពន្យល់ យើងនឹងណែនាំការដោះស្រាយបញ្ហាខាងលើដោយប្រើវ៉ិចទ័រនិងម៉ាទ្រីស។

យើងកំណត់សរសេរម៉ាទ្រីសនិងវ៉ិចទ័រដូចខាងក្រោម។  $X$  ពេលខ្លះត្រូវបានហៅថាម៉ាទ្រីសផែនការ។

$$X = \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_N \end{pmatrix} \in \mathbb{R}^{N \times 2}, \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix} \in \mathbb{R}^N, \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_N \end{pmatrix} \in \mathbb{R}^N, \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} \in \mathbb{R}^2$$

ពេលនេះម៉ូដែលនិងផលបូកតម្លៃការវែនលម្អៀងខាងលើអាចសរសេរដូចខាងក្រោម។

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

$$E(\boldsymbol{\beta}) = \sum_{i=1}^N \epsilon_i^2 = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\top (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$$

ត្រលប់ទៅកាន់បញ្ហាបស់យើងវិញ។ គោលដៅរបស់យើងគឺកំណត់តម្លៃ  $\boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} \in \mathbb{R}^2$  ដែលធ្វើឱ្យតម្លៃនៃ  $E(\boldsymbol{\beta})$  តូចបំផុត។ នៅទីនេះដូចដែលបានឃើញស្រាប់ អនុគមន៍  $E(\boldsymbol{\beta})$  ជាអនុគមន៍ប៉ោង ហេតុនេះយើងអាចកំណត់តម្លៃអប្បបរមារបស់វាបានងាយដោយគ្រាន់តែគណនាដេរីវេធៀបនឹងប៉ារ៉ាម៉ែត្រ  $\boldsymbol{\beta}$  ដូចខាងក្រោម។

$$E(\boldsymbol{\beta}) = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^\top (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = \mathbf{y}^\top \mathbf{y} - 2\mathbf{y}^\top \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{X}\boldsymbol{\beta}$$

$$\frac{\partial}{\partial \boldsymbol{\beta}} E(\boldsymbol{\beta}) = -2\mathbf{X}^\top \mathbf{y} + 2\mathbf{X}^\top \mathbf{X}\boldsymbol{\beta}$$

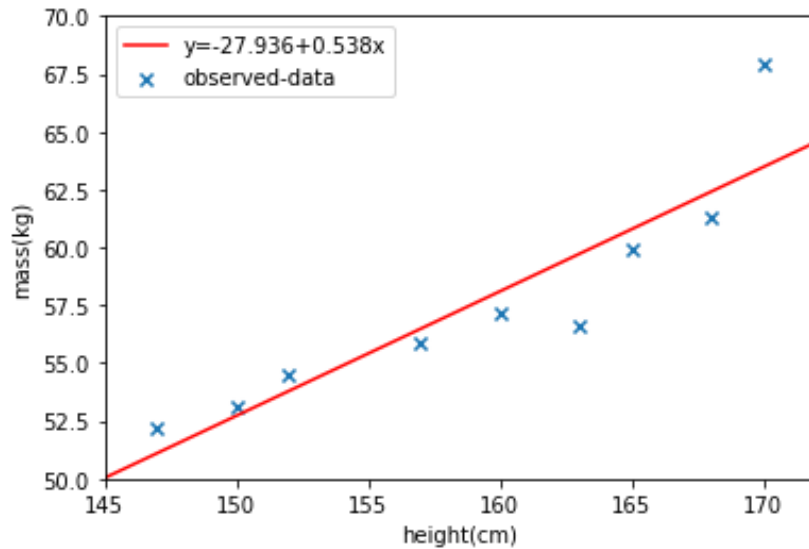
ដោយដោះស្រាយសមីការ  $\frac{\partial}{\partial \boldsymbol{\beta}} E(\boldsymbol{\beta}) = \mathbf{0}$  យើងបាន

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

ដោយជំនួសតម្លៃដែលគណនាបាននេះទៅក្នុងម៉ូដែលដើម យើងអាចគណនាតម្លៃទស្សន៍ទាយនៃអថេរគោលដៅ  $\hat{y}$  នៅពេលស្គាល់តម្លៃអថេរពន្យល់  $x$  បានដូចខាងក្រោម។

$$\hat{y} = x\hat{\boldsymbol{\beta}}$$

ក្នុងករណីទិន្នន័យក្នុងតារាងទី១ខាងលើ យើងអាចទទួលបានតម្លៃនៃមេគុណតម្រិតម្រង់និងបន្ទាត់តាងម៉ូដែលតម្រិតម្រង់លីនេអ៊ែរដូចខាងក្រោម។



រូបទី៣ របាយទិន្នន័យនិងបន្ទាត់តម្រែតម្រង់លីនេអ៊ែរតាមLeast Square Error

### 1.3 ឯករាជភាពនៃតម្លៃលម្អៀង Independence of errors

យើងពិនិត្យលើទំនាក់ទំនងរវាងតម្លៃនៃលម្អៀងនិងអថេរគោលដៅនិងអថេរពន្យល់។

បើសង្កេតលើតម្លៃនៃកូរ៉េឡង់រវាងតម្លៃលម្អៀង  $\epsilon$  និងតម្លៃទស្សន៍ទាយនៃអថេរគោលដៅ  $y$  ឬ តម្លៃនៃកូរ៉េឡង់រវាងតម្លៃលម្អៀង  $\epsilon$  និងតម្លៃទស្សន៍ទាយនៃអថេរពន្យល់  $x$  យើងបានលទ្ធផលដូចខាងក្រោម។ (សម្រាយបញ្ជាក់ទុកជាលំហាត់សម្រាប់អ្នកអាន)

$$Cov[\hat{y}, \epsilon] = 0$$

$$Cov[x, \epsilon] = 0$$

លទ្ធផលនេះបង្ហាញពីឯករាជភាពនៃតម្លៃលម្អៀងដែលបង្កើតដោយម៉ូដែលនិងតម្លៃទស្សន៍ទាយនៃអថេរគោលដៅ ឬ អថេរពន្យល់។

### 1.4 Coefficient of determination ( $R^2$ )

ដើម្បីបង្ហាញពីកម្រិតនៃការពន្យល់របស់ម៉ូដែលទៅលើទំនាក់ទំនងរវាងទិន្នន័យដែលមាន Coefficient of determination ( $R^2$ ) ត្រូវបានប្រើ។ តម្លៃ  $R^2$  កំណត់ដោយផលធៀបរវាង "រ៉ាងនៃតម្លៃទស្សន៍ទាយរបស់អថេរគោលដៅ និង "រ៉ាងនៃតម្លៃអថេរគោលដៅពិត។

$$R^2 = \frac{Var[\hat{y}]}{Var[y]} = 1 - \frac{Var[\epsilon]}{Var[y]}$$

តម្លៃនេះយកតម្លៃលើចន្លោះ  $[0,1]$  ដែលតម្លៃខិតជិត១បង្ហាញពីភាពល្អប្រសើរនៃការពន្យល់របស់ម៉ូដែលទៅលើទិន្នន័យ។



## 2. ការសិក្សាលើទំនាក់ទំនងរវាងច្រើនអថេរ

ក្នុងការសិក្សាលើទំនាក់ទំនងរវាងអថេរច្រើន ចំនួនអថេរពន្យល់អាចមានលើសពី១ ។  
ក្នុងករណីនេះយើងសន្មតម៉ូដែលដូចខាងក្រោម ។

$$y = f(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_d x_d$$

ជាឧទាហរណ៍ដូចជាការសិក្សាទំនាក់ទំនងរវាងម៉ាស់ (kg) ជាអថេរគោលដៅ និង កម្ពស់ (cm) ភេទ (0/1) ទំហំបង្កេះ (cm) ជាអថេរពន្យល់ជាដើម ។

ទោះបីជាចំនួននៃអថេរពន្យល់មានការកើនឡើងក្តី ការវិភាគដោយប្រើម៉ូដែលតម្រូវតម្រង់លីនេអ៊ែរ មិនមានអ្វីប្រែប្រួលជាដុំនោះឡើយ ។ អ្នកអាចបង្ហាញម៉ូដែលខាងលើជាទម្រង់វ៉ិចទ័រនិងម៉ាទ្រីសរួចសិក្សា ដូចគ្នា ។

$$X = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1d} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & x_{N1} & \cdots & x_{Nd} \end{pmatrix} \in \mathbb{R}^{N \times (d+1)}, \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix} \in \mathbb{R}^N, \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_N \end{pmatrix} \in \mathbb{R}^N, \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_d \end{pmatrix} \in \mathbb{R}^{d+1}$$

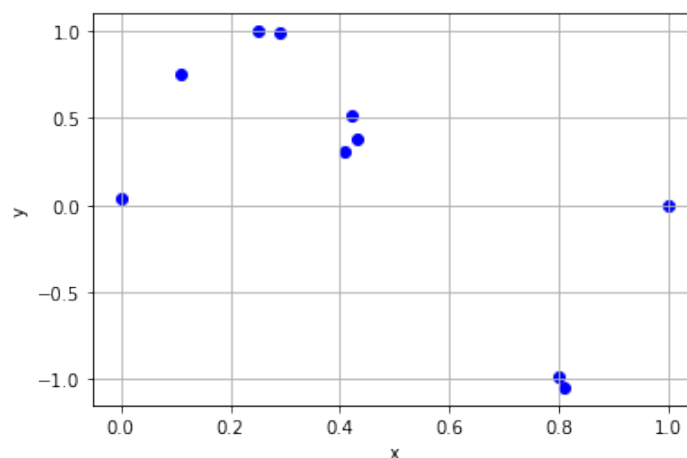
ពេលនេះម៉ូដែលនិងផលបូកតម្លៃការនែលម្យ៉ាងខាងលើអាចសរសេរដូចខាងក្រោម ។

$$\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad E(\boldsymbol{\beta}) = \sum_{i=1}^N \epsilon_i^2 = (\mathbf{y} - X\boldsymbol{\beta})^T (\mathbf{y} - X\boldsymbol{\beta})$$

ដោយប្រើវិធីសាស្ត្រ Least Square Error ដូចខាងលើយើងបានលទ្ធផលដូចគ្នា ។

$$\hat{\boldsymbol{\beta}} = (X^T X)^{-1} X^T \mathbf{y}$$

ក្នុងករណីខ្លះការប្រើប្រាស់តម្លៃផ្ទាល់នៃអថេរពន្យល់មិនអាចពណ៌នាទំនាក់ទំនងរវាងអថេរគោលដៅ និងអថេរពន្យល់បានល្អឡើយ ។ ហេតុនេះការបង្កើតតម្លៃអថេរពន្យល់ត្រូវបានអនុវត្ត ។



រូបទី៤ របាយទិន្នន័យដែលមិនអាចពន្យល់ដោយតម្លៃអថេរពន្យល់ផ្ទាល់

ជាឧទាហរណ៍ដូចជាការប្រើអនុគមន៍ដឺក្រេទី២ ឬខ្ពស់ជាងនេះ ឬការប្រើអនុគមន៍មិនមែនលីនេអ៊ែរ ជាដើម។ ក្នុងករណីនេះម៉ូដែលតម្រូវតម្រង់លីនេអ៊ែរអាចបង្ហាញដូចខាងក្រោម។ នៅទីនេះទោះបីជាអនុគមន៍  $\phi_i(x)$  ជាទម្រង់មិនមែនលីនេអ៊ែរក្តី ការហៅថាម៉ូដែលតម្រូវតម្រង់លីនេអ៊ែរ ព្រោះចង់សង្កត់ធ្ងន់លើផលបូកជា ទម្រង់លីនេអ៊ែរនៃអនុគមន៍ទាំងអស់នោះ។

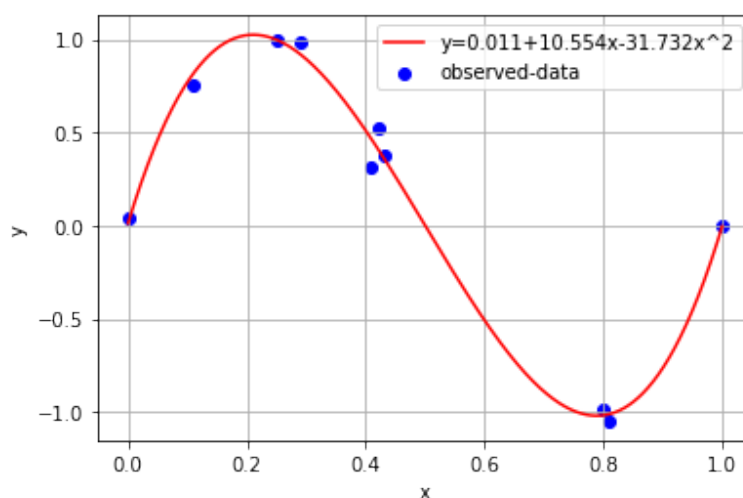
$$y = f(x) = \beta_0\phi_0(x) + \beta_1\phi_1(x) + \cdots + \beta_d\phi_d(x)$$

ករណីប្រើពហុធានីក្រេទី៣យើងបាន

$$y = \beta_0 + \beta_1x + \beta_2x^2 + \beta_3x^3$$

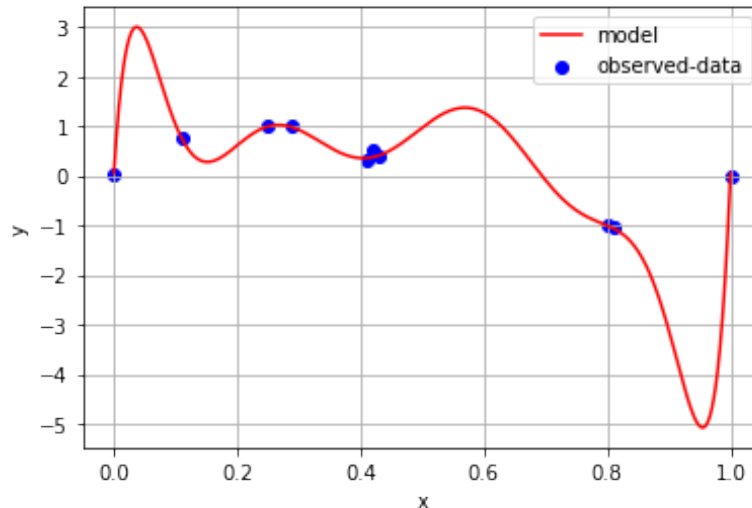
អ្នកអាចបង្ហាញម៉ូដែលខាងលើជាទម្រង់វ៉ិចទ័រនិងម៉ាទ្រីសរួចសិក្សាដូចគ្នា។

$$X = \begin{pmatrix} 1 & x_1 & x_1^2 & x_1^3 \\ \vdots & \vdots & \cdots & \vdots \\ 1 & x_N & x_N^2 & x_N^3 \end{pmatrix} \in \mathbb{R}^{N \times 4}, \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix} \in \mathbb{R}^N, \boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_N \end{pmatrix} \in \mathbb{R}^N, \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_3 \end{pmatrix} \in \mathbb{R}^4$$



រូបទី៥ របាយទិន្នន័យនិងខ្សែកោងម៉ូដែលតម្រូវតម្រង់ជាពហុធានីក្រេទី៣

តាមការសង្កេត អ្នកអាចនឹងមើលឃើញថាប្រសិនបើយើងតម្លើងដីក្រៃនៃពហុធានោះកម្រិតនៃការពណ៌នារបស់ម៉ូដែលលើទិន្នន័យនឹងមានការកើនឡើង។ ប៉ុន្តែការបង្ហាញម៉ូដែលដែលមានភាពស្មុគស្មាញពេកអាចត្រឹមតែពណ៌នាលើទិន្នន័យដែលមានតែប៉ុណ្ណោះ ផ្ទុយទៅវិញវាមិនអាចទស្សន៍ទាយបានល្អឡើយចំពោះទិន្នន័យដែលមិនមានក្នុងដៃ។ រូបភាពទី៥បង្ហាញករណីនេះ។



### រូបទី៦ របាយទិន្នន័យនិងខ្សែកោងម៉ូដែលតម្រេតប្រុងជាពហុធានីក្រេទី៩

ក្នុងរូបទី៤នេះ បើយើងគណនា Coefficient of determination ( $R^2$ ) យើងនឹងបានតម្លៃ 1 ដែលជាតម្លៃយ៉ាងល្អក្នុងការពណ៌នាទិន្នន័យដែលមានក្នុងដៃ។ ប៉ុន្តែបើយើងសង្កេតលើគម្រោងទិន្នន័យជាក់ស្តែង និងក្រាបនៃម៉ូដែល តម្លៃនៃការទស្សន៍ទាយត្រង់តំបន់ដែលគ្មានទិន្នន័យ គឺគ្មានលំនឹង និងចេញផុតឆ្ងាយពីដែននៃទិន្នន័យដែលមានក្នុងដៃ។

ហេតុនេះក្នុងការជ្រើសរើសម៉ូដែល អ្នកគួរសង្កេតលើចរិតលក្ខណៈនៃទិន្នន័យព្រមទាំងភាពល្អប្រសើរនៃការពន្យល់របស់វាចំពោះទាំងទិន្នន័យដែលមានក្នុងដៃស្រាប និងទិន្នន័យដែលមិនមានពេលគឺភាពប្រសើរក្នុងការទស្សន៍ទាយឬប៉ាន់ស្មាននាពេលអនាគត។

គណនាមេគុណតម្រេតប្រុងដោយប្រើម៉ូដែលជាពហុធានីក្រេទីk និងការទស្សន៍ទាយតម្លៃអថេរ គោលដៅជាមួយPython

```
import numpy as np
```

```
def fit(x,y,k):
    X_ = np.zeros((len(x),k+1))
    for i in range(k+1):
        X_[i,:] = x**i
    w = np.linalg.inv(X_.T@X_)@X_.T@y
    return w
```

```
def predict(x,w,k):
    X_ = np.zeros((len(x),k+1))
    for i in range(k+1):
        X_[i,:] = x**i
    return X_@w
```

# វិធីសាស្ត្របរមាភក្កតាមរយៈ SGD

( SGD: Stochastic Gradient Descent )

ក្នុងអត្ថបទមុនយើងបានណែនាំអំពីម៉ូដែលតម្រូវតម្រង់លីនេអ៊ែរ ដែលត្រូវបានប្រើប្រាស់សម្រាប់សិក្សាពីការទំនាក់ទំនងរវាងអថេរពន្យល់និងអថេរគោលដៅ ។

ក្នុងការកំណត់តម្លៃប៉ារ៉ាម៉ែត្រនៃម៉ូដែល ( មេគុណតម្រូវតម្រង់ )

យើងបានដោះស្រាយតាមរយៈវិធីសាស្ត្រជាមូលដ្ឋាននៃគណិតវិទ្យាវិភាគដូចខាងក្រោមនេះ ។

$$\text{ម៉ូដែល } y = X\beta + \epsilon$$

$$\text{មេគុណតម្រូវតម្រង់ } \hat{\beta} = (X^T X)^{-1} X^T y$$

ប៉ុន្តែក្នុងជីវភាពរស់នៅ ករណីភាគច្រើនចំនួននៃអថេរពន្យល់មានចំនួនច្រើនលើសលប់ ដែលធ្វើឱ្យវិមាត្រនៃម៉ាទ្រីសផែនការមានការកើនឡើងខ្ពស់ ។ ហេតុនេះ វាមានការលំបាកក្នុងការគណនាម៉ាទ្រីសប្រាស់ដូចក្នុងរបៀបខាងលើទោះបីប្រើប្រាស់ម៉ាស៊ីនកុំព្យូទ័រក្តី ។

ក្នុងអត្ថបទនេះ យើងនឹងណែនាំវិធីសាស្ត្រកំណត់តម្លៃប៉ាន់ស្មាននៃមេគុណតម្រូវតម្រង់ដោយវិធីគណនាដដែលៗលើតម្លៃលេខតាមប្រមាណវិធីងាយៗគឺ Stochastic Gradient Descent ( SGD ) ។

ដើម្បីងាយស្រួលស្វែងយល់អំពីSGD ជាដំបូងយើងនឹងណែនាំអំពីគំនិត និងការគណនាក្នុងវិធីសាស្ត្រ Gradient Descent ជាមុន ។

## 1. Gradient Descent

ដូចដែលបានបង្ហាញក្នុងអត្ថបទមុន យើងចង់កំណត់យកមេគុណតម្រូវតម្រង់ណាដែលធ្វើឱ្យតម្លៃផលបូកការនែកម្រិតលម្អៀងតូចបំផុត ។ គោលគំនិតក្នុងGradient Descent គឺផ្លាស់ប្តូរតម្លៃនៃមេគុណតម្រូវតម្រង់ ( ប៉ារ៉ាម៉ែត្រ )

បន្តិចម្តងៗ ទៅតាមទិសដៅដែលធ្វើឱ្យតម្លៃផលបូកការនែកម្រិតលម្អៀងមានការថយចុះ ។ អ្នកអាចប្រដូចវិធីនេះទៅនឹងការចុះដំរាលឬចុះពីទីភ្នំ ដោយរំកិលខ្លួនអ្នកបន្តិចម្តងៗ ទៅកាន់ទីដែលទាបជាងកន្លែងដែលអ្នកនៅ ។ នៅពេលដែលអ្នករំកិលខ្លួនដល់ទីដែលលែងមានបម្រែបម្រួលនៃរយៈកម្ពស់ អ្នកអាចសន្និដ្ឋានបានថាអ្នកដល់ទីដែលទាបបំផុតហើយ ។ ដូចគ្នានេះដែរ នៅក្នុងវិធីសាស្ត្រGradient Descent តាមលក្ខណៈគណិតវិទ្យានៃ gradient ( តម្លៃដេរីវេនៃអនុគមន៍ត្រង់ចំណុចណាមួយ ) តម្លៃgradientត្រង់ចំណុចណាមួយគឺជាតម្លៃមេគុណ

ប្រាប់ទិសនៃខ្សែកោងត្រង់ចំណុចនោះហើយក៏ជាតម្លៃដំបំផុតនៃបម្រែបម្រួលតម្លៃអនុគមន៍ពេល  
អ្នកធ្វើបម្រែបម្រួលលើអថេរមិនអាស្រ័យ ។



រូបទី១ គំនិតក្នុង Gradient Descent

រូបទី១បង្ហាញអំពីគំនិតក្នុងវិធីសាស្ត្រធ្វើអប្បបរមាកម្មតាម Gradient Descent ។ ដូចដែល  
អ្នកអាចធ្វើការកត់សម្គាល់បាន ពេលខ្លះអ្នកអាចនឹងធ្លាក់ចុះទៅក្នុងទីតាំងដែលជាបរមាជ្រៀបតែ  
មិនមែនជាកន្លែងអប្បបរមាពិតប្រាកដប្រសិនបើទីតាំងនៃការចាប់ផ្តើមរបស់អ្នក  
មិនប្រសើរ ។ ប៉ុន្តែក្នុងករណីធ្វើបរមាកម្មតម្លៃផលបូកការនៃកម្រិតលម្អៀងរបស់យើង ដោយសារ  
អនុគមន៍ដែលត្រូវធ្វើបរមាកម្មគឺជាអនុគមន៍ដឺក្រេទី២ ហេតុនេះយើងមិនមានការព្រួយបារម្ភក្នុងករណី  
នេះឡើយ ។

ពេលនេះ យើងពិនិត្យលើការគណនាក្នុងវិធីសាស្ត្រ Gradient Descent ។

យើងសិក្សាលើអនុគមន៍ដែលយកតម្លៃស្កាលែ  $f(x)$  ដែល  $x \in \mathbb{R}^d$  ។ សន្មតថាអនុគមន៍  
នេះយកតម្លៃអប្បបរមាត្រង់ចំណុច  $x^*$  ។ វិធីសាស្ត្រ Gradient Descent អាចឱ្យយើងគណនាតម្លៃ  
(ប្រហែល) នៃ  
 $x^*$  បានដោយចាប់ផ្តើមពីតម្លៃ  $x^{(0)}$  ណាមួយ រួចធ្វើការផ្លាស់ប្តូរតម្លៃនេះតាមការគណនាដូចខាងក្រោម  
។

$$x^{(t+1)} = x^{(t)} - \eta_t \left. \frac{\partial f(x)}{\partial x} \right|_{x=x^{(t)}}$$

នៅទីនេះ  $t = 0, 1, \dots$  គឺជាលេខរៀងនៃការផ្លាស់ប្តូរតម្លៃអថេរ  $x$  ។  $\frac{\partial f(x)}{\partial x}$  គឺជាដេរីវេដោយផ្នែក  
នៃអនុគមន៍  $f$  ធៀបនឹងអថេរ  $x$  ឬហៅថា gradient ។  $\eta_t$  គឺជាកម្រិតនៃការផ្លាស់ប្តូរតម្លៃអថេរដោយ  
គ្រប់គ្រងលើឥទ្ធិពលនៃតម្លៃ gradient ។ នៅក្នុង Machine Learning វាត្រូវបានហៅថា  
ជា អត្រារៀនឬ learning rate  
។ ជាទូទៅតម្លៃនៃ  $\eta_t$  ត្រូវបានកំណត់យកចន្លោះ ០ និង ១ ដោយតម្លៃយ៉ាងតូច ។

យើងអាចកំណត់លក្ខខណ្ឌសម្រាប់បញ្ចប់ការផ្លាស់ប្តូរតម្លៃនៃអថេរបាន ដោយយកពេលដែលតម្លៃដាច់ខាតនៃ gradient យកតម្លៃសូន្យឬក្បែរសូន្យ ។

ពិនិត្យលើករណីតម្រងាយមួយ  $f(x) = x^2 - 2x - 3$  ។ ករណីនេះយើងដឹងច្បាស់ថាតម្លៃអប្បបរមានៃអនុគមន៍គឺ  $-4$  នៅពេលដែល  $x^* = 1$  ។ យើងនឹងផ្ទៀងផ្ទាត់ជាមួយតម្លៃដែលគណនាតាមរយៈ Gradient Descent ។

ជំហានយើងគណនាអនុគមន៍ដេរីវេ  $\frac{df(x)}{dx} = 2x - 2$  និង កំណត់យកអត្រា  $\eta = 0.1$  បើ ។ យើងចាប់ផ្តើមពីចំណុច  $x^{(0)} = 0$  ,  $f(x^{(0)}) = -3$  ។ ដោយផ្លាស់ប្តូរតម្លៃអថេរតាមរយៈ Gradient Descent ខាងលើយើងបានបម្រែបម្រួលនៃតម្លៃអថេរនិងតម្លៃអនុគមន៍ដូចតារាងខាងក្រោម ។

តារាងទី១ បម្រែបម្រួលនៃតម្លៃអថេរនិងអនុគមន៍តាម Gradient Descent

$t$	$x^{(t)}$	$\frac{df(x)}{dx}$	$f(x)$
0	0.00	-2.00	-3.00
1	0.20	-1.60	-3.36
2	0.36	-1.28	-3.59
⋮	⋮	⋮	⋮
44	0.999946	-0.000109	-4.00
45	0.999956	-0.000087	-4.00

យើងត្រលប់ទៅកាន់ម៉ូដែលតម្រិតម្រង់របស់យើងវិញ ។ អនុគមន៍ដែលយើងចង់ធ្វើអប្បបរមាកម្មគឺ  $E(\beta)$  ដោយយក  $\beta$  ជាអថេរ ។

$$E(\beta) = \sum_{i=1}^N \epsilon_i^2 = (y - X\beta)^T (y - X\beta)$$

អនុគមន៍ដេរីវេ (gradient) របស់វាគឺ

$$E(\beta) = (y - X\beta)^T (y - X\beta) = y^T y - 2y^T X\beta + \beta^T X^T X\beta$$

$$\frac{\partial}{\partial \beta} E(\beta) = -2X^T y + 2X^T X\beta = 2X^T (X\beta - y) = 2X^T (\hat{y} - y)$$

ហេតុនេះ កន្សោមសម្រាប់ការផ្លាស់ប្តូរតម្លៃអថេរគឺ

$$\beta^{(t+1)} = \beta^{(t)} - \eta_t \frac{\partial E(\beta)}{\partial \beta} \Big|_{\beta=\beta^{(t)}}$$

$$\beta^{(t+1)} = \beta^{(t)} - 2\eta_t X^T (\hat{y}^{(t)} - y)$$

ដែល  $\hat{y}^{(t)} = X\beta^{(t)}$  ។

យើងសាកល្បងគណនាតម្លៃប្រហែលនៃមេគុណតម្រេតម្រង់ដែលបានសិក្សាក្នុងអត្ថបទមុន ដោយប្រើ gradient descent ។ លើកនេះយើងយកតម្លៃកម្ពស់គិតជាម៉ែត្រដើម្បីបង្រួមតម្លៃលេខ ។

តារាងទី២ ទិន្នន័យកម្ពស់(m)និងម៉ាស់(kg)មនុស្ស៩នាក់

កម្ពស់(m)	1.52	1.57	1.60	1.63	1.50	1.47	1.65	1.68	1.78
ម៉ាស់(kg)	54.48	55.84	57.20	58.57	53.12	52.21	59.93	61.29	69.92

ជាមួយPythonអ្នកអាចសរសេរCodeដូចខាងក្រោម ។

នៅទីនេះយើងកំណត់យកតម្លៃចាប់ផ្តើមនៃ  $\beta^{(0)} = \mathbf{0}$  និង  $\eta = 0.001$

---

```
import numpy as np
X = np.array([1.52,1.57,1.60,1.63,1.50,1.47,1.65,1.68,1.70])
y =
np.array([54.48,55.84,57.20,56.57,53.12,52.21,59.93,61.29,67.92])
XP = np.vstack([np.ones_like(X), X]).T
beta = np.zeros(XP.shape[1])

eta = 1e-3
for t in range(10000):
    y_hat = XP @ beta
    beta -= 2 * eta * XP.T @ (y_hat - y)
```

---

ជាលទ្ធផលយើងបានតម្លៃប្រហែលនៃមេគុណតម្រេតម្រង់គឺ

---

Beta  
array([-25.76358113, 52.40677129])

---

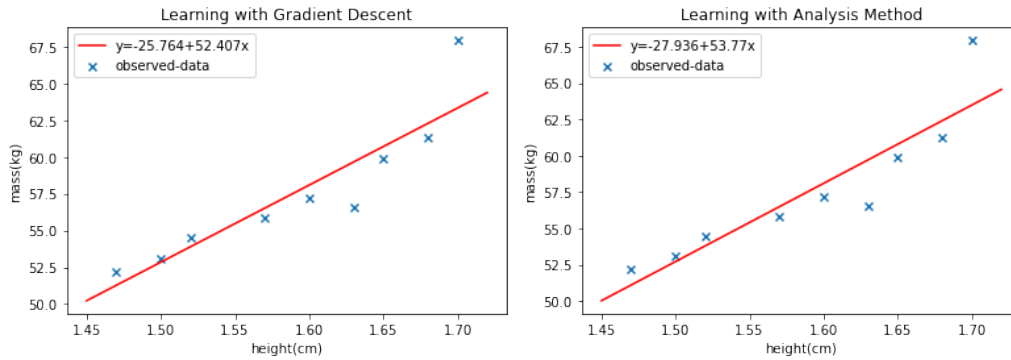
អ្នកអាចផ្ទៀងផ្ទាត់តម្លៃនេះតាមរយៈការគណនាដោយប្រើម៉ាទ្រីសប្រាស់ដូចក្នុងអត្ថបទមុនបាន ។  
ក្នុងករណីនេះអ្នកនឹងបានលទ្ធផល

---

Beta  
array([-27.93562969, 53.76959967])

---

តាមរយៈលទ្ធផលនេះយើងឃើញថា ការគណនាដោយប្រើ gradient descent អាចជួយ  
យើងឱ្យធ្វើការប៉ាន់ស្មានតម្លៃនៃមេគុណតម្រេតម្រង់បាន ។



រូបទី២ លទ្ធផលនៃតម្រូវប្រុងតាម Gradient Descent និង តាមការគណនាដោយគណិតវិទ្យាវិភាគ

## 2. Stochastic Gradient Descent

ការធ្វើបរមាមូលីតម្លៃអនុគមន៍ដោយប្រើ Gradient Descent អាចជួយយើងឱ្យធ្វើការគណនា

បានយ៉ាងមានប្រសិទ្ធភាពទោះបីជាវិមាត្រប្រចំនួននៃអថេរពន្យល់ច្រើនក៏ដោយ។ ប៉ុន្តែក្នុងវិធីសាស្ត្រ Gradient Descent ការគណនា gradient ត្រូវបានធ្វើឡើងដោយប្រើប្រាស់ទិន្នន័យទាំងអស់ដែលមានក្នុងដៃ។ ក្នុងករណីដែលចំនួនទិន្នន័យមានច្រើន វិធីនេះត្រូវបានគេដឹងថាមានភាពយឺតយ៉ាវក្នុងការរួមទៅរកតម្លៃបរមាមូលីតអនុគមន៍។

ដើម្បីដោះស្រាយបញ្ហានេះ Stochastic Gradient Descent (SGD) ត្រូវបានប្រើប្រាស់ជំនួសវិញ។ ក្នុងករណីចំនួនទិន្នន័យដែលមាន (N) មានបរិមាណច្រើន ក្នុងវិធីSGD ទិន្នន័យម្តងមួយៗ ត្រូវបានជ្រើសយកដោយចៃដន្យដើម្បីគណនា gradient នៃអនុគមន៍ រួចធ្វើការផ្លាស់ប្តូរតម្លៃអថេរតែម្តង ដោយមិនចាំបាច់ធ្វើការបូកសរុបគ្រប់ទិន្នន័យដែលមាននោះឡើយ។

ជាទូទៅ ដើម្បីអនុវត្តSGDបាន ចំពោះទិន្នន័យសរុបដែលមានអនុគមន៍ដែលត្រូវធ្វើបរមាមូលីត ត្រូវតែអាចសរសេរជាផលបូកនៃអនុគមន៍ដែលយកករណីទិន្នន័យនីមួយៗជាធាតុចូលដូចខាងក្រោម។

$$E_D(\beta) = \sum_{(x,y) \in D} e(\beta)$$

ក្នុងករណីយើងកំពុងសិក្សានេះ ដោយសារ  $E_D(\beta)$  ត្រូវបានកំណត់ដោយផលបូកការនែកម្រិតលម្អៀងគ្រប់ទិន្នន័យទាំងអស់  $E_D(\beta) = \sum_{i=1}^N \epsilon_i^2$  ហេតុនេះ លក្ខខណ្ឌខាងលើត្រូវបានផ្ទៀងផ្ទាត់។

ចំពោះទិន្នន័យនីមួយៗ  $(x_i, y_i)$  gradient នៃអនុគមន៍ដែលត្រូវធ្វើបរមាមូលីតអាចគណនាបានដូចខាងក្រោម។



$$\frac{\partial e(\beta)}{\partial \beta} = \frac{\partial}{\partial \beta} (y_i - x_i^T \beta)^2 = -2(y_i - x_i^T \beta) x_i^T = 2(\hat{y}_i - y_i) x_i^T$$

កន្សោមសម្រាប់ធ្វើការផ្លាស់ប្តូរតម្លៃនៃអថេរតាម SGD គឺអាចបង្ហាញដូចទម្រង់ខាងក្រោម។

$$\begin{aligned}\beta^{(t+1)} &= \beta^{(t)} - \eta_t \frac{\partial e(\beta)}{\partial \beta} \Big|_{\beta=\beta^{(t)}} \\ \beta^{(t+1)} &= \beta^{(t)} - 2\eta_t (\hat{y}_i^{(t)} - y) x_i^T \\ \beta^{(t+1)} &= \beta^{(t)} - 2\eta_t \delta_i\end{aligned}$$

ដែល  $\delta_i = (\hat{y}_i^{(t)} - y) x_i^T$  ។

ជាមួយ Python អ្នកអាចសរសេរ Code ដូចខាងក្រោម។

នៅទីនេះយើងកំណត់យកតម្លៃចាប់ផ្តើមនៃ  $\beta^{(0)} = \mathbf{0}$  និង  $\eta = 0.001$

---

```
import random
import numpy as np
```

---

```
X = np.array([1.52, 1.57, 1.60, 1.63, 1.50, 1.47, 1.65, 1.68, 1.70])
y =
np.array([54.48, 55.84, 57.20, 56.57, 53.12, 52.21, 59.93, 61.29, 67.92])
```

---

```
beta = np.zeros(2)
d_index = list(range(len(X)))

eta = 1e-3
for t in range(100000):
    random.shuffle(d_index)
    for i in d_index :
        XP = np.vstack([np.ones_like(X[i]), X[i]]).T
        y_hat = XP @ beta
        beta -= 2 * eta * XP.T @ (y_hat - y[i])
```

---

ជាលទ្ធផលយើងបានតម្លៃប្រហែលនៃមេគុណតម្រិតគឺ

---

```
Beta
array([-25.78979689,  52.42501619])
```

---

បើយើងធ្វើការប្រៀបធៀបរវាង Gradient Descent និង SGD យើងអាចនិយាយបានថា SGD គឺជាវិធីសាស្ត្រដែលសន្មតយកតម្លៃ gradient ចំពោះគ្រប់ទិន្នន័យទាំងអស់ក្នុង Gradient Descent ដោយតម្លៃប្រហែល  $\delta_i = (\hat{y}_i^{(t)} - y) x_i^T$  ពេលគឺ  $\frac{\partial E_D(\beta)}{\partial \beta} \approx \frac{\partial e_{x_i y_i}(\beta)}{\partial \beta} = \delta_i$  ។

## Regularization in Machine Learning

ក្នុងអត្ថបទមុនយើងបានស្វែងយល់អំពីម៉ូដែលតម្រូវប្រុងលីនេអ៊ែរនិងវិធីសាស្ត្រក្នុងការកំណត់តម្លៃមេគុណតម្រូវប្រុង (ប៉ារ៉ាម៉ែត្រ)

ដោយប្រើប្រាស់គណិតវិទ្យាវិភាគនិងវិធីសាស្ត្រប៉ាន់ស្មានតម្លៃតាម

វិធីសាស្ត្រSGD ។ ប៉ុន្តែបញ្ហាដែលនៅសល់គឺជាតើយើងគួរជ្រើសរើសយកម៉ូដែលបែបណាទើបអាចឱ្យវាពណ៌នាទំនាក់ទំនងរវាងទិន្នន័យបានល្អប្រសើរ ។ យើងពិនិត្យករណីខាងក្រោមជាឧទាហរណ៍ ។

សន្មតថាយើងមានទិន្នន័យដូចរូបទី១ (a) ។ យើងចង់បង្កើតម៉ូដែលតម្រូវប្រុងដើម្បីសិក្សាពីទំនាក់ទំនងរវាងអថេរពន្យល់និងអថេរគោលដៅ ។

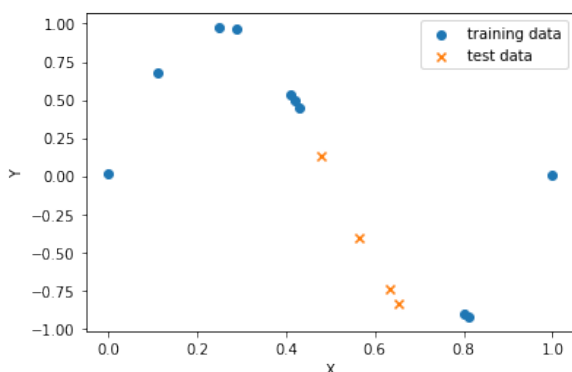
ដើម្បីសិក្សាពីភាពល្អប្រសើរនៃម៉ូដែល យើងបែងចែកទិន្នន័យជាពីរផ្នែកគឺ training

data ដែលប្រើសម្រាប់កំណត់ប៉ារ៉ាម៉ែត្រក្នុងម៉ូដែលនិង test data សម្រាប់ធ្វើការវាយតម្លៃ ។

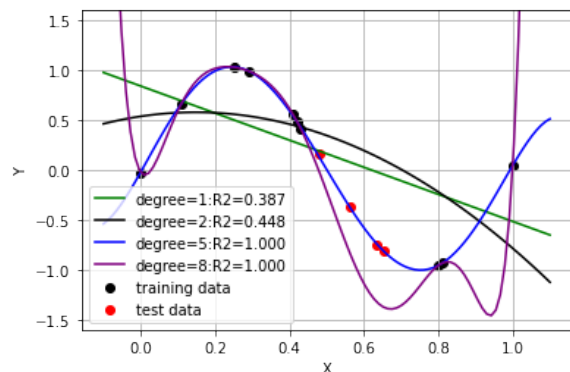
ក្នុងរូបទី១ (b) បង្ហាញពីលទ្ធផលនៅពេលដែលយើងអនុវត្តម៉ូដែលជាពហុធានីក្រេទី១ (បន្ទាត់) និងពហុធានីក្រេខ្ពស់ (ខ្សែកោង) ។ យើងអាចពិនិត្យឃើញថា នៅពេលយើងជ្រើសយកម៉ូដែលសាមញ្ញបំផុតពេលគឺបន្ទាត់ដ៏ក្រេទី១ នោះកម្រិតនៃការពណ៌នារបស់ម៉ូដែលទៅលើទិន្នន័យ

(Coefficient of determination:  $R^2$ ) មានតម្លៃទាបដែលបង្ហាញថាមិនអាចពន្យល់បានល្អឡើយចំពោះទិន្នន័យដែលមាន ។

ផ្ទុយទៅវិញ នៅពេលដែលយើងតម្លើងដ៏ក្រៃនៃម៉ូដែលកាន់តែខ្ពស់ យើងពិនិត្យឃើញថាតម្លៃនៃ  $R^2$  មានការកើនឡើងខ្លះដែលអាចឱ្យយើងនិយាយបានថាវាពន្យល់លើទំនាក់ទំនងរបស់ទិន្នន័យបានប្រសើរ ។ ប៉ុន្តែតើការតម្លើងម៉ូដែលឱ្យកាន់តែស្មុគស្មាញ (តម្លើងដ៏ក្រៃ) វាពិតជាផ្តល់ឱ្យយើងនូវម៉ូដែលដែលល្អមែនឬ ?



(a)



(b)

រូបទី១ ទិន្នន័យនិងម៉ូដែលតម្រូវប្រុង

នៅក្នុងរូបទី១(b)បើយើងពិនិត្យលើទិន្នន័យដែលមិនត្រូវបានប្រើក្នុងការកំណត់តម្លៃមេគុណតម្រិតប្រុង (test data) នោះយើងឃើញថា ម៉ូដែលដែលមានដឺក្រេលំដាប់ខ្ពស់ឬម៉ូដែលដែលស្មុគស្មាញខ្លាំងមិនអាចប៉ាន់ស្មាន ឬពន្យល់ទំនាក់ទំនងរវាងអថេរពន្យល់និងអថេរគោលដៅបានល្អឡើយបើប្រៀបធៀបជាមួយម៉ូដែលដែលមានដឺក្រេទាបជាងវា ។

ហេតុនេះ តើយើងគួរធ្វើបែបណាដើម្បីជ្រើសបានម៉ូដែលដែលអាចពន្យល់បានល្អទាំងចំពោះទិន្នន័យដែលប្រើក្នុងដំណាក់កាលកំណត់ប៉ារ៉ាម៉ែត្រ (learning) trainig data និងទាំងចំពោះទិន្នន័យដែលមិនត្រូវបានប្រើនៅដំណាក់កាល learning (test data) ?

## 1. Regularization

ដើម្បីដោះស្រាយបញ្ហានេះ Regularization ត្រូវបានប្រើប្រាស់។ ក្នុងវិធីសាស្ត្រនេះ ភាពស្មុគស្មាញនៃម៉ូដែលត្រូវបានគិតគូររួមគ្នាជាមួយនិងតម្លៃនៃកម្រិតល្អៀងរបស់ម៉ូដែល។ ឧទាហរណ៍ក្នុងករណីម៉ូដែលតម្រិតប្រុងលីនេអ៊ែរដែលយើងបានសិក្សាកន្លងមកនេះ ការធ្វើបម្រែបម្រួលលើតម្លៃលម្អៀងត្រូវបានផ្លាស់ប្តូរទៅជាទម្រង់  $L(\beta, \alpha)$  ដូចខាងក្រោម។ នៅទីនេះ ផ្នែក  $R(\beta)$  គឺជាផ្នែកដែលបង្ហាញពីកម្រិតនៃភាពស្មុគស្មាញរបស់ម៉ូដែល ហើយ  $\alpha$  ជាមេគុណដែលប្រើដើម្បីកម្រិតឥទ្ធិពលនៃ  $R(\beta)$  ពេលធ្វើបម្រែបម្រួល។

$$\begin{aligned} \text{ម៉ូដែល } y &= X\beta + \epsilon \\ L(\beta, \alpha) &= E(\beta) + \alpha R(\beta) \end{aligned}$$

ផ្នែក  $R(\beta) : \beta = (\beta_1, \dots, \beta_d)^T$  ដែលបង្ហាញពីកម្រិតនៃភាពស្មុគស្មាញរបស់ម៉ូដែលត្រូវបាន បង្ហាញជាទម្រង់នានាដូចជា

$$\text{Ridge penalty (L2 regularization) : } R(\beta) = \|\beta\|^2 = \beta_1^2 + \dots + \beta_d^2$$

$$\text{L1 regularization : } R(\beta) = |\beta_1 + \dots + \beta_d|$$

ក្នុងអត្ថបទនេះ យើងនឹងណែនាំអំពី  $L2 regularization$  ចំពោះម៉ូដែលតម្រិតប្រុងដែលហៅថា Ridge Regression Model ។

## 2. Ridge Regression Model

នៅក្នុង Ridge Regression Model តម្លៃការនៃណាមរបស់វ៉ិចទ័រមេគុណតម្រែតម្រង់ត្រូវបានប្រើប្រាស់សម្រាប់បង្ហាញពីកម្រិតស្មុគស្មាញរបស់ម៉ូដែល។ ក្នុងករណីនេះ ដើម្បីកំណត់តម្លៃមេគុណតម្រែតម្រង់ យើងនឹងធ្វើអប្បបរមាកម្មលើ អនុគមន៍ដែលកំណត់ដូចខាងក្រោម។

$$\begin{aligned} L(\boldsymbol{\beta}, \alpha) &= E(\boldsymbol{\beta}) + \alpha R(\boldsymbol{\beta}) \\ &= (\mathbf{y} - X\boldsymbol{\beta})^T (\mathbf{y} - X\boldsymbol{\beta}) + \alpha \|\boldsymbol{\beta}\|^2 \\ \hat{\boldsymbol{\beta}} &= \arg \min_{\boldsymbol{\beta}} L(\boldsymbol{\beta}, \alpha) \end{aligned}$$

តាមរយៈការធ្វើបរមាកម្មបែបនេះ យើងនឹងអាចកំណត់បាននូវម៉ូដែលដែលមានកម្រិតលម្អៀងតូចព្រមទាំងទំហំ នៃមេគុណតម្រែតម្រង់ (ដែលយើងសន្មតថាជាកម្រិតភាពស្មុគស្មាញក្នុងករណីនេះ) បានព្រមគ្នា។ នៅទីនេះដើម្បីធ្វើតុល្យកម្មរវាងកម្រិតលម្អៀងរបស់ម៉ូដែលនិងភាពស្មុគស្មាញ (ទំហំនៃមេគុណតម្រែតម្រង់) យើងអាចកែសម្រួលតម្លៃនៃមេគុណ  $\alpha$  បាន។

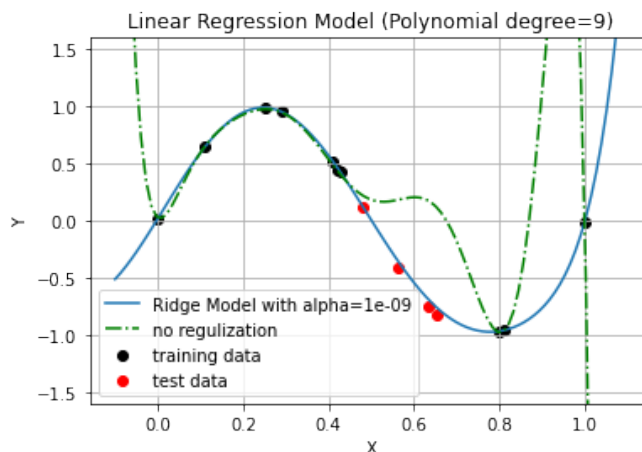
ដើម្បីដោះស្រាយបង្ហាញបរមាកម្មខាងលើ ដូចក្នុងអត្ថបទមុនៗដែរ យើងអាចដោះស្រាយតាមគណិតវិទ្យាវិភាគដោយធ្វើដេរីវេចរកតម្លៃនៃអថេរត្រង់ដេរីវេស្មើសូន្យ (ករណីអនុគមន៍ប៉ោង) ឬ ដោះស្រាយដោយប្រើវិធីសាស្ត្រSGD។

ចំពោះចម្លើយតាមគណិតវិទ្យាវិភាគ ទម្រង់នៃមេគុណតម្រែតម្រង់ត្រូវបានគណនាដូចខាងក្រោម (ដំណោះស្រាយទុកជាកិច្ចការផ្ទះជូនមិត្តអ្នកអាន) ដែល  $I_d$  ជាម៉ាទ្រីសឯកតា។

$$\hat{\boldsymbol{\beta}} = (X^T X + \alpha I_d)^{-1} X^T \mathbf{y}$$

ចំពោះចម្លើយតាមSGD ការផ្លាស់ប្តូរតម្លៃមេគុណតម្រែតម្រង់ត្រូវបានគណនាដូចខាងក្រោម (ដំណោះស្រាយទុកជាកិច្ចការផ្ទះជូនមិត្តអ្នកអាន) ដែល  $N$  ជាចំនួន training data និង  $\eta_t$  ជា learning rate ។

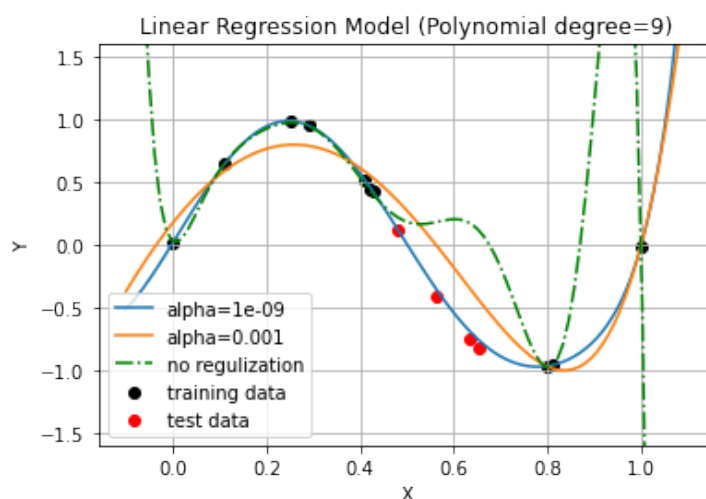
$$\boldsymbol{\beta}^{(t+1)} = \left(1 - \frac{2\alpha\eta_t}{N}\right) \boldsymbol{\beta}^{(t)} - 2\eta_t (\hat{\mathbf{y}}_i^{(t)} - y) \mathbf{x}_i^T$$



រូបទី២ ការប្រៀបធៀបរវាងម៉ូដែលដែលប្រើនិងមិនប្រើRegularization

រូបទី២បង្ហាញពីលទ្ធផលនៅពេល L2 regularization ត្រូវបានប្រើលើម៉ូដែលតម្រែតម្រង់ជាទម្រង់ពហុធានីក្រេទី៩ ។ យើងពិនិត្យឃើញថា នៅពេល regularization ត្រូវបានប្រើ ម៉ូដែលអាចពន្យល់បានល្អទាំងចំពោះ training data និង test data ព្រមគ្នា ផ្ទុយពីម៉ូដែលដែលមិនប្រើ regularization ។

តាមពិតទៅយើងនៅសល់បញ្ហាមួយទៀតគឺការកំណត់តម្លៃនៃមេគុណ  $\alpha$  ។ ដំណោះស្រាយក្នុងបញ្ហានេះអាចធ្វើបានតាមរយៈការសាកល្បងលើតម្លៃជាច្រើននៃ  $\alpha$  ចំពោះទិន្នន័យមួយផ្នែកដែលមិនមែនជា test data , training data ដែលយើងហៅថា validation data ។ យើងអាចកំណត់តម្លៃ  $\alpha$  ដោយជ្រើសយកតម្លៃ  $\alpha$  ណាដែលធ្វើឲ្យស្ថានភាពនៃម៉ូដែលល្អបំផុតចំពោះ validation data ។ រូបទី៣បង្ហាញពីការប្រៀបធៀបចំពោះតម្លៃមួយចំនួននៃ  $\alpha$  ។



រូបទី៣ ការប្រៀបធៀបលើមេគុណ  $\alpha$

ជាមួយPythonអ្នកអាចសរសេរCodeងាយៗដូចខាងក្រោម។

---

```
import numpy as np
```

---

ករណីដំណោះស្រាយតាមគណិតវិទ្យាវិភាគ

---

```
def ridge_fit(x,y,k,alpha):
    X_ = np.zeros((len(x),k+1))
    for i in range(k+1):
        X_[ :,i] = x**i
    beta = np.linalg.inv(X_.T@X_+alpha*np.eye(k+1))@X_.T@y
    return beta
```

---

ករណីដំណោះស្រាយតាមSGD

---

```
# learning with SGD
def ridge_sgd_fit(x,y,k,alpha):
    beta = np.zeros(k+1)
    d_index = list(range(len(x)))

    eta = 1e-4
    for t in range(500000):
        random.shuffle(d_index)
        for i in d_index :
            xi = np.zeros(k+1)
            for j in range(k+1):
                xi[j] = x[i]**j
            y_hat = xi.T @ beta
            beta = (1-2*alpha*eta/len(x))*beta - 2 * eta * (y_hat - y[i]) *
            xi
    return beta
```

---

## បញ្ហាចំណាត់ថ្នាក់២ក្រុម ( Binary Classification Problem )

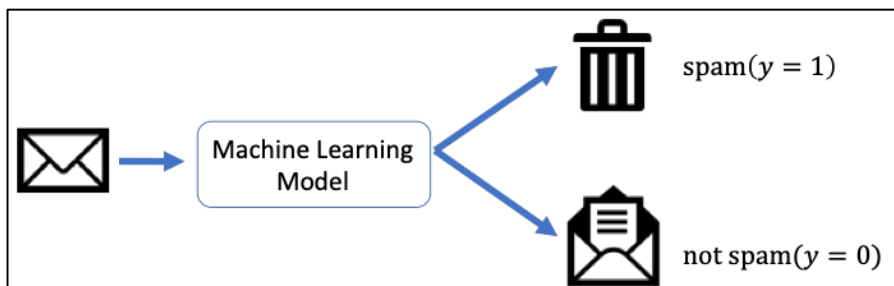
ក្រៅពីការសិក្សាពីទំនាក់ទំនងរវាងអថេរពិបាកច្រើនតាមរយៈម៉ូដែលតម្រូវតម្រង់ ការបែងចែកទិន្នន័យដោយផ្អែកលើអថេរពន្យល់របស់វាជាក្រុមហៅថាចំណាត់ថ្នាក់ក្រុម ( classification ) ។ តាមពិតចំណាត់ថ្នាក់ក្រុម

ក៏អាចបង្ហាញបានជាទម្រង់ម៉ូដែលតម្រូវតម្រង់ផងដែរ ដោយគ្រាន់តែតម្លៃនៃអថេរគោលដៅមិនយកតម្លៃទូលាយលើសំណុំចំនួនពិតឡើយ ផ្ទុយទៅវិញគឺយកតម្លៃជាចំនួនជា  $\{0,1\}$  ជាដើម ។

ក្នុងអត្ថបទនេះយើងនឹងណែនាំករណីដែលអថេរគោលដៅយកតម្លៃតែពីរប្រភេទដែលយើងកំណត់ហៅថាចំណាត់ថ្នាក់២ក្រុម ( binary classification ) ។ ឧទាហរណ៍នៃការអនុវត្តចំណាត់ថ្នាក់២ក្រុមក្នុងជីវភាពមានដូចជា៖ ការកំណត់អត្តសញ្ញាណសារអេឡិចត្រូនិចចំណោម ( spam or not ) ការវិភាគជម្ងឺតាមរយៈរោគសញ្ញា ( មានជម្ងឺឬគ្មាន )

ការទស្សន៍ទាយលទ្ធផលបោះឆ្នោត ( ជាប់ឬធ្លាក់ ) ជាដើម ។

ជាឧទាហរណ៍យើងនឹងលើកយកការកំណត់អត្តសញ្ញាណសារអេឡិចត្រូនិចចំណោម ( spam or not ) មកបង្ហាញដើម្បីស្វែងយល់បន្ថែមពីដំណោះស្រាយក្នុងបញ្ហាចំណាត់ថ្នាក់ក្រុមតាមរយៈ machine learning ។ រូបទី១បង្ហាញពីដំណើរការនៃការកំណត់សារចំណោម ។



រូបទី១ ការកំណត់spam mailដោយប្រើម៉ូដែលmachine learning

### 1. ចំណាត់ថ្នាក់២ក្រុមដោយម៉ូដែលលីនេអ៊ែរ

ម៉ូដែលលីនេអ៊ែរនៃចំណាត់ថ្នាក់២ក្រុមគឺជាម៉ូដែលដែលទស្សន៍ទាយប្រភេទនៃទិន្នន័យ  $\hat{y} \in \{0,1\}$  ដោយកំណត់តាមតម្លៃវិជ្ជមានឬអវិជ្ជមាននៃផលគុណស្កាលែររវាងធាតុចូល (input)  $x \in \mathbb{R}^d$  និងប៉ារ៉ាម៉ែត្រម៉ូដែល  $w \in \mathbb{R}^d$  ។

$$\hat{y} = \begin{cases} 1 & (x^T w > 0) \\ 0 & (x^T w \leq 0) \end{cases}$$

ក្នុងករណីការកំណត់spam mail យើងអាចកំណត់ជាទម្រង់ម៉ូដែលលីនេអ៊ែរខាងលើបាន ដោយកំណត់យក  $y = 1$  សម្គាល់spam mail និង  $y = 0$  សម្គាល់សារធម្មតា ។

ដើម្បីងាយស្រួលយល់ពីម៉ូដែលនេះ យើងមកមើលឧទាហរណ៍ងាយមួយក្នុងការកំណត់ spam

mailដូចខាងក្រោម។ សន្មតថា វ៉ិចទ័រធាតុចូល(អត្ថបទសារ)មានវិមាត្រ10។ យើងកំណត់វ៉ិចទ័រនេះ បង្ហាញពីវត្តមាននៃពាក្យគន្លឹះចំនួន១០ ដោយយកតម្លៃ០ ឬ 1 ត្រង់ពាក្យគន្លឹះនីមួយៗប្រសិនបើពាក្យ គន្លឹះនោះគ្មានឬមានវត្តមានក្នុងអត្ថបទសារ។

ឧទាហរណ៍ ពាក្យគន្លឹះដែលរៀបចំទុកមាន “assignment”, “boy”, “file”, “hello”, “love”, “my”, “photo”, “password”, “school”, “text”។ ក្នុងករណីនេះ អត្ថបទសារ “Hello my boy, I sent you my photo in the attached file” អាចបង្ហាញជាទម្រង់វ៉ិចទ័រ  $x \in \mathbb{R}^d$  បានដូចខាងក្រោម

$$x = (0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0)^T$$

ចំពោះការប៉ាន់ស្មានដោយម៉ូដែលលីនេអ៊ែរខាងលើ ចំពោះប៉ារ៉ាម៉ែត្រ  $w = (w_1 \quad \dots \quad w_{10})^T$  ផលគុណស្កាលែក្នុងករណីអត្ថបទសារខាងលើត្រូវបានគណនាដូចខាងក្រោម។ ក្នុងករណីនេះប្រសិនបើតម្លៃផលគុណស្កាលែនេះវិជ្ជមាននោះ អត្ថបទនេះត្រូវបានកំណត់ថាជាspam mail ។

$$x^T w = (0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0) \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_9 \\ w_{10} \end{pmatrix} = w_2 + w_3 + w_4 + w_6 + w_7$$

ប៉ុន្តែដូចដែលអ្នកកត់សម្គាល់បាន បញ្ហាសម្រាប់យើងគឺថា តើនឹងត្រូវកំណត់តម្លៃនៃ ប៉ារ៉ាម៉ែត្ររបស់ម៉ូដែលដោយរបៀបណា។ មានវិធីសាស្ត្រជាច្រើនត្រូវបានប្រើ តែក្នុងអត្ថបទនេះយើង នឹងណែនាំម៉ូដែលតម្រិតម្រង់Logistic ។

## 2. តម្រិតម្រង់Logistic (Logistic Regression)

តម្រិតម្រង់Logisticគឺជាម៉ូដែលលីនេអ៊ែរនៃចំណាត់ថ្នាក់២ក្រុមមួយប្រភេទ ដែល ប្រូបាបមាន លក្ខខណ្ឌ  $p(y|x)$  ពោលគឺប្រូបាបដែលប្រភេទនៃទិន្នន័យ  $y$  ត្រូវបានទស្សន៍ទាយចំពោះអថេរពន្យល់  $x$  ត្រូវបានគណនាដូចខាងក្រោម។

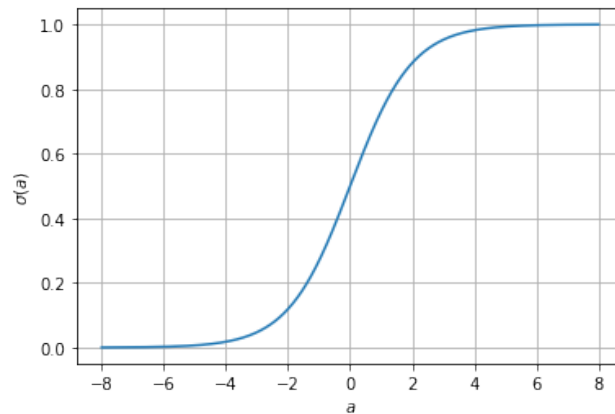


$$P(\hat{y} = 1|x) = \sigma(x^T w) = \frac{\exp(x^T w)}{1 + \exp(x^T w)} = \frac{1}{1 + \exp(-x^T w)}$$

$$P(\hat{y} = 0|x) = 1 - P(\hat{y} = 1|x) = 1 - \sigma(x^T w) = \sigma(-x^T w)$$

នៅទីនេះ  $\sigma(x)$  ជាអនុគមន៍ sigma ដែលត្រូវបានគណនាដូចខាងក្រោម។

$$\sigma(x) = \frac{\exp(x)}{1 + \exp(x)} = \frac{1}{1 + \exp(-x)}$$



រូបទី២ ក្រាបនៃអនុគមន៍ Sigmoid

ចំពោះទិន្នន័យ  $x$  ក្នុងករណីដែលប្រូបាប  $P(\hat{y} = 1|x) > 0.5$

នោះទិន្នន័យត្រូវបានទស្សន៍ទាយថាក្នុងចំណាត់ថ្នាក់ក្រុម  $\hat{y} = 1$  ។ ការកំណត់បែបនេះគឺសមមូលគ្នានឹងលក្ខខណ្ឌផលគុណស្កាលែវីជ្ជមានដែលបានបង្ហាញខាងដើម។

$$P(\hat{y} = 1|x) > 0.5 \Leftrightarrow \frac{1}{1 + \exp(-x^T w)} > \frac{1}{2} \Leftrightarrow \exp(-x^T w) < 1 \Leftrightarrow x^T w > 0$$

### 3. កម្រិតសាកសមនៃទិន្នន័យ Likelihood

ឧបមាថា ប៉ារ៉ាម៉ែត្រនៃម៉ូដែលតម្រេតម្រង់ Logistic ត្រូវបានកំណត់ដូចខាងក្រោម។

$$w = (-2 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad -1 \quad 0)^T$$

ក្នុងករណីនេះ តម្លៃផលគុណស្កាលែវីជ្ជមានរវាងវ៉ិចទ័រអថេរពន្យល់នៃទិន្នន័យ  $x$  និងប៉ារ៉ាម៉ែត្រអាច

គណនានិងបកស្រាយជាទម្រង់ប្រូបាបដូចខាងក្រោម។

$$\mathbf{x}^T \mathbf{w} = (0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0) \begin{pmatrix} -2 \\ 1 \\ \vdots \\ -1 \\ 0 \end{pmatrix} = 3$$

$$P(\hat{y} = 1|\mathbf{x}) = \sigma(3) = \frac{1}{1 + \exp(-3)} = 0.95$$

ចំពោះលទ្ធផលនេះយើងអាចបកស្រាយថា ចំពោះអត្ថបទសារដែលបានផ្តល់ ម៉ូដែលបានប៉ាន់ស្មានថាជាspam mailដោយតម្លៃប្រូបាប0.95 ។ តម្លៃនេះធំជាង0.5 ហេតុនេះយើងថាម៉ូដែលទស្សន៍ទាយថាវាជាspam mail ។

ក្នុងការពិភាក្សាខាងលើមកដល់ត្រឹមនេះ ចំពោះទិន្នន័យ $\mathbf{x}$ និងប៉ារ៉ាម៉ែត្រ $\mathbf{w}$  ដែលត្រូវបានផ្តល់ឲ្យការប៉ាន់ស្មានរបស់ម៉ូដែលត្រូវបានគណនាតាមវិធីដែលបានរៀបរាប់ខាងលើ។ ពេលនេះយើងពិនិត្យករណីដែលប្រភេទទិន្នន័យត្រូវបានកំណត់ជាក់លាក់ តែប៉ារ៉ាម៉ែត្រអាចត្រូវបានផ្លាស់ប្តូរ។ នៅទីនេះ យើងសិក្សាលើកម្រិតសាកសមនៃទិន្នន័យដែលម៉ូដែលប៉ាន់ស្មានបានដោយកំណត់ជាតម្លៃប្រូបាប $\hat{l}_{(x,y)}(\mathbf{w})$ ដូចខាងក្រោម និងសន្មតហៅថា កម្រិតសាកសម Likelihood។ ពេលគឺកម្រិតដែលម៉ូដែលអាចប៉ាន់ស្មានបានត្រឹមត្រូវ(គិតជាភាគរយ)លើប្រភេទទិន្នន័យនីមួយៗ។

$$\hat{l}_{(x,y)}(\mathbf{w}) = P(\hat{y} = y|\mathbf{x})$$

ឧទាហរណ៍ក្នុងករណីអត្ថបទសារខាងលើ សន្មតថាប្រភេទទិន្នន័យពិតគឺ  $y = 1$  ។ ចំពោះប៉ារ៉ាម៉ែត្រ $\mathbf{w}$  ដែលត្រូវបានផ្តល់ឲ្យ ការប៉ាន់ស្មានរបស់ម៉ូដែលខាងលើគឺ  $P(\hat{y} = 1|\mathbf{x}) = 0.95$

ហេតុនេះកម្រិតសាកសមLikelihood :  $\hat{l}_{(x,1)}(\mathbf{w}) = P(\hat{y} = 1|\mathbf{x}) = 0.95$

ដែលមានន័យថា ម៉ូដែលអាចបែងចែកថាជាspam

mailបានត្រឹមត្រូវក្នុងកម្រិត95%( ប្រូបាប0.95 ) ។ ផ្ទុយទៅវិញ ឧបមាថាយើងមានអត្ថបទសារមិនមែនspam mail មួយផ្សេង “Please submit your assignment file by tomorrow morning” ។

នោះវ៉ិចទ័រ  $\mathbf{x} = (1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0)^T$  ។

$$\mathbf{x}^T \mathbf{w} = (1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0) \begin{pmatrix} -2 \\ 1 \\ \vdots \\ -1 \\ 0 \end{pmatrix} = -1$$

$$P(\hat{y} = 1|\mathbf{x}) = \sigma(-1) = \frac{1}{1 + \exp(1)} = 0.27$$

ដោយចម្លើយពិតគឺមិនមែនជាspam mail ( $y=0$ ) ហេតុនេះកម្រិតសាកសមLikelihood :  
 $\hat{l}_{(x,0)}(\mathbf{w}) = P(\hat{y} = 0|\mathbf{x}) = 1 - P(\hat{y} = 1|\mathbf{x}) = 1 - 0.27 = 0.73$  ដែលមានន័យថាម៉ូដែលអាច  
 បែងចែកថាមិនមែនជាspam mailបានត្រឹមត្រូវក្នុងកម្រិត73%( ប្រូបាប0.73 ) ។

ចំពោះគ្រប់ប្រភេទទិន្នន័យ យើងអាចសរសេរកន្សោមកម្រិតសាកសមបានក្រោមទម្រង់

$$\hat{l}_{(x,y)}(\mathbf{w}) = P(\hat{y} = y|\mathbf{x}) = \begin{cases} P(\hat{y} = 1|\mathbf{x}) & (y = 1) \\ P(\hat{y} = 0|\mathbf{x}) & (y = 0) \end{cases} = \pi^y(1 - \pi)^{1-y}$$

ដែល  $\pi = P(\hat{y} = 1|\mathbf{x}) = \sigma(\mathbf{x}^T \mathbf{w})$  ។

#### 4. ការប៉ាន់ស្មានមេគុណតម្លៃតម្រង់តាម Maximum Likelihood

នៅក្នុងម៉ូដែលតម្រង់លីនេអ៊ែរ យើងបានកំណត់តម្លៃមេគុណតម្រង់ដោយធ្វើអប្បបរមាកម្មលើតម្លៃកម្រិតលម្អៀងរវាងតម្លៃប៉ាន់ស្មានដោយម៉ូដែលនិងតម្លៃពិតប្រាកដនៃអថេរគោលដៅ។ ស្រដៀងគ្នានេះ ក្នុងម៉ូដែលតម្រង់Logistic

យើងអាចកំណត់តម្លៃនៃប៉ារ៉ាម៉ែត្ររបស់ម៉ូដែលដោយធ្វើអតិបរមាកម្មលើកម្រិត

សាកសមរបស់ទិន្នន័យដែលម៉ូដែលអាចប៉ាន់ស្មានបាន ។ វិធីសាស្ត្រនេះត្រូវបានគេហៅថា

Maximum Likelihood Estimation (MLE) ។

ចំពោះសំណុំទិន្នន័យទាំងអស់ដែលមានចំនួន  $N$  យើងកំណត់កម្រិតសាកសមនៃទិន្នន័យ (likelihood) ពេលគឺកម្រិតដែលម៉ូដែលអាចប៉ាន់ស្មានបានត្រឹមត្រូវចំពោះគ្រប់ទិន្នន័យទាំងអស់ដោយកន្សោមខាងក្រោម ។ នៅទីនេះយើងសន្មតថា របាយនៃគ្រប់ទិន្នន័យទាំងអស់គឺឯករាជ្យនិងមានឯកសណ្ឋានភាព (i.i.d : independent and identically distributed) ។

$$\hat{L}_{\mathcal{D}}(\mathbf{w}) = \prod_{i=1}^N \hat{l}_{(x_i, y_i)}(\mathbf{w})$$

ដោយសារ កម្រិតសាកសមនៃទិន្នន័យ (likelihood) គឺជាតម្លៃប្រូបាប ហេតុនេះតម្លៃរបស់វាតូចខ្លាំង

ដែលធ្វើឱ្យតម្លៃផលគុណកាន់តែតូចខ្លាំងពេលចំនួនទិន្នន័យមានច្រើន ។ ដើម្បីបញ្ចៀសនូវបញ្ហាតម្លៃតូចពេកក្នុងការគណនាជាមួយកុំព្យូទ័រ នៅទីនេះយើងសិក្សាបរមាកម្មលើតម្លៃលោការីតរបស់វា ។

ការធ្វើបែបនេះមិនប៉ះពាល់ដល់ការធ្វើបរមាកម្មឡើយ

ព្រោះអនុគមន៍លោការីតជាអនុគមន៍កើនដាច់ខាត ។

$$\log \hat{L}_{\mathcal{D}}(\mathbf{w}) = \log \prod_{i=1}^N \hat{l}_{(x_i, y_i)}(\mathbf{w}) = \sum_{i=1}^N \log \hat{l}_{(x_i, y_i)}(\mathbf{w})$$

ដើម្បីងាយស្រួលក្នុងការដោះស្រាយបញ្ហាបរមាគម យើងប្តូរពីការធ្វើអតិបរមាគមលើ Likelihood ទៅជាការធ្វើអប្បបរមាគមដោយគុណកន្សោមខាងលើ -1 ។

$$\hat{\mathcal{L}}_{\mathcal{D}}(\mathbf{w}) = -\log \hat{L}_{\mathcal{D}}(\mathbf{w}) = -\sum_{i=1}^N \log \hat{l}_{(x_i, y_i)}(\mathbf{w})$$

ក្នុងករណីយើងចង់សិក្សាបន្ថែមដោយបញ្ចូលផ្នែក Regularization ( Ridge ) ចូលក្នុងម៉ូដែល Likelihood ដែលត្រូវធ្វើបរមាគម អាចប្តូរទៅសរសេរជាទម្រង់ខាងក្រោម ។

$$\hat{\mathcal{L}}_{\mathcal{D}}(\mathbf{w}) = -\log \hat{L}_{\mathcal{D}}(\mathbf{w}) + \alpha \|\mathbf{w}\|_2^2 = -\sum_{i=1}^N \log \hat{l}_{(x_i, y_i)}(\mathbf{w}) + \alpha \|\mathbf{w}\|_2^2 \quad (\alpha > 0)$$

## 5. ការដោះស្រាយតាមរយៈវិធី SGD

ដើម្បីធ្វើអប្បបរមាគម  $\hat{\mathcal{L}}_{\mathcal{D}}(\mathbf{w})$  យើងនឹងប្រើប្រាស់វិធី SGD ដែលបានសិក្សាក្នុងអត្ថបទមុន ។ ជាដំបូងយើងពិនិត្យលើអនុគមន៍ដេរីវេ  $\frac{\partial}{\partial \mathbf{w}} \log \hat{l}_{(x, y)}(\mathbf{w})$  ។

$$\log \hat{l}_{(x, y)}(\mathbf{w}) = \log(\pi^y (1 - \pi)^{1-y}) = y \log \pi + (1 - y) \log(1 - \pi)$$

$$\frac{\partial}{\partial \mathbf{w}} \log \hat{l}_{(x, y)}(\mathbf{w}) = \frac{y}{\pi} \frac{\partial \pi}{\partial \mathbf{w}} + \frac{1-y}{1-\pi} \times \left( -\frac{\partial \pi}{\partial \mathbf{w}} \right) = \frac{y - \pi}{\pi(1 - \pi)} \frac{\partial \pi}{\partial \mathbf{w}}$$

បន្ទាប់ពីនេះ ដើម្បីគណនា  $\frac{\partial \pi}{\partial \mathbf{w}}$  យើងពិនិត្យលើដេរីវេនៃអនុគមន៍ Sigmoid ។

$$\frac{\partial}{\partial a} \sigma(a) = \frac{\partial}{\partial a} \left\{ \frac{1}{1 + \exp(-a)} \right\} = -\frac{\frac{\partial}{\partial a} \exp(-a)}{(1 + \exp(-a))^2} = \frac{1}{1 + \exp(-a)} \times \frac{\exp(-a)}{1 + \exp(-a)}$$

$$\begin{aligned} \frac{\partial}{\partial a} \sigma(a) &= \sigma(a)(1 - \sigma(a)) \\ \frac{\partial \pi}{\partial \mathbf{w}} &= \frac{\partial}{\partial \mathbf{w}} P(\hat{y} = 1 | \mathbf{x}) = \frac{\partial}{\partial \mathbf{w}} \sigma(\mathbf{x}^T \mathbf{w}) = \sigma(\mathbf{x}^T \mathbf{w})(1 - \sigma(\mathbf{x}^T \mathbf{w})) = \pi(1 - \pi) \end{aligned}$$

នៅទីនេះ  $a = \mathbf{x}^\top \mathbf{w}$ ,  $\frac{\partial a}{\partial \mathbf{w}} = \frac{\partial}{\partial \mathbf{w}}(\mathbf{x}^\top \mathbf{w}) = \mathbf{x}$  ហេតុនេះ

$$\frac{\partial}{\partial \mathbf{w}} \log \hat{l}_{(x,y)}(\mathbf{w}) = \frac{y - \pi}{\pi(1 - \pi)} \frac{\partial \pi}{\partial \mathbf{w}} = \frac{y - \pi}{\pi(1 - \pi)} \frac{\partial \pi}{\partial a} \frac{\partial a}{\partial \mathbf{w}} = \frac{y - \pi}{\pi(1 - \pi)} \pi(1 - \pi) \mathbf{x}$$

$$\frac{\partial}{\partial \mathbf{w}} \log \hat{l}_{(x,y)}(\mathbf{w}) = (y - \pi) \mathbf{x}$$

ដូចនេះ តាមរយៈវិធីSGD តម្លៃនៃប៉ារ៉ាម៉ែត្រ  $\mathbf{w}$  ដែលធ្វើបម្រាស់កម្មលើតម្លៃកម្រិតសាកសមនៃទិន្នន័យត្រូវបានគណនាដោយផ្លាស់ប្តូរតម្លៃដូចខាងក្រោម។

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta_t \frac{\partial}{\partial \mathbf{w}} \{-\log \hat{l}_{(x,y)}(\mathbf{w})\} \Big|_{\mathbf{w}=\mathbf{w}^{(t)}}$$

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \eta_t \frac{\partial}{\partial \mathbf{w}} \log \hat{l}_{(x,y)}(\mathbf{w}) \Big|_{\mathbf{w}=\mathbf{w}^{(t)}}$$

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \eta_t (y - \pi^{(t)}) \mathbf{x}$$

នៅទីនេះ  $\pi^{(t)}$  ជាតម្លៃប្រូបាបដែលគណនាដោយម៉ូដែលជាមួយតម្លៃប៉ារ៉ាម៉ែត្រនៅដំណាក់កាល  $t$  នៃការផ្លាស់ប្តូរតម្លៃប៉ារ៉ាម៉ែត្រ ពោលគឺ  $\pi^{(t)} = \sigma(\mathbf{x}^\top \mathbf{w}^{(t)})$  ។  $\eta_t$  ជា learning-rate និង  $y$  ជាចំណាត់ថ្នាក់ក្រុមពិតប្រាកដនៃទិន្នន័យ  $\mathbf{x}$  ។

ក្នុងករណីប្រើ Ridge Regularization កន្សោមខាងលើនឹងប្រែទៅជាទម្រង់ខាងក្រោម។

$$\mathbf{w}^{(t+1)} = \left(1 - \frac{2\alpha\eta_t}{N}\right) \mathbf{w}^{(t)} + \eta_t (y - \pi^{(t)}) \mathbf{x}$$

## 6. ការវាយតម្លៃ

កាលពីសិក្សាម៉ូដែលតម្រេតម្រង់លីនេអ៊ែរ យើងវាយតម្លៃម៉ូដែលតាមរយៈតម្លៃកម្រិតលម្អៀង ឬ

មេគុណ $R^2$  ។ នៅករណីម៉ូដែលLogisticចំពោះបញ្ហាចំណាត់ថ្នាក់ក្រុមនេះ យើងអាចវាយតម្លៃតាម រយៈតម្លៃLikelihood បាន។ ប៉ុន្តែការសិក្សាលើតម្លៃLikelihood មានការពិបាកក្នុងការបកស្រាយ ភ្ជាប់នឹងជីវភាពរស់នៅរបស់យើង។ ហេតុនេះក្នុងបញ្ហាចំណាត់ថ្នាក់ក្រុមរង្វាស់សម្រាប់រង្វាយតម្លៃលើ ម៉ូដែលត្រូវបានគណនាដូចខាងក្រោម។

ចំណាត់ថ្នាក់ក្រុមពិតប្រាកដនៃទិន្នន័យសម្រាប់វាយតម្លៃ				
ចំណាត់ថ្នាក់ក្រុម ទស្សន៍ទាយដោយ ម៉ូដែល		$y = 1$	$y = 0$	សរុប
	$y = 1$	TP ( True Positive )	FP ( False Positive )	ចំនួនករណីដែល ត្រូវបានទស្សន៍ទាយថា $y = 1$
	$y = 0$	FN ( False Negative )	TN ( True Negative )	ចំនួនករណីដែល ត្រូវបានទស្សន៍ទាយថា $y = 0$
	សរុប	ចំនួនទិន្នន័យ $y = 1$	ចំនួនទិន្នន័យ $y = 0$	ចំនួនទិន្នន័យសរុប

- $\text{Accuracy} = \frac{\text{ចំនួនករណីដែលម៉ូដែលទស្សន៍ទាយបានត្រឹមត្រូវ}}{\text{ចំនួនករណីសរុប}} = \frac{TP+TN}{TP+TN+FP+FN}$
- $\text{Precision} = \frac{\text{ចំនួនករណីដែលម៉ូដែលទស្សន៍ទាយថា } y=1 \text{ បានត្រឹមត្រូវ}}{\text{ចំនួនករណីដែលម៉ូដែលទស្សន៍ទាយថា } y=1} = \frac{TP}{TP+FP}$
- $\text{Recall} = \frac{\text{ចំនួនករណីដែលម៉ូដែលទស្សន៍ទាយថា } y=1 \text{ បានត្រឹមត្រូវ}}{\text{ចំនួនទិន្នន័យ } y=1} = \frac{TP}{TP+FN}$
- $\text{F1-score} = \frac{\text{Precision} \times \text{Recall}}{\frac{1}{2}(\text{Precision} + \text{Recall})} = \frac{2 \text{ Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$

Accuracyគឺជាបង្ហាញពីអត្រានៃការទស្សន៍ទាយបានត្រឹមត្រូវរបស់ម៉ូដែលដោយមិនបែងចែកចំណាត់ថ្នាក់ក្រុមរបស់ទិន្នន័យ។ អត្រានេះដូចគ្នានឹងពិន្ទុដែលអ្នកឆ្លើយសំណួរបានត្រឹមត្រូវក្នុងការប្រឡងណាមួយដែរ។

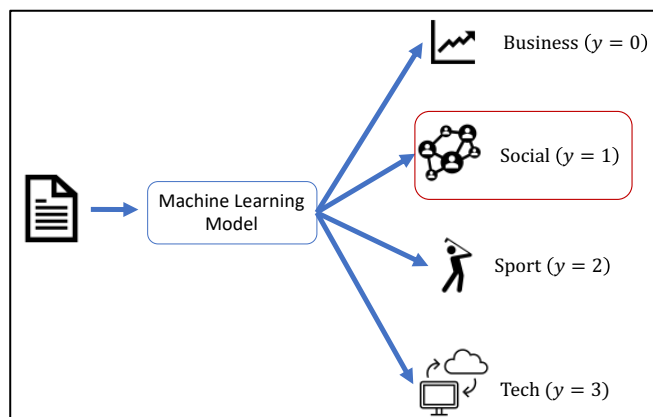
Precisionគឺសំដៅដល់អត្រានៃករណីដែលពិតជានៅក្នុងក្រុម  $y = 1$  មែនក្នុងចំណោមករណីដែលម៉ូដែលបានទស្សន៍ទាយថាស្ថិតក្នុងក្រុម  $y = 1$  ។ Recallសំដៅដល់អត្រានៃករណីដែលម៉ូដែលបានទស្សន៍ទាយថាស្ថិតក្នុងក្រុម  $y = 1$  ក្នុងចំណោមទិន្នន័យក្នុងក្រុម  $y = 1$  សរុប។

ជាទូទៅវាជាការលំបាកក្នុងការបង្កើតម៉ូដែលដែលមានទាំងPrecisionនិងRecallខ្ពស់ដូចគ្នា ( trade-off relation ) ។ ហេតុនេះដើម្បីវាយតម្លៃរួមលើរង្វាស់ទាំងពីរនេះការធ្វើផលធៀបមធ្យមលើតម្លៃទាំងពីរត្រូវបានប្រើពេលគឺតម្លៃ F1-score ។

## បញ្ហាចំណាត់ថ្នាក់ច្រើនក្រុម ( Multiclass Classification Problem )

នៅក្នុងជីវភាពរស់នៅ ការធ្វើចំណាត់ថ្នាក់ក្រុម មានករណីជាច្រើនដែលចំនួនក្រុមត្រូវកំណត់ មានច្រើនលើសពី២ ។ ឧទាហរណ៍ដូចជា ការបែងចែកអត្ថបទជាប្រភេទតាមប្រធានបទ ការធ្វើកំណត់ សម្គាល់ប្រភេទសម្ភារៈ ការកំណត់ប្រភេទវត្ថុយានយន្តលើផ្លូវរបស់យានយន្តបើកដោយស្វ័យប្រវត្តិ ជាដើម ។ ដូចទៅនឹងការធ្វើចំណាត់ថ្នាក់២ក្រុមដែរ តម្លៃនៃអថេរគោលដៅក្នុងករណីចំណាត់ថ្នាក់ ច្រើនក្រុមយកតម្លៃជាដូចជា  $\{0,1,2,3,\dots\}$  ។ ករណីដែលអថេរគោលដៅយកតម្លៃច្រើនប្រភេទ យើងកំណត់ហៅថាចំណាត់ថ្នាក់ច្រើនក្រុម ( multiclass classification ) ។

ក្នុងអត្ថបទនេះ យើងនឹងលើកយកការកំណត់ប្រភេទអត្ថបទតាមប្រធានបទមកបង្ហាញ ដើម្បីស្វែងយល់បន្ថែមពីដំណោះស្រាយក្នុងបញ្ហាចំណាត់ថ្នាក់ច្រើនក្រុមតាមរយៈ machine learning ។ រូបទី១បង្ហាញពីដំណើរការនៃការកំណត់សារវ៉ាន ។



រូបទី១ ការកំណត់ប្រភេទអត្ថបទតាមប្រធានបទដោយប្រើម៉ូដែល Machine Learning

### 1. ចំណាត់ថ្នាក់ច្រើនក្រុមដោយម៉ូដែលលីនេអ៊ែរ

ការបង្ហាញបញ្ហាចំណាត់ថ្នាក់ច្រើនក្រុមដោយប្រើម៉ូដែលលីនេអ៊ែរអាចធ្វើបានដូចករណីនៃ ចំណាត់ថ្នាក់២ក្រុមដែរ គឺជាម៉ូដែលដែលទស្សន៍ទាយប្រភេទនៃទិន្នន័យ  $\hat{y} \in C = \{0,1, \dots, K\}$  ដោយ កំណត់យកក្រុម  $j$  ណាដែលមានតម្លៃធំបំផុតនៃផលគុណស្កាលែរវាងធាតុចូល (input)  $\mathbf{x} \in \mathbb{R}^d$  និង ប៉ារ៉ាម៉ែត្រម៉ូដែល  $\mathbf{w}_j \in \mathbb{R}^d$  របស់ក្រុម  $j$  នោះ ។

$$\hat{y} = \arg \max_{j \in C} \mathbf{x}^T \mathbf{w}_j$$

ក្នុងករណីការកំណត់ប្រភេទអត្ថបទតាមប្រធានបទ យើងអាចកំណត់ជាទម្រង់ម៉ូដែល លីនេអ៊ែរខាងលើបានដោយ ជាឧទាហរណ៍កំណត់យក  $\hat{y} = 0$  សម្គាល់ប្រធានបទអំពី business,  $\hat{y} = 1$  សម្គាល់ប្រធានបទអំពី Social,  $\hat{y} = 2$  សម្គាល់ប្រធានបទអំពី Sport ជាដើម ។

ដើម្បីងាយស្រួលក្នុងការបកស្រាយខាងក្រោម យើងសន្មតថា អត្ថបទនីមួយៗត្រូវបាន បង្ហាញដោយវ៉ិចទ័រ  $\mathbf{x} \in \mathbb{R}^d$  ដែលកំប៉ូសង់នីមួយៗគឺជាប្រេកង់នៃពាក្យក្នុងវចនានុក្រមដែលមាននៅ ក្នុងអត្ថបទនោះ។ ដូចដែលអ្នកកត់សម្គាល់បាន បញ្ហាសម្រាប់យើងគឺថា តើនឹងត្រូវកំណត់តម្លៃនៃ ប៉ារ៉ាម៉ែត្ររបស់ម៉ូដែលដោយរបៀបណា។ ក្នុងអត្ថបទនេះយើងនឹងណែនាំ ម៉ូដែលតម្រេតម្រង់ Logistic សម្រាប់ចំណាត់ថ្នាក់ច្រើនក្រុម។

## 2. តម្រេតម្រង់ Logistic សម្រាប់ចំណាត់ថ្នាក់ច្រើនក្រុម ( Multiclass Logistic Regression )

តម្រេតម្រង់ Logistic សម្រាប់ចំណាត់ថ្នាក់ច្រើនក្រុមគឺជាម៉ូដែលលីនេអ៊ែរមួយប្រភេទ ដែល ប្រូបាបមានលក្ខខណ្ឌ  $p(\hat{y} = j | \mathbf{x})$  ពេលគឺប្រូបាបដែលប្រភេទនៃទិន្នន័យ  $\hat{y} = j$  ត្រូវបានទស្សន៍ទាយ ចំពោះអថេរពន្យល់  $\mathbf{x}$  ត្រូវបានគណនាដូចខាងក្រោម។

$$P(\hat{y} = j | \mathbf{x}) = \frac{\exp(\mathbf{x}^T \mathbf{w}_j)}{\sum_{j=0}^K \exp(\mathbf{x}^T \mathbf{w}_j)}$$

ដើម្បីសម្រួលក្នុងការសរសេរ នៅទីនេះយើងកំណត់សរសេរ  $a_j = \mathbf{x}^T \mathbf{w}_j$  ។ ចំពោះក្រុម  $j \in C = \{0, 1, \dots, K\}$  យើងកំណត់សរសេរវ៉ិចទ័រ  $\mathbf{a}$  ដែលជាតម្លៃផលគុណស្កាលែចំពោះក្រុមទាំងអស់ ដោយទម្រង់ខាងក្រោម។

$$\mathbf{a} = (a_0 \quad a_1 \quad \cdots \quad a_K) = (\mathbf{x}^T \mathbf{w}_0 \quad \mathbf{x}^T \mathbf{w}_1 \quad \cdots \quad \mathbf{x}^T \mathbf{w}_K)$$

ប្រូបាបមានលក្ខខណ្ឌដែលអត្ថបទ  $\mathbf{x}$  ត្រូវបានកំណត់ថាជាមួយក្រុម  $j : p(\hat{y} = j | \mathbf{x})$  អាចបង្ហាញតាមរយៈអនុគមន៍ Softmax បានដូចខាងក្រោមដែល  $\mathbf{a}_j$  សម្គាល់កំប៉ូសង់ទី  $j$  នៃវ៉ិចទ័រ  $\mathbf{a}$  ។

$$P(\hat{y} = j | \mathbf{x}) = \sigma(\mathbf{a})_j = \frac{\exp(a_j)}{\sum_{k=0}^K \exp(a_k)}$$



ចំពោះប៉ារ៉ាម៉ែត្រនៃក្រុមទាំងអស់ដែលមាន បើយើងកំណត់សរសេរដោយម៉ាទ្រីស  $W$  , ប្រូបាបមានលក្ខខណ្ឌចំពោះការធ្វើចំណាត់ថ្នាក់ក្រុមនីមួយៗដោយ  $\pi_j$  នោះយើងអាចបង្ហាញ ការគណនាខាងលើបានដូចខាងក្រោម ។

$$\mathbf{a} = (\mathbf{x}^\top \mathbf{w}_0 \quad \mathbf{x}^\top \mathbf{w}_1 \quad \cdots \quad \mathbf{x}^\top \mathbf{w}_K) = \mathbf{x}^\top W$$

$$(\pi_0 \quad \pi_1 \quad \cdots \quad \pi_K) = (\sigma(\mathbf{x}^\top \mathbf{w}_0)_0 \quad \sigma(\mathbf{x}^\top \mathbf{w}_1)_1 \quad \cdots \quad \sigma(\mathbf{x}^\top \mathbf{w}_K)_K)$$

$$\boldsymbol{\pi} = (\pi_0 \quad \pi_1 \quad \cdots \quad \pi_K) = \sigma(\mathbf{x}^\top W)$$

ឧទាហរណ៍ថាចំពោះ ទិន្នន័យ  $\mathbf{x}$  ប្រូបាបមានលក្ខខណ្ឌចំពោះក្រុមនីមួយៗត្រូវបានគណនា និងបានលទ្ធផល  $P(\hat{y} = 0|\mathbf{x}) = 0.1, P(\hat{y} = 1|\mathbf{x}) = 0.4, P(\hat{y} = 2|\mathbf{x}) = 0.2, P(\hat{y} = 3|\mathbf{x}) = 0.3$  នោះទិន្នន័យត្រូវបានទស្សន៍ទាយថាក្នុងចំណាត់ថ្នាក់ក្រុម  $\hat{y} = 1$  ព្រោះប្រូបាប  $P(\hat{y} = 1|\mathbf{x})$  មាន តម្លៃខ្ពស់ជាងគេ ។

ដើម្បីងាយស្រួលក្នុងការសរសេរនិងគណនាកម្រិតនៃភាពសាកសមរបស់ទិន្នន័យ ភាគច្រើន ចំណាត់ថ្នាក់ក្រុមរបស់ទិន្នន័យត្រូវបានបង្ហាញជាទម្រង់ one-hot vector ។ ឧបមាថាទិន្នន័យ  $\mathbf{x}$  ស្ថិតនៅក្នុងក្រុម 1 នោះវ៉ិចទ័រតាងចំណាត់ថ្នាក់ក្រុមរបស់វាត្រូវបានសរសេរដោយ

$$\mathbf{y} = (0 \quad 1 \quad 0 \quad \cdots \quad 0)^\top$$

ពេលគឺទិន្នន័យស្ថិតនៅក្រុម  $j$  ត្រូវបានសរសេរដោយវ៉ិចទ័រ  $\mathbf{y}$  ដែលកំប៉ូសង់ទី  $j$  របស់វាកំណត់ដោយ 1 និង 0 ចំពោះកំប៉ូសង់ក្រៅពីនេះ ។

$$y_k = \begin{cases} 1 & (k = j) \\ 0 & (k \neq j) \end{cases}$$

### 3. កម្រិតសាកសមនៃទិន្នន័យ Likelihood

គំនិតនៃការបង្ហាញកម្រិតសាកសមរបស់ទិន្នន័យក្នុងម៉ូដែលតម្រិតម្រង់ Logistic សម្រាប់ ចំណាត់ថ្នាក់ច្រើនក្រុម គឺដូចគ្នាទៅនឹងករណីចំណាត់ថ្នាក់ ២ ក្រុមដែរ ។ ចំពោះទិន្នន័យ  $(\mathbf{x}, \mathbf{y})$  និង ប៉ារ៉ាម៉ែត្រនៃម៉ូដែល  $W$  កម្រិតសាកសមនៃទិន្នន័យ (Likelihood) ត្រូវបានកំណត់ដូចខាងក្រោម ។

$$\hat{l}_{(\mathbf{x}, \mathbf{y})}(W) = P(\hat{y} = j|\mathbf{x}) = \pi_j$$

ដោយប្រើទម្រង់ one-hot vector នៃចំណាត់ថ្នាក់ក្រុមរបស់ទិន្នន័យ  $y \in \mathbb{R}^K$  យើងអាចសរសេរដូចខាងក្រោម ។

$$\hat{l}_{(x,y)}(W) = \pi_j = \prod_{k=0}^K \begin{cases} \pi_k & (y_k = 1) \\ 1 & (y_k = 0) \end{cases} = \prod_{k=0}^K \pi_k^{y_k}$$

ចំពោះទិន្នន័យ Training ទាំងអស់  $\mathcal{D}$  ដែលមាន Likelihood អាចសរសេរបានក្រោមទម្រង់

$$\hat{L}_{\mathcal{D}}(W) = \prod_{(x,y) \in \mathcal{D}} \hat{l}_{(x,y)}(W)$$

ដើម្បីបញ្ចៀសនូវបញ្ហាតម្លៃតូចពេកក្នុងការគណនាជាមួយកុំព្យូទ័រ នៅទីនេះយើងសិក្សាបមាគមលើតម្លៃលោការីតរបស់វា ។ ការធ្វើបែបនេះមិនប៉ះពាល់ដល់ការធ្វើបមាគមឡើយ ព្រោះអនុគមន៍លោការីតជាអនុគមន៍កើនជាប់ខាត ។

$$\hat{\mathcal{L}}_{\mathcal{D}}^{MLE}(W) = -\log \hat{L}_{\mathcal{D}}(W) = -\log \prod_{(x,y) \in \mathcal{D}} \hat{l}_{(x,y)}(W) = -\sum_{(x,y) \in \mathcal{D}} \hat{l}_{(x,y)}(W)$$

#### 4. ការប៉ាន់ស្មានមេគុណតម្រិតប្រុងតាម Maximum Likelihood

ចំពោះសំណុំទិន្នន័យទាំងអស់  $\mathcal{D}$  ដែលមានចំនួន  $N$  យើងកំណត់កម្រិតសាកសមនៃទិន្នន័យ (likelihood) ពេលគឺកម្រិតដែលម៉ូដែលអាចប៉ាន់ស្មានបានត្រឹមត្រូវចំពោះគ្រប់ទិន្នន័យទាំងអស់ដោយកន្សោមខាងក្រោម ។ នៅទីនេះយើងសន្មតថា របាយនៃគ្រប់ទិន្នន័យទាំងអស់គឺឯករាជ្យនិងមានឯកសណ្ឋានភាព (i.i.d : independent and identically distributed) ។

$$\hat{\mathcal{L}}_{\mathcal{D}}(\mathbf{w}) = -\log \hat{L}_{\mathcal{D}}(\mathbf{w}) = -\sum_{i=1}^N \log \hat{l}_{(x_i, y_i)}(\mathbf{w})$$

## 5. ការដោះស្រាយតាមរយៈវិធី SGD

ដើម្បីធ្វើអប្បបរមាកម្ម  $\hat{\mathcal{L}}_{\mathcal{D}}(W)$  យើងនឹងប្រើប្រាស់វិធី SGD ដែលបានសិក្សាក្នុងអត្ថបទមុន ។

$$W^{(t+1)} = W^{(t)} - \eta_t \frac{\partial}{\partial W} \{-\log \hat{l}_{(x,y)}(W)\} \Big|_{W=W^{(t)}}$$

$$W^{(t+1)} = W^{(t)} + \eta_t \frac{\partial}{\partial W} \log \hat{l}_{(x,y)}(W) \Big|_{W=W^{(t)}}$$

ជាដំបូងយើងពិនិត្យលើអនុគមន៍ដេរីវេ  $\frac{\partial}{\partial W} \log \hat{l}_{(x,y)}(W)$  ។

$$\log \hat{l}_{(x,y)}(W) = \log \left( \prod_{k=0}^K \pi_k^{y_k} \right) = \sum_{k=0}^K y_k \log \pi_k$$

$$\frac{\partial}{\partial \mathbf{w}_j} \log \hat{l}_{(x,y)}(W) = \frac{\partial}{\partial \mathbf{w}_j} \sum_{k=0}^K y_k \log \pi_k = \sum_{k=0}^K \frac{y_k}{\pi_k} \frac{\partial \pi_k}{\partial \mathbf{w}_j}$$

នៅទីនេះយើងពិនិត្យលើអនុគមន៍បណ្តាក់  $\pi_k = \sigma(\mathbf{a})_k$ ,  $a_j = \mathbf{x}^\top \mathbf{w}_j$

$$\begin{aligned} \frac{\partial}{\partial \mathbf{w}_j} \log \hat{l}_{(x,y)}(W) &= \sum_{k=0}^K \frac{y_k}{\pi_k} \frac{\partial \pi_k}{\partial \mathbf{w}_j} = \sum_{k=0}^K \frac{y_k}{\pi_k} \frac{\partial \pi_k}{\partial a_j} \frac{\partial a_j}{\partial \mathbf{w}_j} \\ &= \sum_{k=0}^K \frac{y_k}{\pi_k} \{ \pi_k (\delta_{kj} - \pi_j) \} \mathbf{x} \quad , \quad \delta_{kj} = \begin{cases} 1 & (k = j) \\ 0 & (k \neq j) \end{cases} \\ &= \mathbf{x} \sum_{k=0}^K y_k (\delta_{kj} - \pi_j) \\ &= \mathbf{x} \left( \sum_{k=0}^K y_k \delta_{kj} - \sum_{k=0}^K y_k \pi_j \right) = \mathbf{x} (y_j - \pi_j) \end{aligned}$$

$$\frac{\partial}{\partial \mathbf{w}_j} \log \hat{l}_{(x,y)}(W) = \mathbf{x} (y_j - \pi_j)$$

ដូចនេះ តាមរយៈវិធីSGD តម្លៃនៃប៉ារ៉ាម៉ែត្រ  $w_j$  នៃ  $w$  ដែលធ្វើបមាតិកម្មលើតម្លៃកម្រិតសាកសមនៃទិន្នន័យត្រូវបានគណនាដោយផ្លាស់ប្តូរតម្លៃដូចខាងក្រោម ។

$$w_j^{(t+1)} = w_j^{(t)} + \eta_t \frac{\partial}{\partial w_j} \log \hat{l}_{(x,y)}(W) \Big|_{w_j=w_j^{(t)}}$$

$$w_j^{(t+1)} = w_j^{(t)} + \eta_t (y_j - \pi_j^{(t)}) x$$

នៅទីនេះ  $\pi^{(t)}$  ជាតម្លៃប្រូបាបដែលគណនាដោយម៉ូដែលជាមួយតម្លៃប៉ារ៉ាម៉ែត្រនៅដំណាក់កាល  $t$  នៃការផ្លាស់ប្តូរតម្លៃប៉ារ៉ាម៉ែត្រ ពេលគឺ  $\pi^{(t)} = \sigma(x^T w^{(t)})$  ។  $\eta_t$  ជា learning-rate និង  $y$  ជាចំណាត់ថ្នាក់ក្រុមពិតប្រាកដនៃទិន្នន័យ  $x$  ។

ក្នុងករណីប្រើ Ridge Regularization កន្សោមខាងលើនឹងប្រែទៅជាទម្រង់ខាងក្រោម ។

$$w^{(t+1)} = \left(1 - \frac{2\alpha\eta_t}{N}\right) w^{(t)} + \eta_t (y - \pi^{(t)}) x$$

## 6. ការវាយតម្លៃ

កាលពីសិក្សាបញ្ចប់ចំណាត់ថ្នាក់២ក្រុម រង្វាស់សម្រាប់វាយតម្លៃលើម៉ូដែលត្រូវបានគណនាដោយប្រើ Accuracy, Precision, Recall, F1-

score ។ ក្នុងករណីចំណាត់ថ្នាក់ច្រើនក្រុម តម្លៃទាំងនេះក៏ត្រូវបានប្រើដើម្បីវាយតម្លៃម៉ូដែលដូចគ្នា ។

ចំពោះ Accuracy អាចគណនាបានដោយធ្វើផលធៀបចំនួនករណីដែលម៉ូដែលធ្វើការប៉ាន់ស្មានបានត្រឹមត្រូវ ធៀបនឹងចំនួនទិន្នន័យសរុប ។ ចំពោះ Precision, Recall, F1-score នៃក្រុមនីមួយៗ

ត្រូវបានគណនាដោយឡែកៗពីគ្នា រួចធ្វើតម្លៃមធ្យម Macro /Micro

ដូចខាងក្រោម ។ ជាមួយគ្នានេះការសិក្សាលើវាយតម្លៃដោយប្រើ Confusion Matrix ដូចក្នុងរូបទី២ ក៏ត្រូវបានប្រើផងដែរ ។

$$\text{Macro Precision} = \frac{\text{Precision}_0 + \text{Precision}_1 + \dots + \text{Precision}_K}{K}$$

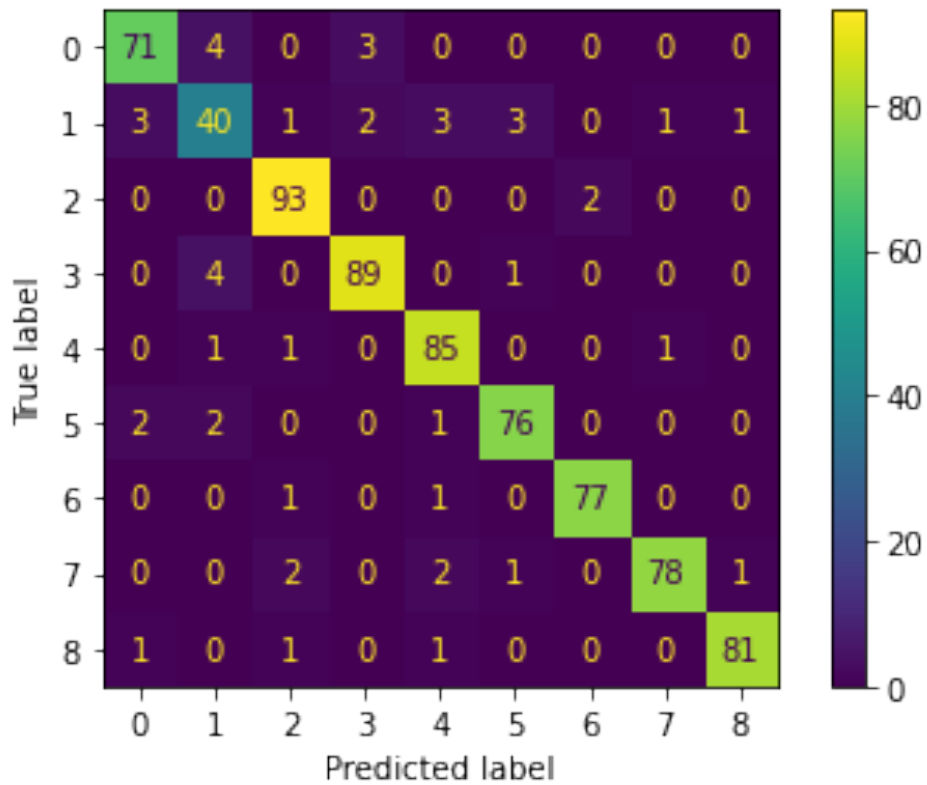
$$\text{Macro Recall} = \frac{\text{Recall}_0 + \text{Recall}_1 + \dots + \text{Recall}_K}{K}$$

$$\text{Macro F1 - score} = \frac{\text{F1}_0 + \text{F1}_1 + \dots + \text{F1}_K}{K}$$

$$\text{Micro Precision} = \frac{PA_0 + PA_1 + \dots + PA_K}{PB_0 + PB_1 + \dots + PB_K}, \quad \text{Precision}_k = \frac{PA_k}{PB_k}$$

$$\text{Micro Recall} = \frac{RA_0 + RA_1 + \dots + RA_K}{RB_0 + RB_1 + \dots + RB_K}, \quad \text{Recall}_k = \frac{RA_k}{RB_k}$$

$$\text{Micro F1 - score} = \frac{FA_0 + FA_1 + \dots + FA_K}{FB_0 + FB_1 + \dots + FB_K}, \quad \text{F1 - score}_k = \frac{FA_k}{FB_k}$$



រូបទី២ Confusion Matrix

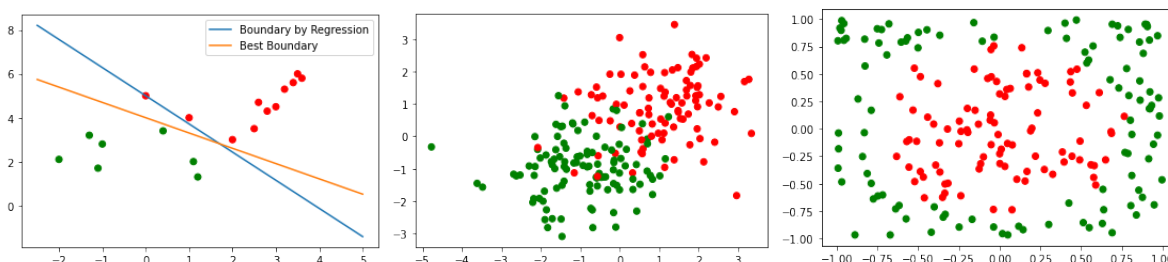
## Support Vector Machine

ក្នុងអត្ថបទមុន យើងបានណែនាំអំពីការធ្វើចំណាត់ថ្នាក់ទិន្នន័យ២ឬច្រើនក្រុមដោយប្រើម៉ូដែលតម្រូវម្រង់ (Linear regression, Logistic regression) ។ ប៉ុន្តែមានចំណុចខ្សោយសំខាន់ពីរកើតមានឡើងក្នុងការប្រើប្រាស់ម៉ូដែលតម្រូវម្រង់

ក្នុងការធ្វើចំណាត់ថ្នាក់ក្រុមទិន្នន័យ ដូចបង្ហាញក្នុងរូបទី១ខាងក្រោម ។

ក្នុងរូបខាងឆ្វេង ដោយសារមានទិន្នន័យប្រមូលផ្តុំច្រើននៅផ្នែកខាងក្រុមក្រហម ម៉ូដែលតម្រូវម្រង់លីនេអ៊ែរនឹងផ្តល់ឱ្យនូវបន្ទាត់ព្រំដែនដែលខិតទៅជិតក្រុមក្រហមខ្លាំង (បន្ទាត់ខៀវ) ។ ប៉ុន្តែតាមពិតបន្ទាត់ព្រំដែនទឹកក្រូចអាចមើលឃើញថាប្រសើរជាង ។ ក្នុងករណីរូបកណ្តាលនិងរូបខាងស្តាំ ច្បាស់ណាស់ថា ទិន្នន័យទាំងនេះមិនអាចធ្វើការបែងចែកដោយបន្ទាត់ត្រង់បានឡើយ ពោលគឺជាប្រភេទដែលមិនអាចបែងចែកលីនេអ៊ែរ (linear non-separable data) ។

ដើម្បីដោះស្រាយបញ្ហាទាំងនេះ ការធ្វើចំណាត់ថ្នាក់ក្រុមទិន្នន័យដោយប្រើSupport Vector Machineត្រូវបានប្រើ ។ ក្នុងអត្ថបទនេះយើងនឹងណែនាំអំពីដំណើរការគណិតវិទ្យាក្នុងការដោះស្រាយបញ្ហាធ្វើចំណាត់ថ្នាក់ក្រុម២ឬច្រើនក្រុមដោយប្រើSupport Vector Machine ។ យើងនឹងពិនិត្យទាំងករណីទិន្នន័យដែលអាចធ្វើបំណែងចែកលីនេអ៊ែរបាន (Linear Seperable) និងមិនបាន (Linear Non- Seperable) ។



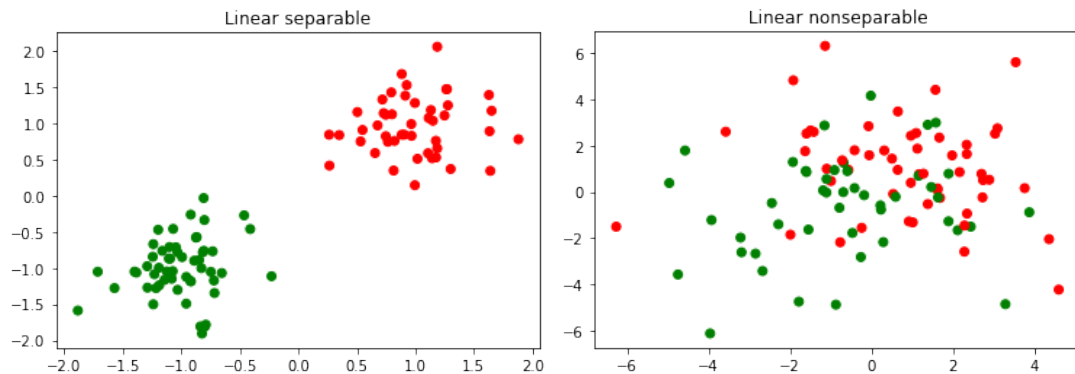
រូបទី១ ករណីធ្វើចំណាត់ថ្នាក់មិនបានល្អជាមួយLinear Regression

### 1. ចំណាត់ថ្នាក់២ក្រុម

នៅទីនេះយើងសិក្សាលើទិន្នន័យពីក្រុម  $(+1, -1)$   $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$  ដែល  $x_i \in \mathbb{R}^d$ ,  $y_i \in \{+1, -1\}$  ។ យើងចង់បង្កើតម៉ូដែលចំណាត់ថ្នាក់ក្រុមមួយដែលធ្វើការបែងចែកតាមម៉ូដែលលីនេអ៊ែរដែលតាងដោយទម្រង់ខាងក្រោម ។

$$\mathcal{M} = \{\text{sign}(f(x)) \mid f(x) = \mathbf{w}^T \mathbf{x} + b, \mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}\}$$

ក្នុងករណីដែលមាន  $m(x) = \text{sign}(w^T x + b)$  ដែលផ្ទៀងផ្ទាត់  $m(x_i) = y_i \quad \forall i = 1, 2, \dots, N$  នោះ ទិន្នន័យដែលមានហៅថា អាចធ្វើបំណែងចែកលីនេអ៊ែរបាន (Linear Seperable) ។ ពេលគឺ យើងអាចកំណត់បន្ទាត់ឬប្លង់ព័ងដើម្បីបែងចែកទិន្នន័យទាំងពីរប្រភេទបានច្បាស់លាស់ដោយគ្មាន កំហុសចំពោះគ្រប់ទិន្នន័យ ។ ផ្ទុយទៅវិញ ករណីដែលមិនអាចរកបានម៉ូដែលណាដែលផ្ទៀងផ្ទាត់លក្ខខណ្ឌខាងលើ យើងហៅថា មិនអាចធ្វើបំណែងចែកលីនេអ៊ែរបាន (Linear Non-Seperable) ។



រូបទី២ ទិន្នន័យដែលអាចបែងចែកលីនេអ៊ែរបាននិងទិន្នន័យដែលមិនអាចបែងចែកលីនេអ៊ែរបាន

### 1.1. ករណីទិន្នន័យដែលអាចធ្វើបំណែងចែកលីនេអ៊ែរបាន (Linear Seperable)

ក្នុងករណីទិន្នន័យដែលអាចធ្វើបំណែងចែកលីនេអ៊ែរបាន ចំពោះទិន្នន័យ  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$  យើងអាចសរសេរទំនាក់ទំនងខាងក្រោមបាន

$$y_i = +1 \Rightarrow w^T x_i + b > 0$$

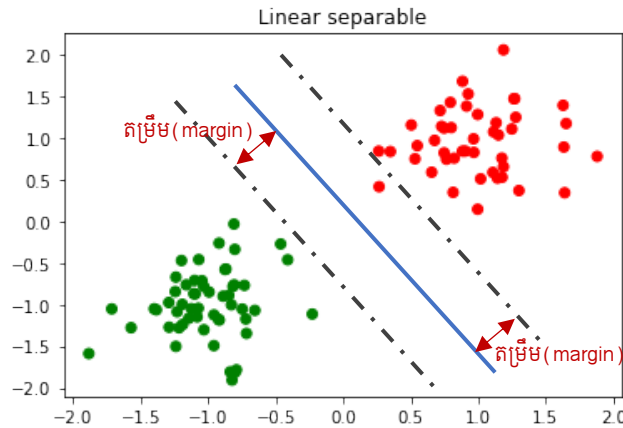
$$y_i = -1 \Rightarrow w^T x_i + b < 0$$

ឬ ជារួម

$$y_i(w^T x_i + b) > 0 \quad (i = 1, 2, \dots, N)$$

ជាទូទៅ តម្លៃនៃប៉ារ៉ាម៉ែត្រ  $(w, b)$  ដែលផ្ទៀងផ្ទាត់លក្ខខណ្ឌខាងលើមានច្រើនលើសពីមួយ ។ ក្នុងចំណោមនោះ ដើម្បីជ្រើសបានប៉ារ៉ាម៉ែត្រដែលប្រសើរ គោលគំនិតក្នុង Support Vector Machine គឺកំណត់យកករណីដែលតម្រឹម (margin) នៃទិន្នន័យនិងព័ងដែលមានទំហំធំបំផុត ។

ការកំណត់តម្រឹម (margin) ដែលធំបំផុតនៅទីនេះ គឺសំដៅដល់ការយកប៉ារ៉ាម៉ែត្រណាដែលធ្វើឱ្យចន្លោះរវាងទិន្នន័យទាំងពីរក្រុមនិងបន្ទាត់ (ឬប្លង់) ព័ងមានគម្លាតឆ្ងាយពីគ្នាបំផុត (រូបទី៣) ។



រូបទី៣ ការកំណត់តម្រឹម (margin) អតិបរមា

យើងនឹងបង្ហាញបញ្ហាខាងលើជាទម្រង់គណិតវិទ្យា។

សន្មតបន្ទាត់ឬប្លង់ព្រំដែនមានទម្រង់ដូចម្តងដែលខាងលើ  $\mathbf{w}^T \mathbf{x} + b = 0$  ។ ចម្ងាយពីចំណុច  $\mathbf{x}_0 \in \mathbb{R}^d$  ទៅព្រំដែនអាចកំណត់បានដូចខាងក្រោម។

$$\frac{|\mathbf{w}^T \mathbf{x}_0 + b|}{\|\mathbf{w}\|}$$

ហេតុនេះ ការកំណត់ប៉ារ៉ាម៉ែត្រ  $(\mathbf{w}, b)$  ដែលធ្វើឱ្យតម្រឹមអតិបរមាអាចកំណត់បានជាចំណោទបរមាដូចខាងក្រោម។

$$\max_{\mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}} \min_{i=1,2,\dots,N} \frac{|\mathbf{w}^T \mathbf{x}_i + b|}{\|\mathbf{w}\|}$$

ក្រោមលក្ខខណ្ឌ

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) > 0 \quad (i = 1, 2, \dots, N)$$

ចំណោទបរមាខាងលើអាចសរសេរជាទម្រង់សមមូលបែបងាយដូចខាងក្រោមបានដោយសារតែអនុគមន៍គោលដៅដែលត្រូវធ្វើបរមាកម្មមិនប្រែប្រួលតម្លៃឡើយពេល  $\mathbf{w}, b$  ត្រូវបានគុណនឹងចំនួនថេរក៏ដោយ។

$$\min_{\mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}} \frac{\|\mathbf{w}\|^2}{2}$$

ក្រោមលក្ខខណ្ឌ

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) > 0 \quad (i = 1, 2, \dots, N)$$

អនុគមន៍គោលដៅខាងលើមានទម្រង់ជាអនុគមន៍ប៉ោងដ៏ក្រេឡី២ ហើយលក្ខខណ្ឌរបស់វាដែលត្រូវផ្ទៀងផ្ទាត់មានទម្រង់



ជាលីនេអ៊ែរ ។ ចំណោទបរមាបែបនេះហៅថាចំណោទប្រក្រាមលំដាប់២ (QP:Quadratic Programming) ។ ក្នុងការដោះស្រាយចំណោទបែបនេះមានalgorithmដែលមានប្រសិទ្ធិភាពខ្ពស់ជាច្រើនត្រូវបានស្រាវជ្រាវ។ នៅទីនេះយើងនឹងមិនធ្វើការបកស្រាយលំអិតឡើយ។ សន្មតថាចម្លើយនៃចំណោទបរមាខាងលើគឺ  $\hat{w} \in \mathbb{R}^d, \hat{b} \in \mathbb{R}$  នោះការធ្វើចំណាត់ថ្នាក់ក្រុមទិន្នន័យអាចធ្វើបានដោយគណនាតាមទម្រង់ខាងក្រោម។

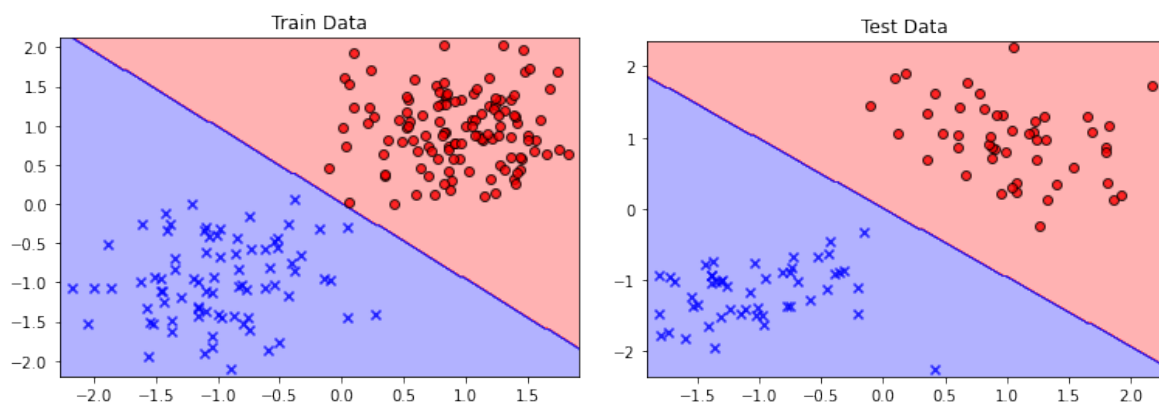
$$\hat{m}(x) = \text{sign}(\hat{w}^T x + \hat{b})$$

ជាមួយPython អ្នកអាចប្រើ sklearn.svm packageបានប្រើSupport Vector Machine Model ។

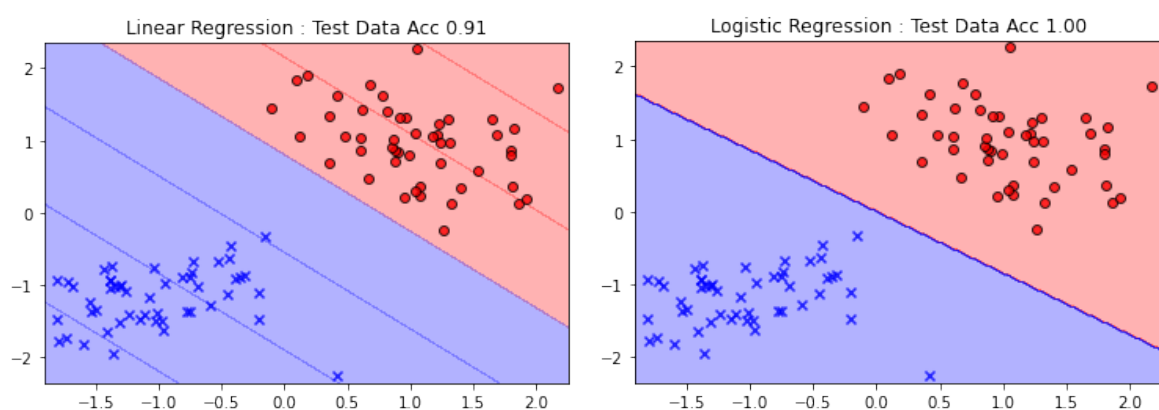
---

```
from sklearn.svm import SVC, LinearSVC
sv_model = SVC(kernel="linear", C=1.0, random_state=1)
sv_model.fit(Xtrain,ytrain)
```

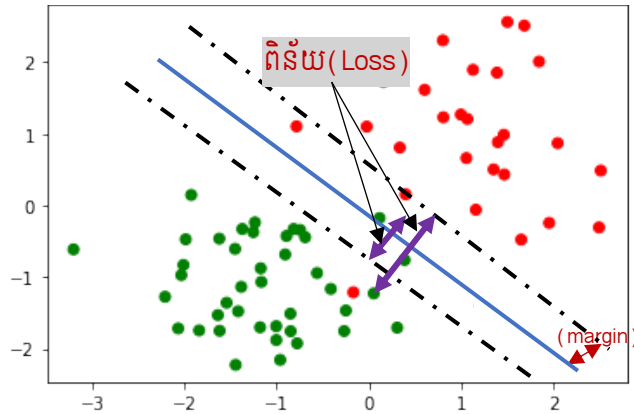
---



រូបទី៤ ការធ្វើចំណាត់ថ្នាក់ក្រុម (កំណត់ព្រំដែនក្រុម) ដោយSupport Vector Machineលើ Linear Seperable Data



រូបទី៥ ការធ្វើចំណាត់ថ្នាក់ក្រុម (កំណត់ព្រំដែនក្រុម) ដោយ Regression Model លើ Linear Seperable Data



រូបទី៦ បន្ទាត់ព្រំដែន និង កម្រិតពិន័យ(loss) ក្នុង Soft Support Vector Machine

## 1.2. ករណីទិន្នន័យដែលអាចធ្វើបំណែងចែកលីនេអ៊ែរបាន (Linear Seperable)

ក្នុងករណីដែលទិន្នន័យមិនអាចធ្វើបំណែងចែកលីនេអ៊ែរបាន ចំណោទបរមាខាងលើគ្មាន តម្លៃប៉ារ៉ាម៉ែត្រណាដែលផ្ទៀងផ្ទាត់លក្ខខណ្ឌឡើយ ។ ហេតុនេះដើម្បីដោះស្រាយបញ្ហាបំណែងចែក បែបនេះករណីកំហុស (មិនបំពេញលក្ខខណ្ឌ  $y_i(\mathbf{w}^T \mathbf{x}_i + b) > 0$  ( $i = 1, 2, \dots, N$ )) ក្នុងបំណែងចែក ខ្លះត្រូវបានលើកលែង ។ ដំណោះស្រាយបែបនេះហៅថា Soft Support Vector Machine ។

នៅទីនេះដោយប្រើម៉ូដែលបំណែងចែក  $m(\mathbf{x}) = \text{sign}(\mathbf{w}^T \mathbf{x} + b)$  ករណីទិន្នន័យ  $(\mathbf{x}_i, y_i)$  តម្លៃនៃការពិន័យ (Loss) លើកំហុសនៃចំណាត់ថ្នាក់ក្រុមត្រូវបានកំណត់ដូចខាងក្រោម ។

(១) ចំពោះ  $y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1$  យើងសន្មតថាម៉ូដែលអាចបែងចែកបានល្អ ដោយកំណត់តម្លៃពិន័យ ០

(២) ចំពោះ  $y_i(\mathbf{w}^T \mathbf{x}_i + b) < 1$  យើងសន្មតថាម៉ូដែលអាចបែងចែកមិនបានល្អដោយកំណត់តម្លៃនៃ

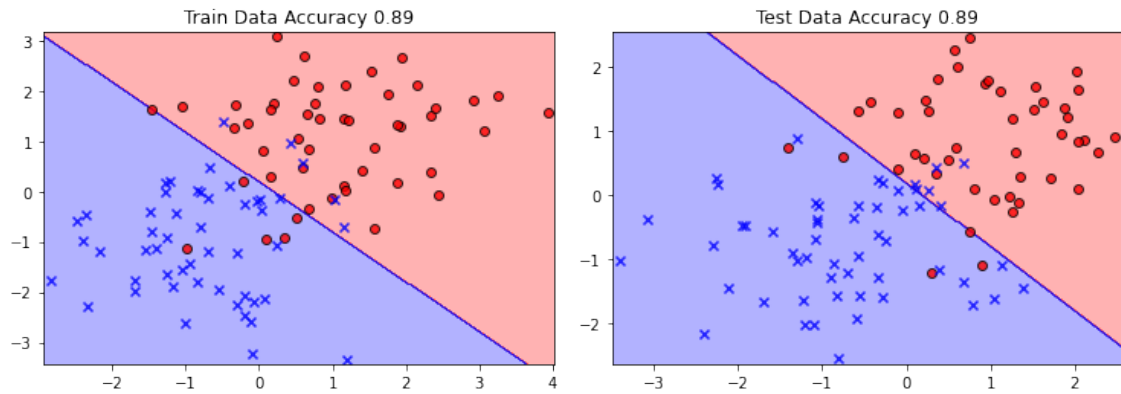
$$\text{ពិន័យ } 1 - y_i(\mathbf{w}^T \mathbf{x}_i + b) > 0$$

ក្នុងករណីនេះដើម្បីកំណត់ប៉ារ៉ាម៉ែត្រដែលប្រសើរសម្រាប់ធ្វើចំណាត់ថ្នាក់ក្រុមជាមួយ Soft Support Vector Machine យើងនឹងកំណត់តម្លៃប៉ារ៉ាម៉ែត្រណាដែលធ្វើឱ្យតម្រឹមធំបំផុត តែកម្រិតពិន័យ (Loss) តូចបំផុត ។ យើងអាចបង្ហាញបញ្ហានេះជាទម្រង់ចំណោទបរមាបានដូចខាងក្រោម ។

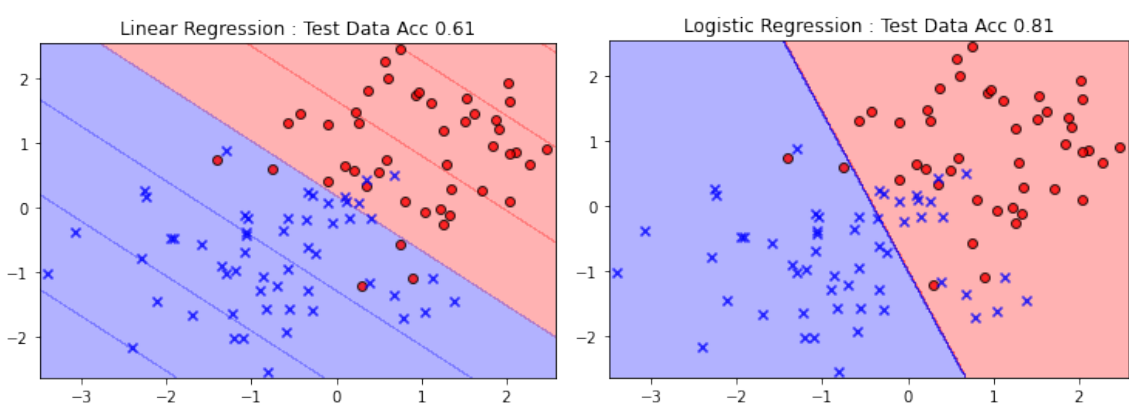
$$\min_{\mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}} \frac{C}{N} \sum_{i=1}^N \max\{1 - y_i(\mathbf{w}^T \mathbf{x}_i + b), 0\} + \frac{1}{2} \|\mathbf{w}\|^2$$

$C$  ជាតម្លៃសម្រាប់កំណត់កម្រិតនៃការផ្ដោតលើបរមាកម្មរវាងតម្រឹម (margin) និងពិន័យ (loss) ។

នៅទីនេះដូចដែលអ្នកអាចចាប់អារម្មណ៍បាន ការកំណត់តម្លៃ  $C$  ជារឿងពិបាក ។ វិធីដែលច្រើនអនុវត្ត គឺការធ្វើ Cross-validation ។ យើងនឹងមិនលំអិតលើវិធីសាស្ត្រនេះឡើយនៅទីនេះ ។



រូបទី៧ ការធ្វើចំណាត់ថ្នាក់ក្រុម( កំណត់ព្រំដែនក្រុម )ដោយSupport Vector Machineលើ Linear Non-Seperable Data



រូបទី៨ ការធ្វើចំណាត់ថ្នាក់ក្រុម( កំណត់ព្រំដែនក្រុម )ដោយRegression Model លើ Linear Non-Seperable Data

តាមលទ្ធផលក្នុងរូបខាងលើ យើងអាចផ្ទៀងផ្ទាត់ពីការធ្វើបំណែងចែកក្រុមបានល្អប្រសើរក្នុងករណីប្រើSoft Support Vector Machine ប្រើប្រៀបធៀបទៅនឹងម៉ូដែលតម្រេតម្រង់ដែលបានសិក្សាកន្លងមក ។ប៉ុន្តែទោះជាយ៉ាងណា ករណីទិន្នន័យដែលមិនអាចធ្វើបំណែងចែកលីនេអ៊ែរបាន កម្រិតត្រឹមត្រូវនៃការធ្វើចំណាត់ថ្នាក់ក្រុមនៅមានកម្រិតទោះបីជាប្រើSoft Support Vector Machineក្តី ។

## 2. Kernel Support Vector Machine

ក្នុងករណីដែលទិន្នន័យមិនអាចបែងចែកលីនេអ៊ែរបាន ការប្រើKernelជាជំនួយត្រូវបានអនុវត្តជាទូទៅ ។ ជាមួយការប្រើប្រាស់Kernel សមត្ថភាពនៃការបង្ហាញលក្ខណៈពិសេសរបស់ទិន្នន័យនឹងត្រូវបានបង្កើន ។ កាលពីសិក្សាពីម៉ូដែលតម្រេតម្រង់ យើងបានលើកឡើងអំពីការប្រើអនុគមន៍គោលដែលជាអនុគមន៍មិនលីនេអ៊ែរដើម្បីបង្ហាញលក្ខណៈពិសេសរបស់ទិន្នន័យមួយចំនួន ។

ស្រដៀងគ្នានេះជាមួយគោលគំនិតក្នុងការប្រើKernel ឧបមាថាយើងមានអនុគមន៍គោល  $\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \dots, \varphi_D(\mathbf{x})$  នោះអនុគមន៍Kernel ត្រូវបានកំណត់ដូចខាងក្រោម ដែលនៅទីនេះ  $\Phi(\mathbf{x}) = (\varphi_1(\mathbf{x}) \ \dots \ \varphi_D(\mathbf{x}))^T$  ។

$$k(\mathbf{x}, \mathbf{x}') = \sum_{d=1}^D \varphi_d(\mathbf{x}) \varphi_d(\mathbf{x}') = \Phi(\mathbf{x})^T \Phi(\mathbf{x}')$$

ក្នុងករណីនេះ ទម្រង់នៃ ម៉ូដែលចំណាត់ថ្នាក់ក្រុមនិងបន្ទាត់ឬប្លង់ព្រំដែនអាចបង្ហាញ ដូចទម្រង់ខាងក្រោម ដោយសន្មតយក  $\boldsymbol{\beta} = (\beta_1 \ \dots \ \beta_n)^T$  ជំនួស  $\mathbf{w}$  ។

$$f(\mathbf{x}) = \mathbf{w}^T \Phi(\mathbf{x}) + b$$

$$f(\mathbf{x}) = \sum_{i=1}^N \beta_i k(\mathbf{x}, \mathbf{x}_i) + b$$

ដូចគ្នានឹងករណីទូទៅនៃSupport Vector Machine ដែរ ក្នុងករណី Kernel Support Vector Machine បញ្ហាខាងលើអាចបង្ហាញជាទម្រង់ចំណោទបរមាដូចខាងក្រោម ។

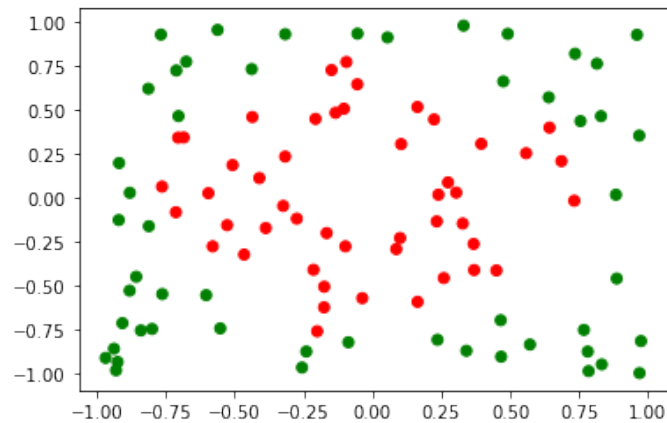
$$\min_{\boldsymbol{\beta}, b} \frac{C}{N} \sum_{i=1}^N \max\{1 - y_i f(\mathbf{x}_i), 0\} + \frac{1}{2} \boldsymbol{\beta}^T K \boldsymbol{\beta}$$

ក្រោមលក្ខខណ្ឌ

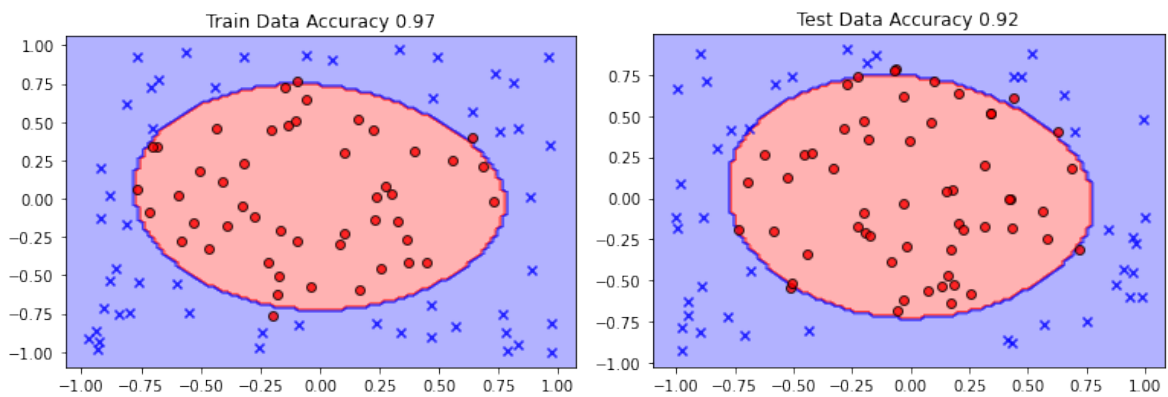
$$f(\mathbf{x}_i) = \sum_{j=1}^N \beta_j K_{ij} + b \quad (i = 1, 2, \dots, N)$$

ដែល  $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$  ។

ជាមួយPython អ្នកអាចជ្រើសរើសប្រភេទ Kernel ដែលនិយមប្រើដូចជា: លីនេអ៊ែរ linear , ពហុធា poly , Gaussian Kernel: rbf , Sigmoid Kernel: sigmoid ។



រូបទី៨ ទិន្នន័យដែលមិនអាចបែងចែកលីនេអ៊ែរបាន



រូបទី៩ ការធ្វើចំណាត់ថ្នាក់ក្រុម(កំណត់ព្រំដែនក្រុម)ដោយKernel Support Vector Machine (kernel=rbf)  
លើ Linear Non-Seperable Data

---

```
from sklearn.svm import SVC

sv_model = SVC(kernel="rbf", C=1.0, random_state=1)
sv_model.fit(Xtrain,ytrain)
```

---

```
print("Train score:",sv_model.score(Xtrain,ytrain))
print("Test score:",sv_model.score(Xtest,ytest))
```

---

```
Train score: 0.97
Test score: 0.92
```

---

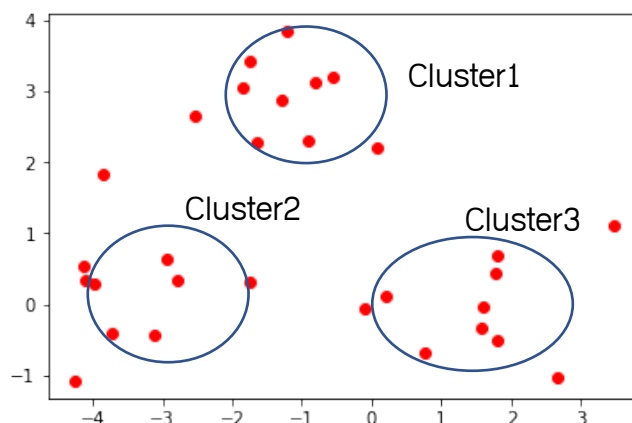
## Clustering

ក្នុងបញ្ហានៃការធ្វើចំណាត់ថ្នាក់ក្រុមទិន្នន័យដែលយើងបានណែនាំក្នុងអត្ថបទមុនៗ ទិន្នន័យនីមួយៗជាគូ  $(x, y)$  ពេលគឺយើងប្រើទិន្នន័យដែលមានចម្លើយជាមុនដើម្បីបង្រៀនដល់ម៉ូដែលរបស់យើងដែលហៅថា Supervised Learning ។ ផ្ទុយពីនេះ ប្រភេទបញ្ហាក្នុង Unsupervised Learning យើងមានតែទិន្នន័យ  $x$  តែប៉ុណ្ណោះ។ ក្នុងករណីនេះការប្រមូលផ្តុំទិន្នន័យដែលមានលក្ខណៈដូច ឬស្រដៀងគ្នាជាក្រុមឬជាចង្កោមដោយផ្អែកលើលក្ខណៈបង្ហាញដោយផ្ទាល់ឬប្រយោលរបស់ទិន្នន័យត្រូវបានហៅថាជា Clustering ។ ក្នុងអត្ថបទនេះយើងនឹងណែនាំអំពីវិធីសាស្ត្រក្នុងការដោះស្រាយបញ្ហាបែបនេះ។

### 1. ចង្កោម(Cluster)ទិន្នន័យនិងចម្ងាយ

ការធ្វើចំណាត់ក្រុមទិន្នន័យដែលគ្មានសញ្ញាសម្គាល់ប្រភេទជាក្រុមដូចក្នុងរូបទី១គឺជាគោលដៅចម្បងនៃការសិក្សាក្នុង Unsupervised Learning ។ ក្រុមដែលប្រមូលបានក្នុងទិន្នន័យនោះហៅថាចង្កោមទិន្នន័យ (Cluster) ហើយដំណើរការនៃការប្រមូលជាក្រុមបែបនេះហៅថា Clustering ។

ដើម្បីធ្វើចំណាត់ក្រុមទិន្នន័យបែបនេះ គោលគំនិតសំខាន់គឺការស្វែងរកលក្ខណៈរួមឬប្រហាក់ប្រហែលរវាងទិន្នន័យ។ ពេលគឺ ទិន្នន័យដែលមានលក្ខណៈស្រដៀងគ្នាលើរង្វាស់ណាមួយអាចប្រមូលផ្តុំជាចង្កោមបាន។ ហេតុនេះការកំណត់និយមន័យជាក់លាក់នៃលក្ខណៈរួមឬប្រហាក់ប្រហែលនេះជាដំណើរការសំខាន់ក្នុងការចាប់ផ្តើម។ នៅទីនេះយើងកំណត់យកចម្ងាយជាង្វាស់សម្រាប់បង្ហាញភាពស្រដៀងគ្នានៃទិន្នន័យ។ ដូច្នេះយើងនឹងធ្វើការពិនិត្យលើនិយមន័យនៃចម្ងាយដូចខាងក្រោម។



រូបទី១ ទិន្នន័យនិងចង្កោមដែលអាចកំណត់បាន

### 1.1. ចម្ងាយ Euclid

សន្មតថា  $x, y \in \mathbb{R}^d$  ជាពីរចំណុចក្នុងលំហទិន្នន័យ។

បើ  $x = (x_1 \ \cdots \ x_d)^\top, y = (y_1 \ \cdots \ y_d)^\top$  នោះ ចម្ងាយ Euclid រវាងចំណុចទាំងពីរនេះអាចកំណត់បានដូចខាងក្រោម។

$$D_{Euclid}(x, y) = \|x - y\|_2 = \left( \sum_{i=1}^d (x_i - y_i)^2 \right)^{\frac{1}{2}}$$

### 1.2. ចម្ងាយ Manhattan

សន្មតថា  $x, y \in \mathbb{R}^d$  ជាពីរចំណុចក្នុងលំហទិន្នន័យ។

បើ  $x = (x_1 \ \cdots \ x_d)^\top, y = (y_1 \ \cdots \ y_d)^\top$  នោះ ចម្ងាយ Manhattan រវាងចំណុចទាំងពីរនេះអាចកំណត់បានដូចខាងក្រោម។

$$D_{Manhattan}(x, y) = \|x - y\|_1 = \sum_{i=1}^d |x_i - y_i|$$

### 1.3. កាតស្រដៀងគ្នាកូស៊ីនុស

សន្មតថា  $x, y \in \mathbb{R}^d$  ជាពីរចំណុចក្នុងលំហទិន្នន័យ។ ដោយប្រើមុំផ្គុំដោយវ៉ិចទ័រចំណុចទាំងពីរយើងអាចកំណត់ថាវ៉ិចទ័រចំណុចទាំងពីរមានទិសដៅដូចគ្នានៅពេលរង្វាស់មុំនោះកាន់តែតូច។ ហេតុនេះការប្រើតម្លៃកូស៊ីនុសនៃមុំផ្គុំដោយវ៉ិចទ័រចំណុចទាំងពីរអាចប្រើជារង្វាស់ប្រៀបធៀបលក្ខណៈស្រដៀងគ្នានៃទិន្នន័យបាន។ បើ  $(x, y)$  ជាផលគុណស្កាលែនៃវ៉ិចទ័រទាំងពីរ នោះ កាតស្រដៀងគ្នាកូស៊ីនុសរវាងចំណុចទាំងពីរនេះអាចកំណត់បានដូចខាងក្រោម។

$$\cos(x, y) = \frac{(x, y)}{\|x\|_2 \cdot \|y\|_2}$$

### 1.4. មេគុណ Jaccard

ក្នុងករណីដែលទិន្នន័យបង្ហាញជាទម្រង់សំណុំ ពេលគឺ  $x = \{x_1, \dots, x_d\}, y = \{y_1, \dots, y_d\}$  នោះកម្រិតស្រដៀងគ្នានៃទិន្នន័យទាំងពីរអាចកំណត់បានដោយមេគុណ Jaccard ដូចខាងក្រោមដែល  $|a|$  ជាកាឌីណាល់ (ចំនួនធាតុ) នៃសំណុំ  $a$  ។

$$\text{Jaccard}(x, y) = \frac{|x \cap y|}{|x \cup y|}$$

### 1.5. KL divergence

ក្នុងករណីដែលទិន្នន័យបង្ហាញជាទម្រង់អនុគមន៍បំណែងចែកប្រូបាប  $p(x), p(y)$  នោះកម្រិតស្រដៀងគ្នានៃទិន្នន័យទាំងពីរអាចកំណត់បានដូចខាងក្រោមដែល ។

$$KL(p||q) = \int p(x) \log \frac{p(x)}{q(x)} dx$$

កម្រិតស្រដៀង KL មិនមានលក្ខណៈឆ្លុះឡើយ ពោលគឺ  $KL(p||q) \neq KL(q||p)$  ។ ហេតុនេះក្នុងករណីខ្លះ Jensen-Shannon Divergence ត្រូវបានប្រើជំនួស ។

$$D_{js} = \frac{1}{2} (KL(p||q) + KL(q||p))$$

## 2. វិធីសាស្ត្រ K-means

សន្មតថាយើងមានចំណុចទិន្នន័យ  $x_1, \dots, x_N \in \mathbb{R}^d$  ។ យើងចង់ធ្វើចំណាត់ថ្នាក់ក្រុមដោយស្វ័យប្រវត្តិលើចំណុចទិន្នន័យទាំងនេះជា  $K$  ក្រុម  $C_1, \dots, C_K$  ។ ដូចដែលបង្ហាញខាងលើទិន្នន័យនីមួយៗមិនមានភ្ជាប់ជាមួយនូវកំណត់សម្គាល់ (label) អំពីក្រុមដែលខ្លួនស្ថិតនៅឡើយ ។ ហេតុនេះ ការធ្វើចំណាត់ក្រុមត្រូវពិនិត្យលើកម្រិតស្រដៀងគ្នានៃទិន្នន័យដោយផ្អែកលើចម្ងាយរវាងគ្នា ។

ដំបូង យើងកំណត់ហៅចំណុចតំណាងនៃក្រុមដោយ  $\mu_1, \dots, \mu_K \in \mathbb{R}^d$  ។ ចំណុចតំណាងទាំងនេះត្រូវបានហៅថា Centroids ។ ចំណុចទិន្នន័យដែលនៅជិតចំណុចតំណាង  $\mu_k$  នឹងត្រូវចាត់ចូលជាសមាជិកនៃក្រុម  $C_k$  ។

បើចម្ងាយរវាងពីរចំណុចទិន្នន័យកំណត់ដោយ  $d(x_1, x_2)$  នោះកម្រិតគម្លាតសរុបរវាងក្រុមនីមួយៗត្រូវបានកំណត់ដូចខាងក្រោម ។ ការធ្វើចំណាត់ថ្នាក់ក្រុមដែលល្អត្រូវមានតម្លៃនៃកម្រិតគម្លាតសរុបរវាងក្រុមតូចបំផុត ។ ការធ្វើចំណាត់ថ្នាក់ក្រុមបែបនេះហៅថា វិធីសាស្ត្រ K-means ។

$$\sum_{k=1}^K \sum_{x \in C_k} d(x, \mu_k)^2$$

ក្នុងការបកស្រាយខាងក្រោម យើងកំណត់ប្រើចម្ងាយ Euclid ។ តាង  $|C_k|$  ជាចំនួនចំណុចទិន្នន័យក្នុងក្រុម  $C_k$  នោះវ៉ិចទ័រមធ្យមនៃចំណុចទិន្នន័យក្នុងក្រុមនេះកំណត់ដោយ  $\bar{x}_k$  ។

$$\bar{x}_k = \frac{1}{|C_k|} \sum_{x \in C_k} x$$

ចំពោះក្រុម  $C_1, \dots, C_K$  យើងបានទំនាក់ទំនងខាងក្រោម ។ ហេតុនេះ បើយើងកំណត់យកចំណុចតំណាងនៃក្រុមនីមួយៗ  $\mu_k$  ដោយវ៉ិចទ័រមធ្យមនៃចំណុចទិន្នន័យក្នុងក្រុម នោះយើងនឹងបានតម្លៃនៃកម្រិតគម្លាតសរុបរវាងក្រុមតូចបំផុត ។



$$\sum_{x \in C_k} \|x - \mu_k\|_2^2 \geq \sum_{x \in C_k} \|x - \bar{x}_k\|_2^2 \quad (k = 1, \dots, K)$$

ដោយផ្អែកលើទំនាក់ទំនងនេះ ដំណើរការនៃវិធីសាស្ត្រ K-means ចំពោះចម្ងាយ Euclid អាចសរុបដូចខាងក្រោម ។

---

**Input:** ចំណុចទិន្នន័យ  $x_1, \dots, x_N \in \mathbb{R}^d$ , ចំនួនក្រុម  $K$

**Initialization:** កំណត់តម្លៃចាប់ផ្តើមនៃចំណុចតំណាង  $\mu_1, \dots, \mu_K \in \mathbb{R}^d$

**Step-1 :** អនុវត្តជំហាន (1), (2), (3) ខាងក្រោមដដែលៗ

- (1) ផ្លាស់ប្តូរសមាជិកក្រុម  $C_1, \dots, C_K$   
(សន្មតថាទិន្នន័យនីមួយៗត្រូវស្ថិតនៅក្នុងក្រុមណាមួយក្នុងចំណោមនេះ)  
$$C_k = \{x_n \mid \|x_n - \mu_k\|_2 \leq \|x_n - \mu_{k'}\|_2, k' \neq k\}$$
- (2) ផ្លាស់ប្តូរតម្លៃនៃចំណុចតំណាង  $\mu_1, \dots, \mu_K$   
$$\mu_k = \frac{1}{|C_k|} \sum_{x \in C_k} x \quad (k = 1, \dots, K)$$
- (3) បន្តទៅ Step-2 នៅពេលដែលតម្លៃនៃកម្រិតគម្លាតសរុបរវាងក្រុមរួម  
(ស្មើឬក្បែរសូន្យ)

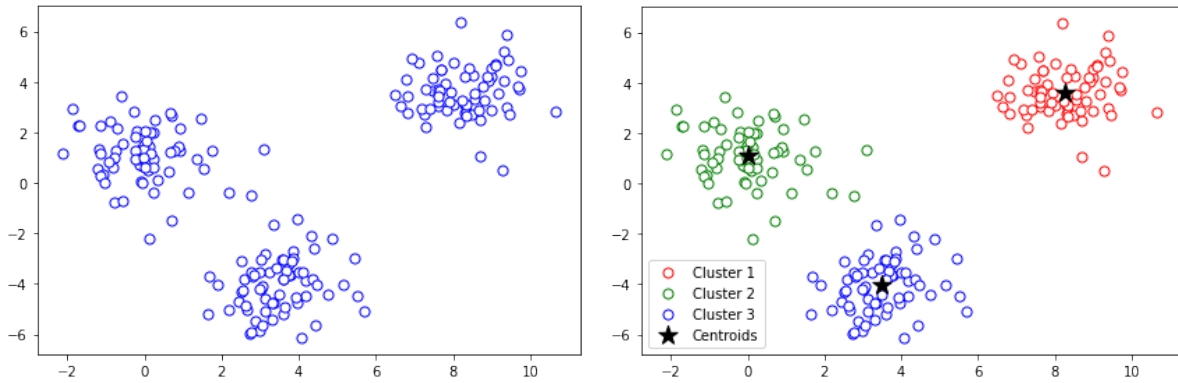
**Step-2 :** យកការធ្វើចំណាត់ថ្នាក់ក្រុម  $C_1, \dots, C_K$  ជាចម្លើយ

---

ជាមួយ Python អ្នកអាចប្រើ sklearn.cluster បាន ។

```
from sklearn.cluster import KMeans
KM = KMeans(n_clusters=3, init='random', n_init=10, max_iter=500, tol=1e-6)
y_KM = KM.fit_predict(X)
```

---



រូបទី២ ចំណុចទិន្នន័យមុនធ្វើចំណាត់ថ្នាក់ក្រុម និងក្រោយធ្វើចំណាត់ក្រុមដោយK-means

ក្នុងការអនុវត្តវិធីសាស្ត្រK-means ការកំណត់តម្លៃដំបូងនៃចំណុចតំណាងមានឥទ្ធិពលខ្លាំងលើលទ្ធផលនៃការធ្វើចំណាត់ក្រុម ។ ជាទូទៅការកំណត់តម្លៃដំបូងនេះធ្វើឡើងដោយតម្លៃចៃដន្យ ។

ក្នុងករណីsklearn.cluster.KMeans យើងអាចកំណត់ដោយ `init='random'` ។

ប៉ុន្តែករណីខ្លះការកំណត់ដោយចៃដន្យនេះអាចនឹងបានលទ្ធផលចំណាត់ក្រុមដែលមិនល្អ ។

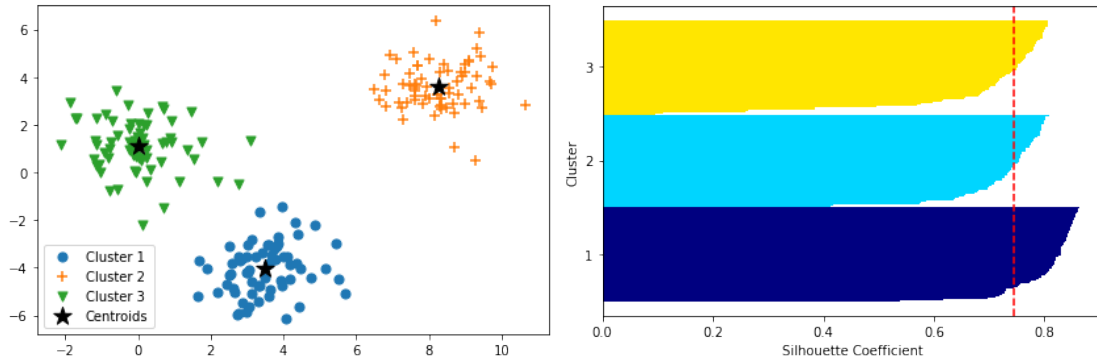
ដើម្បីដោះស្រាយបញ្ហានេះ ការប្រើ K-means++ អាចសម្រួលបាន ។ ក្នុងK-means++ វិធីសាស្ត្រK-meansត្រូវបានអនុវត្តដោយចៃដន្យជាច្រើនលើកទៅលើទិន្នន័យដែលមាន រួចតម្លៃមធ្យមនៃកម្រិតគម្លាតសរុបនឹងត្រូវធ្វើអប្បបរមាកម្ម ។

ករណីsklearn.cluster.KMeans យើងអាចកំណត់ដោយ `init='k-means++'` ។

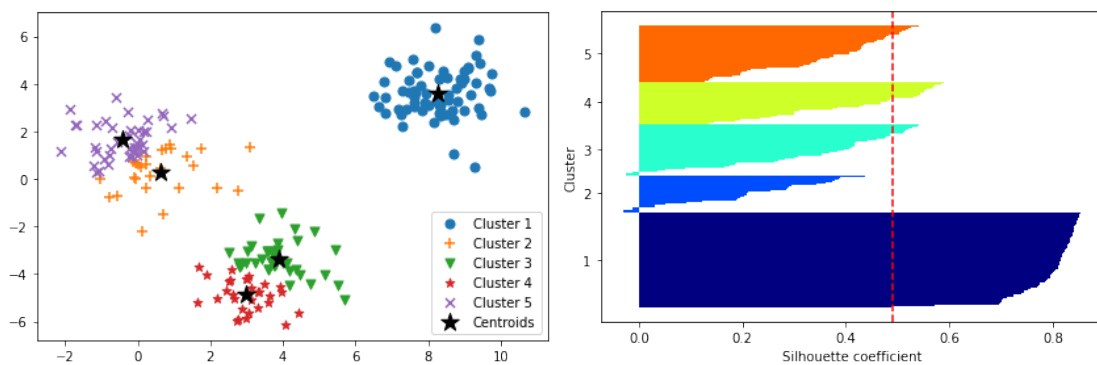
### 3. ការវាយតម្លៃClusteringដោយប្រើមេគុណSilhouette

ក្រៅពីការសិក្សាលើតម្លៃនៃកម្រិតគម្លាតសរុបរវាងក្រុមនីមួយៗ ការវិភាគលើកម្រិតកំហាប់នៃទិន្នន័យក្នុងក្រុម (ភាពជិតស្និទ្ធក្នុងក្រុម) ដូចជា Silhouetter Analysis ក៏ត្រូវបានប្រើប្រាស់សម្រាប់វាយតម្លៃលើ Clustering ផងដែរ ។ ក្នុង Silhouetter Analysis កម្រិតកំហាប់នៃការប្រមូលផ្តុំរបស់ទិន្នន័យក្នុងក្រុមនីមួយៗត្រូវបានគណនាដោយមេគុណsilhouetterនិងបង្ហាញជាក្រាប ។ មេគុណsilhouetter នៃទិន្នន័យ  $x_i : s^{(i)}$  អាចគណនាបានតាមរាងខាងក្រោម ។

- (1) កំណត់កម្រិតកំហាប់ប្រមូលផ្តុំនៃក្រុម  $a^{(i)}$  ដោយតម្លៃមធ្យមនៃចម្ងាយរវាងចំណុចទិន្នន័យ  $x_i$  និងចំណុចទិន្នន័យដទៃទៀតក្នុងក្រុមជាមួយគ្នា ។
- (2) កំណត់កម្រិតគម្លាតរវាងក្រុមជិតបំផុត  $b^{(i)}$  ដោយតម្លៃមធ្យមនៃចម្ងាយរវាងចំណុចទិន្នន័យ  $x_i$  និងចំណុចទិន្នន័យទាំងអស់នៅក្នុងក្រុមដែលជិតនឹងក្រុមរបស់ខ្លួនបំផុត ។
- (3) កំណត់តម្លៃមេគុណ silhouetterដោយ  $s^{(i)} = (b^{(i)} - a^{(i)}) / \max\{a^{(i)}, b^{(i)}\}$



រូបទី៣ មេគុណsilhouetterតាមក្រុមនីមួយៗ( ចំណាត់ថ្នាក់ជា៣ក្រុម )



រូបទី៤ មេគុណsilhouetterតាមក្រុមនីមួយៗ( ចំណាត់ថ្នាក់ជា៣ក្រុម )

ក្នុងរូបទី៣និងទី៤ ខាងលើបង្ហាញក្រហមបង្ហាញតម្លៃមធ្យមនៃមេគុណsilhouetter លើក្រុមនីមួយៗក្នុងករណីចំនួនក្រុមត្រូវបានកំណត់ជា៣និង៥ ។ Clustering ដែលល្អនឹងមានតម្លៃនៃ មេគុណsilhouetterខិតទៅជិត១ ។ ក្នុងរូបខាងលើ យើងអាចឃើញថាករណីClustering ដោយ ៣ក្រុមមានមេគុណsilhouetterប្រសើរជាងបើប្រៀបធៀបនឹងករណី៥ក្រុម ។

#### 4. Gaussian Mixture Models Clustering

ក្នុងវិធីសាស្ត្រ K-means យើងមិនបានធ្វើកំណត់លក្ខខណ្ឌសន្មតណាមួយលើរបាយ បំណែងចែកនៃទិន្នន័យឡើយ ។ ពេលនេះយើងសន្មតថាទិន្នន័យទាំងអស់គឺស្ថិតក្នុងរបាយបំណែង ចែកប្រូបាបមួយ ។ ក្នុងករណីនេះយើងកំណត់យកបំណែងចែកនេះម៉ាល់ និងម៉ូដែលបន្សំនៃនរម៉ាល់ ពោលគឺ Gaussian Mixture Models ដើម្បីដោះស្រាយបញ្ហាClustering ។

សន្មតថាក្រុមនីមួយៗត្រូវបានកំណត់ដោយបំណែងចែកប្រូបាប $Q$ ហើយទិន្នន័យក្នុងក្រុម នីមួយៗត្រូវបានកំណត់ដូចខាងក្រោម ។

$$C_k \sim Q, \quad x \sim p_k(x)$$

ពីទិន្នន័យយើងនឹងធ្វើការកំណត់បំណែងចែក  $Q, p_k$  ដែលនឹងនាំឱ្យយើងអាចកំណត់ចំណាត់ថ្នាក់ក្រុមសម្រាប់ទិន្នន័យនីមួយៗបាន។ ក្នុងម៉ូដែលបន្សំ Mixture Models, ចំពោះបំណែងចែកពហុធាតុ និង បំណែងចែកនរម្យាស់  $p_k(x): N_d(\mu_k, \Sigma_k)$  បើយកប្រូបាបដែលទិន្នន័យជាសមាជិកក្រុម  $C_k$  ដោយ  $q_k$  នោះ ម៉ូដែលស្ថិតិនៃទិន្នន័យ  $x$  អាចកំណត់ដោយទម្រង់ខាងក្រោម។

$$\sum_{k=1}^K q_k p_k(x)$$

ដើម្បីកំណត់ប៉ារ៉ាម៉ែត្រនៃម៉ូដែលនេះយើងអាចសិក្សាលើអនុគមន៍លោការីតនៃកម្រិតសាកសមនៃទិន្នន័យ (log-likelihood function) បាន ពោលគឺ

$$\sum_{n=1}^N \log \left( \sum_{k=1}^K q_k p_k(x_n) \right)$$

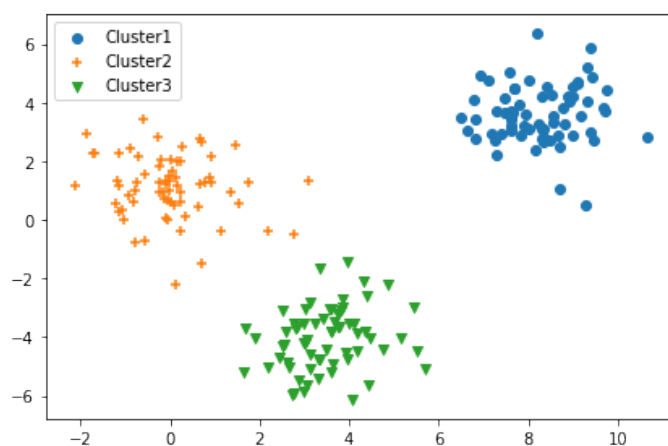
ក្នុងការដោះស្រាយបញ្ហាអតិបរមាកម្មនៃអនុគមន៍ខាងលើនេះ EM algorithm ត្រូវបានប្រើ។ នៅទីនេះយើងនឹងមិនលំអិតអំពី EM algorithm ឡើយ ប៉ុន្តែយើងណែនាំអំពីការប្រើប្រាស់វាក្នុងការធ្វើ Clustering ។

ជាមួយ Python អ្នកអាចប្រើ sklearn.mixture.GaussianMixture បាន។

---

```
from sklearn.mixture import GaussianMixture
model = GaussianMixture(3).fit(X)
classes = model.predict(X)
```

---



រូបទី៥ លទ្ធផលធ្វើ Clustering ដោយ Gaussian Mixture Models

## ឯកសារពិគ្រោះ

Takafumi Kanamori, Python で学ぶ統計的機械学習, 2018

Sadanori Konishi, 多変量解析入門—線形から非線形へ, 2010

Hiroshi Nakagawa, Machine Learning, 2015

Sebastian Raschka, Vahid Mirjalili, Python Machine Learning, 2018