FB: <https://www.facebook.com/groups/2032749180362722/>

Cupoy: <https://ai100-2.cupoy.com/>

D17：特徵工程簡介

### Sample Code & 作業內容

請參考範例程式碼 Day\_017\_Introduction\_of\_Feature\_Engineering.ipynb: 房價預測

請參閱作業範例Day\_017\_HW.ipynb：鐵達尼生存預測

作業1 : A~E五個程式區塊中，哪一塊是特徵工程?

作業2 : 對照程式區塊 B 與 C 的結果，請問那些欄位屬於「類別型欄位」? (回答欄位英文名稱即可)

作業3 : 續上題，請問哪個欄位是「目標值」?

作業請提交Day\_017\_HW.ipynb

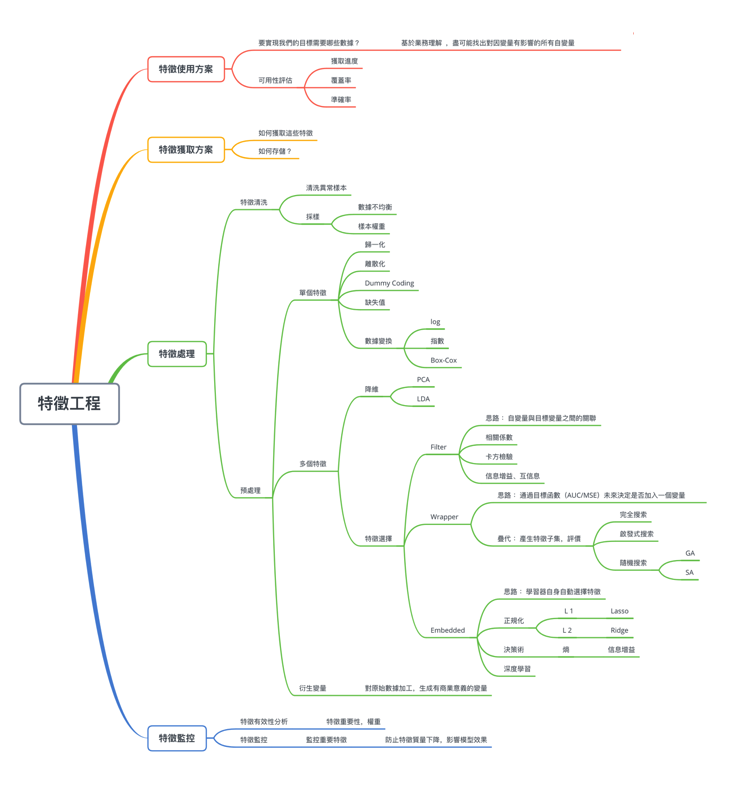
提醒：作業與範例程式碼會使用到的資料，請於檢視範例內data下載  [**1.house**](http://1.house/)\_test  [**2.house**](http://2.house/)\_train  3.titanic\_test  4.titanic\_train

### 參考資料

## 知乎-特徵工程到底是什麼

[**網頁連結**](https://www.zhihu.com/question/29316149)

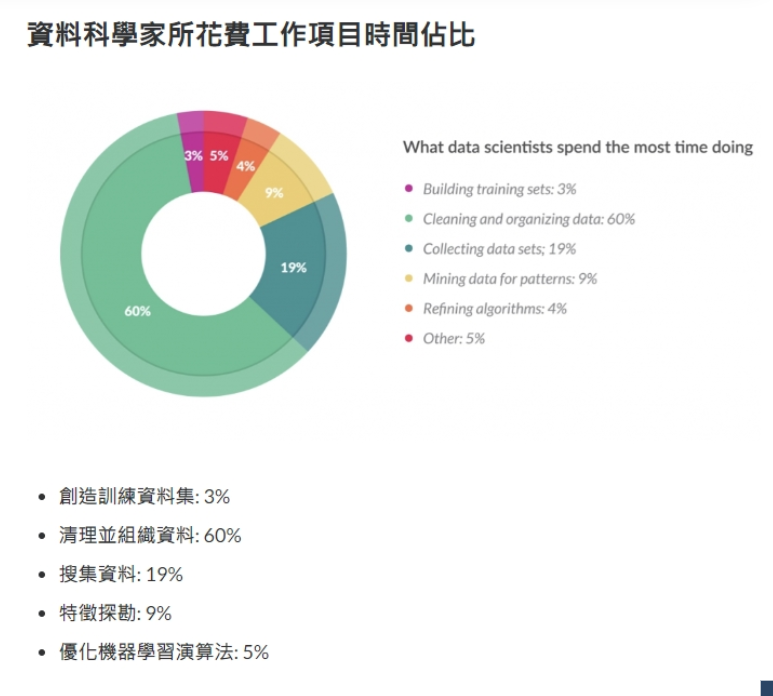
本文重點為右圖, 主要是希望同學大致知道特徵工程大致包含哪些部分, 若對細節有興趣, 還可以從這篇中了解一些概念其中一部分的內容, 會在後面的課程中說明並練習, 詳情請參閱百日馬拉松課綱。



## 痞客幫-iT邦2019鐵人賽 : 為什麼特徵工程很重要

[**網頁連結**](https://ithelp.ithome.com.tw/articles/10200041?sc=iThelpR)

本文主要在描述現實中資料科學工作的時間比重(下圖), 其中大部分的時間在於資料清理, 少部分為特徵探勘, 雖然這兩部份都是特徵工程, 但在學習階段與實務階段, 比重卻有著天壤之別。



## D18：特徵類型

### Sample Code & 作業內容

請參考範例程式碼 Day\_018\_Feature\_Types.ipynb: 房價預測

執行作業範例Day\_018\_HW.ipynb：鐵達尼生存預測

作業1 :

試著執行作業程式，觀察三種類型的欄位分別進行( 平均 mean / 最大值 Max / 相異值 nunique ) 中的九次操作會有那些問題? 並試著解釋那些發生Error的程式區塊的原因?

作業2 :

思考一下，試著舉出今天五種類型以外的一種或多種資料類型，你舉出的新類型是否可以歸在三大類中的某些大類? 所以三大類特徵中，哪一大類處理起來應該最複雜?

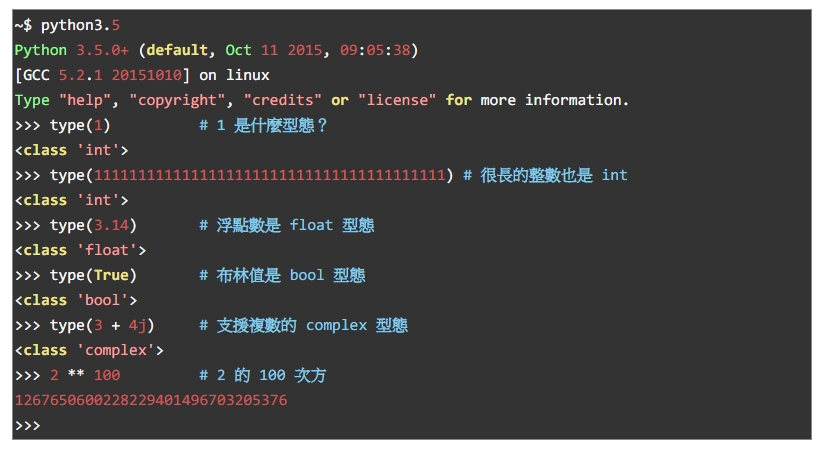
作業請提交Day\_018\_HW.ipynb

### 參考資料

## Python Tutorial 第二堂 - 數值與字串型態

[**網頁連結**](https://openhome.cc/Gossip/CodeData/PythonTutorial/NumericStringPy3.html)

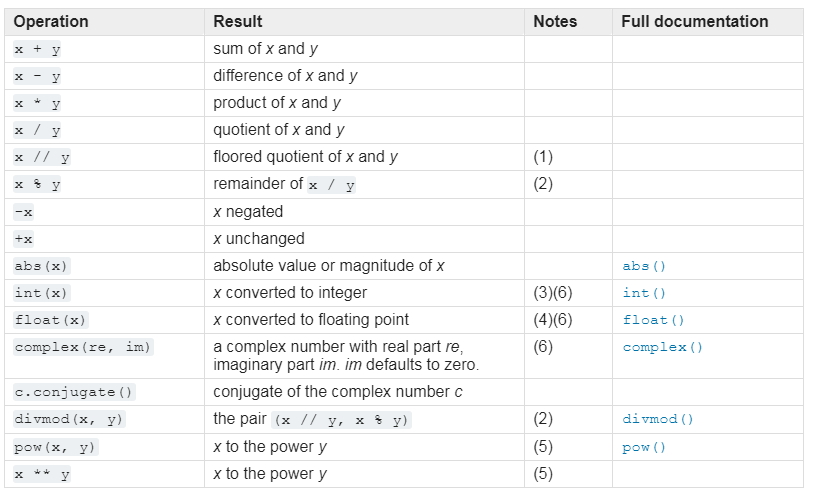
焦點主要在於：搭配一些 python 語法，向同學更清楚的展示型態的差異，如果同學看完今天的課程人不是很了解，建議可以搭配此內容理解，當然如果覺得已經能自己分辨，就可以跳過這份參考資料了。



## Python3.7.2 : Built-in Types Python 官方說明

[**網頁連結**](https://docs.python.org/3/library/stdtypes.html)

這份官方網頁內容資料則是從語言的角度，列舉這些型態，以及對應的操作，因為是官方網頁，會隨著版本更新而略有修改，建議同學有需要的時候(例如版本變更)，再來查詢即可，不用細讀。



## D19：數值型特徵-補缺失值與標準化

### Sample Code & 作業內容

請參考範例程式碼 Day\_019\_Fill\_NaN\_and\_Scalers.ipynb: 房價預測

執行作業範例Day\_019\_HW.ipynb：鐵達尼生存預測

作業1 : 試著在補空值區塊, 替換並執行兩種以上填補的缺值, 看看何者比較好?

作業2 : 使用不同的標準化方式 ( 原值 / 最小最大化 / 標準化 )，搭配羅吉斯迴歸模型，何者效果最好?

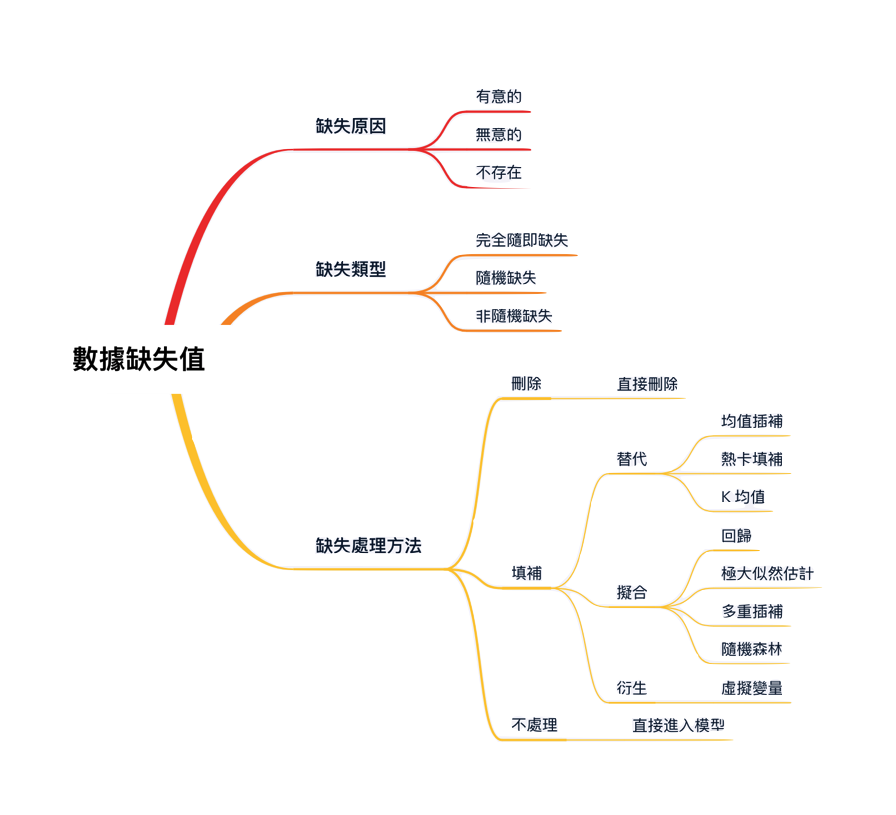
作業請提交Day\_019\_HW.ipynb

### 參考資料

## 掘金 : Python數據分析基礎 : 數據缺失值處理

[**網頁連結**](https://juejin.im/post/5b5c4e6c6fb9a04f90791e0c)

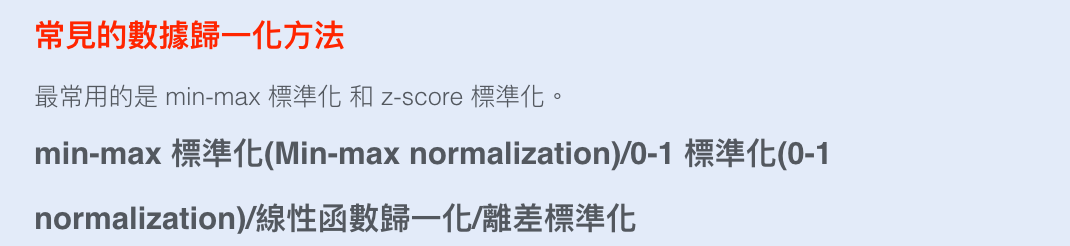
這篇文章更詳細地介紹了各種缺失值的種類，以及處理的各種方式優缺點，如果要徹底搞懂缺失值的話，這是一份不錯的補充資料。但是我們還是要強調：補缺是因資料而異，所以熟悉方法與觀察資料本身都是同樣重要的，因此這在實務上會是一個辛苦的環節。



## 數據標準化 / 歸一化normalization

[**網頁連結**](https://blog.csdn.net/pipisorry/article/details/52247379)

本文重點如下圖，介紹了標準化 / 最大最小化以外的一些特徵縮放方式，雖然文中也提到這兩種就是最常見的方式了，但是其他幾種方式也是不錯的參考，提供同學查閱。



## D20：數值型特徵 - 去除離群值

### Sample Code & 作業內容

請參閱範例Day\_020\_Outliers.ipynb : 房價預測

作業1 : 試著限制 '1樓地板面積(平方英尺)' (1stFlrSF) 欄位的上下限，看看能否再進一步提高分數?

作業2 : 續前題，去除離群值有兩類方式 :  捨棄離群值(刪除離群的資料) 以及調整離群值，請試著用同樣的上下限，改為 '捨棄離群值' 的方法，看看結果會變好還是變差? 並試著解釋原因。

作業請繳交Day\_020\_HW.ipynb

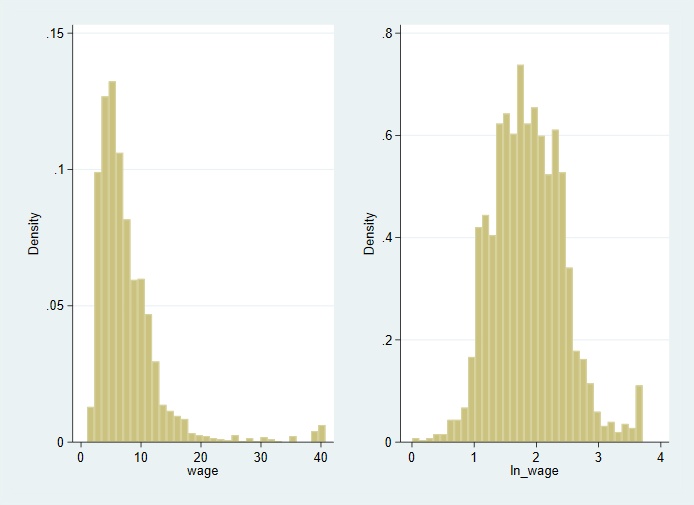
### 參考資料

## 離群值! 離群值? 離群值!

Python 知乎-連玉君stata

[**網頁連結**](https://zhuanlan.zhihu.com/p/33468998)

本文除了談到離群值的定義外，主要在第3部分: 離群值處理方法，不僅僅告訴你有哪幾類方式，並以圖示的方式呈現讓同學能以常識推論分析，其中對數轉換的部分，我們會在明天的內容有更多的講解，同學大致了解方向即可。



## D21：數值型特徵 - 去除偏態

### Sample Code & 作業內容

請參閱作業範例Day\_021\_Reduce\_Skewness.ipynb : 鐵達尼生存預測

作業1 : 試著在鐵達尼的票價 (Fare) 欄位中使用對數去偏 (log1p) ，結果是否更好?

作業2 : 最後的 boxcox 區塊直接執行會造成錯誤，起因為輸入值有負值，請問如何修正後可以使用 boxcox? (Hint : 試圖修正資料)

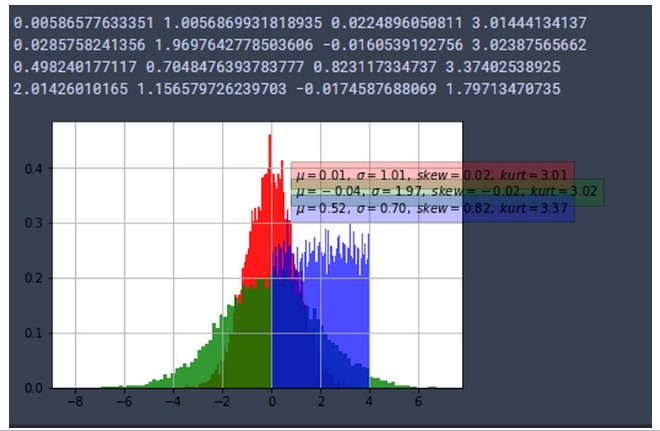
作業請繳交：Day\_021\_HW.ipynb

### 參考資料

## 機器學習數學|偏度與峰度及其python 實現

程式前沿 [**網頁連結**](https://blog.csdn.net/u013555719/article/details/78530879)

本文提到了峰度與偏度的定義與統計上的意義之外，也提供了Python 上的實現，同學可以參考一下這些圖片，而把其對應的程式碼當作工具參考。



## D22：類別型特徵 - 基礎處理

### Sample Code & 作業內容

作業1 : 觀察範例Day\_022\_LabelEncoder\_and\_OneHotEncoder，在房價預測中調整標籤編碼(Label Encoder) / 獨熱編碼 (One Hot Encoder) 方式，對於線性迴歸以及梯度提升樹兩種模型，何者影響比較大?

作業2 : 鐵達尼號例題中，標籤編碼 / 獨熱編碼又分別對預測結果有何影響?

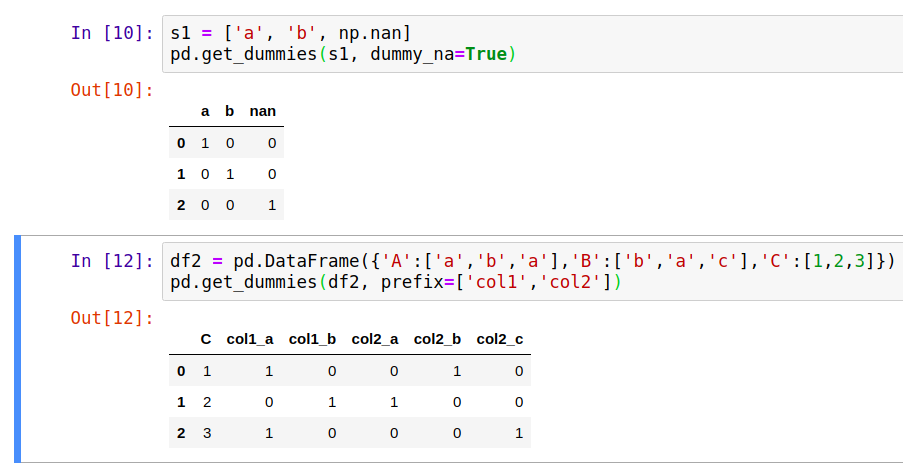
作業請繳交：Day\_022\_HW.ipynb

### 參考資料

## 數據預處理：獨熱編碼（One-Hot Encoding）和 LabelEncoder標籤編碼

神馬文庫 [**網頁連結**](https://www.twblogs.net/a/5baab6e32b7177781a0e6859/zh-cn/)

其實 One Hot Encoding 與 Label Encoder 是類別型資料最常見的編碼方式，因此實現的程式碼也頗為常用，其中 One Hot Encoding 常見的兩種做法 : pandas.get\_dummies 與 sklearn 的 OneHotEncoder 在這網頁中都有清楚的展示，本課程今日範例中會用到前者，在之後的葉編碼中則會用到後者，所以同學不妨先了解一下寫法。



## D23：類別型特徵 - 均值編碼

### Sample Code & 作業內容

作業1 : 請仿照範例Day\_023\_Mean\_Encoder.ipynb，將鐵達尼範例中的類別型特徵改用均值編碼實作一次。

作業2 : 觀察鐵達尼生存預測中，均值編碼與標籤編碼兩者比較，哪一個效果比較好? 可能的原因是什麼?

作業請提交Day\_023\_HW.ipynb

### 參考資料

## 平均數編碼 ：針對高基數定性特徵(類別特徵)的數據處理/ 特徵工程

知乎 [**網頁連結**](https://zhuanlan.zhihu.com/p/26308272)

就實務上而言，均值編碼的意義在於當一個特徵有明顯意義，但是類別數量特別多(這裡說的"高基數")時可能有用，但最麻煩的點在於極度容易OverFitting， 所以需要不同的平滑化方式。

在課程內使用平均因子的方法只是其一，這邊的內容也介紹了另一種較複雜的平滑化方式，提供同學參考。



## D24：類別型特徵 - 其他進階處理

### Sample Code & 作業內容

本範例中，將數值型特徵做類別型編碼

作業1 : 參考範例Day\_024\_CountEncoder\_and\_FeatureHash.ipynb，將鐵達尼的艙位代碼( 'Cabin' )欄位使用特徵雜湊 / 標籤編碼 / 目標均值編碼三種轉換後，與其他類別型欄位一起預估生存機率。

作業2 : 承上題，三者比較效果何者最好?

作業請提交Day\_024\_HW.ipynb

### 參考資料

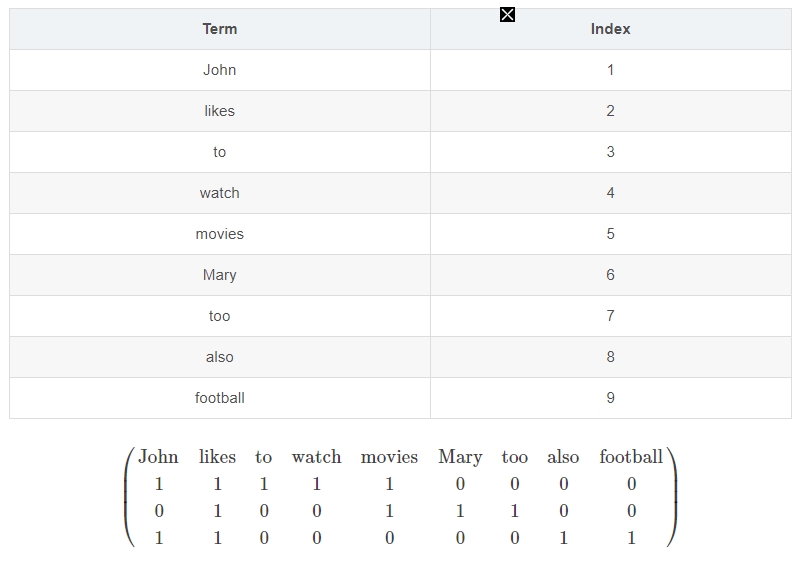
### Feature hashing (特徵哈希)

CSDN 大師魯 [**網頁連結**](https://blog.csdn.net/laolu1573/article/details/79410187)

由下圖可以理解 : 雜湊編碼是比標籤編碼(上表)更緊密的編碼方式(下表)

但要注意的是這樣的編碼 : 雖然在計算上比獨熱編碼省去很多時間，但是關鍵在雜湊後的特徵是否有意義

這邊有除了範例以外的細節講述，提供各位同學參考。

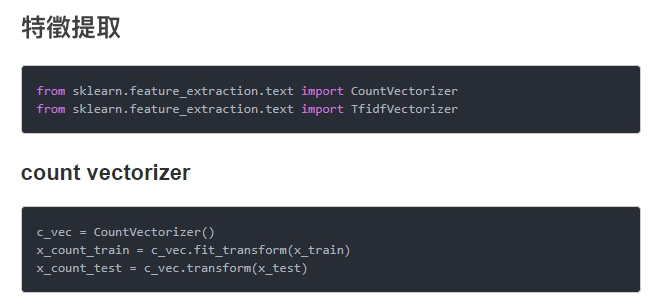


### 基於sklearn的文本特徵抽取

簡書 [**網頁連結**](https://www.jianshu.com/p/063840752151)

這裡講到的是 count vectorizer 與 tfidf vectorizer，是自然語言處理 (NLP) 時用的基礎技術之一，其中 count vectorizer 就是一種計數編碼的變形。

雖然上述兩種編碼方式現階段暫時不用弄懂，但是我們可以藉此理解 : 計數編碼有其泛用性，甚至我們可以這樣理解 - 不需要局限於我們教會各位的編碼方式，只要在您的知識中有更適合的擷取特徵方式，並且能使用程式寫作出來的，建議不妨一試，就算不是泛用的編碼法，只要包含領域知識就可能有用。



## D25：時間型特徵

### Sample Code & 作業內容

作業1 : 對照範例Day\_025\_DayTime\_Features.ipynb，試著加入星期幾 (day of week) 與第幾周 (week of year )這兩項特徵，看看結果會比原本只有時間特徵分解更好或更差?

作業2 : 對照範例的日週期效果，試著參考投影片完成年週期與周週期的特徵 (也可以用你自己想到的方式)，看看結果會比範例中的結果更好或更差?

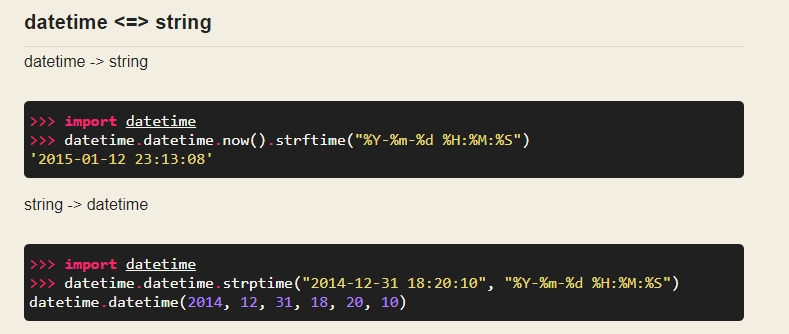
作業請提交Day\_025\_HW.ipynb

### 參考資料

### PYTHON-基礎-時間日期處理小結

wklken [**網頁連結**](http://www.wklken.me/posts/2015/03/03/python-base-datetime.html)

這裡主要是摘錄一些 datetime 套件的時間欄位操作方式，使用這一個套件就能將各式時間格式轉成統一的時間物件，方便做轉換與特徵工程。

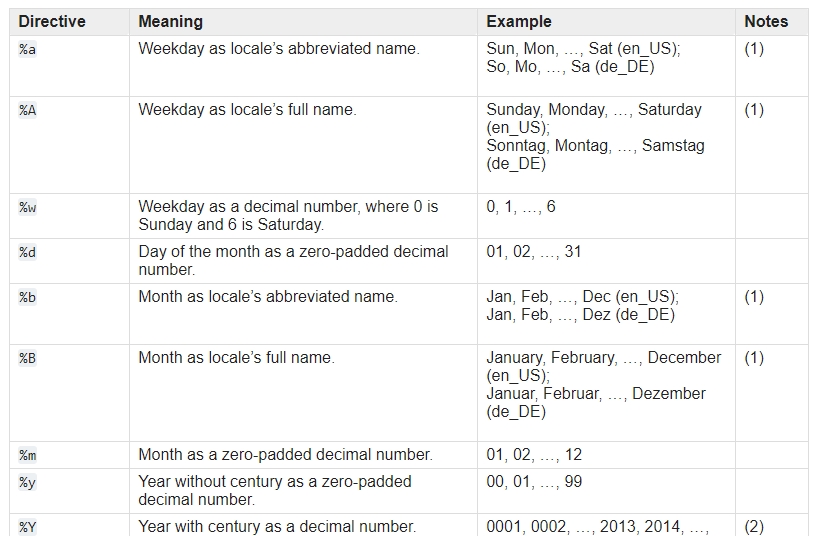


### datetime — Basic date and time types

python官方文件 [**網頁連結**](https://docs.python.org/3/library/datetime.html)

既然要使用 datetime 套件，詳細的內容當然要參考官方的文件囉。

除了各種細節的寫法，老師個人最常用到的還是 .strptime(將文字解析成時間物件) 以及 .strftime (轉出時間物件各欄位成文字)，以及右邊這張表(網頁最下方)，其中除了常見的年月日時分秒外，還有 %w(星期幾)，%U(第幾周) 這種方便的轉換函數呢!



## D26：特徵組合 - 數值與數值組合

### Sample Code & 作業內容

請參閱範例Day\_026\_Feature\_Combination.ipynb中的計程車費率預估

作業1 : 參考今日教材，試著使用經緯度一圈的長度比這一概念，組合出一個新特徵，再觀察原特徵加上新特徵是否提升了正確率?

作業2 : 試著只使用新特徵估計目標值(忽略原特徵)，效果跟作業1的結果比較起來效果如何?

作業請提交Day\_026\_HW.ipynb

### 參考資料

### 特徵組合&特徵交叉 (Feature Crosses)

SegmentFalut [**網頁連結**](https://segmentfault.com/a/1190000014799038)

這裡有一些延伸的特徵組合方式，例如講義中提到的運算組合方式，或者使用綜合特徵的離散化/分箱，或者將兩種獨熱編碼綜合...等有趣的特徵組合方式。這些方式提供同學在合成特徵時參考，但這些只是方法，合成特徵比較有效的方式還是參考領域知識(如果有的話)。

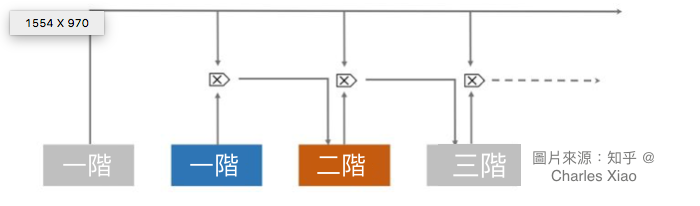


### 簡單高效的組合特徵自動挖掘框架

知乎 [**網頁連結**](https://zhuanlan.zhihu.com/p/42946318)

這裡討論的「自動學習交叉特徵」， 講的是當資料中類別種類多的類別特徵，這類問題我們一般的做法是使用LR(邏輯斯回歸) 或者FM(分解機器)...而本文則是討論FM進階的FFM。

不過這是在特徵都是屬於種類多的類別特徵時，才只好做的處理方式，如果這類特徵很少，或者在競賽中跳過這類特徵就可以有不錯準確度了，建議可以跳過這部分內容。



## D26：特徵組合 - 數值與數值組合

### Sample Code & 作業內容

請參閱範例Day\_026\_Feature\_Combination.ipynb中的計程車費率預估

作業1 : 參考今日教材，試著使用經緯度一圈的長度比這一概念，組合出一個新特徵，再觀察原特徵加上新特徵是否提升了正確率?

作業2 : 試著只使用新特徵估計目標值(忽略原特徵)，效果跟作業1的結果比較起來效果如何?

作業請提交Day\_026\_HW.ipynb

### 參考資料

### 特徵組合&特徵交叉 (Feature Crosses)

SegmentFalut [**網頁連結**](https://segmentfault.com/a/1190000014799038)

這裡有一些延伸的特徵組合方式，例如講義中提到的運算組合方式，或者使用綜合特徵的離散化/分箱，或者將兩種獨熱編碼綜合...等有趣的特徵組合方式。這些方式提供同學在合成特徵時參考，但這些只是方法，合成特徵比較有效的方式還是參考領域知識(如果有的話)。

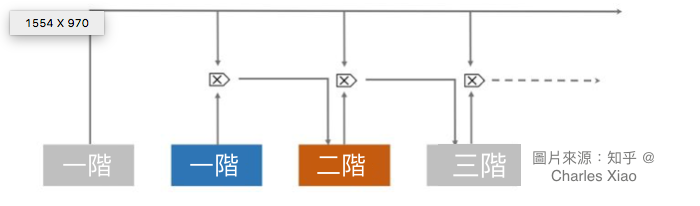


### 簡單高效的組合特徵自動挖掘框架

知乎 [**網頁連結**](https://zhuanlan.zhihu.com/p/42946318)

這裡討論的「自動學習交叉特徵」， 講的是當資料中類別種類多的類別特徵，這類問題我們一般的做法是使用LR(邏輯斯回歸) 或者FM(分解機器)...而本文則是討論FM進階的FFM。

不過這是在特徵都是屬於種類多的類別特徵時，才只好做的處理方式，如果這類特徵很少，或者在競賽中跳過這類特徵就可以有不錯準確度了，建議可以跳過這部分內容。



## D27：特徵組合 - 類別與數值組合

### Sample Code & 作業內容

請參考範例Day\_027\_GroupBy\_Encoder.ipynb

作業1 : 試著使用鐵達尼號的例子，創立兩種以上的群聚編碼特徵( mean、median、mode、max、min、count 均可)

作業2 : 將上述的新特徵，合併原有的欄位做生存率預估，結果是否有改善?

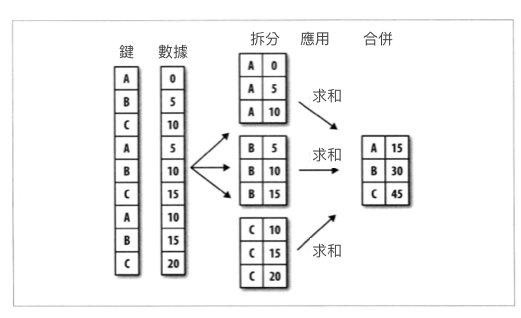
作業請提交Day\_027\_HW.ipynb

### 參考資料

### 利用 Python 數據分析之數據聚合與分組

知乎 [**網頁連結**](https://zhuanlan.zhihu.com/p/27590154)

這份網頁雖然談的也都是群聚編碼，不過卻包含相當多種組合的方式，更重要的，是包含了許多的聚類編碼程式範例，很適合在同學自己使用聚類編碼時參考。



## D28：特徵選擇

### Sample Code & 作業內容

請參考範例 Day\_028\_Feature\_Selection.ipynb

作業1 : 鐵達尼生存率預測中，試著變更兩種以上的相關係數門檻值，觀察預測能力是否提升?

作業2 : 續上題，使用 L1 Embedding 做特徵選擇(自訂門檻)，觀察預測能力是否提升?

作業請提交Day\_028\_HW.ipynb

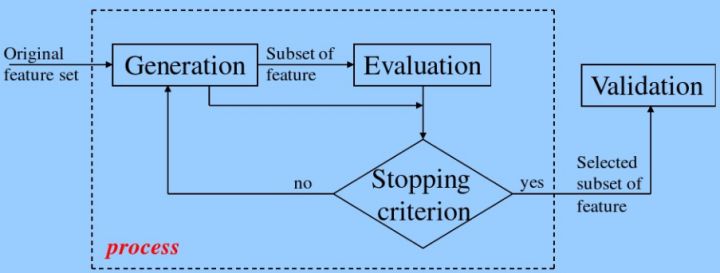
### 參考資料

### 特徵選擇

知乎 [**網頁連結**](https://zhuanlan.zhihu.com/p/32749489)

特徵選擇這一知識因為在統計時代就存在，因此教學文獻就比較多了，因此我們在這邊挑選的反而是比較精簡的說明，同學可以在網頁的內容中，了解過濾法/包裝法/嵌入法三類方法更清楚的說明。

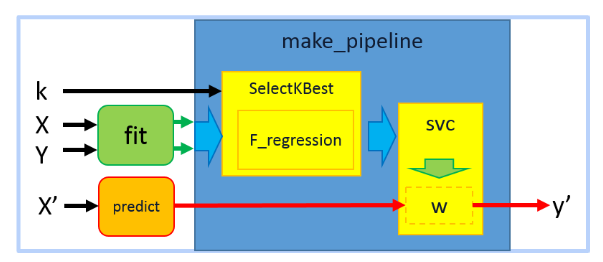
下列是特徵選擇的基礎流程，但因為執行起來很費時，因此也需要領域知識輔助，所以關鍵還是在領域知識。



### 特徵選擇線上手冊

machine-learning-python [**網頁連結**](https://machine-learning-python.kspax.io/intro-1)

對比上一參考資料的精簡，這一份手冊的說明就比較完整 : 有各式各樣的特徵選擇方式，建議同學有需要時再來查詢即可。



## D29：特徵評估

### Sample Code & 作業內容

參考範例程式碼Day\_029\_Feature\_Importance.ipynb

先用隨機森林對鐵達尼生存預測做訓練，再用其特徵重要性回答下列問題：

作業1 : 將特徵重要性較低的一半特徵刪除後，再做生存率預估，正確率是否有變化?

作業2 : 將特徵重要性最高的兩個特徵做特徵組合，是否能再進一步提升預測力?

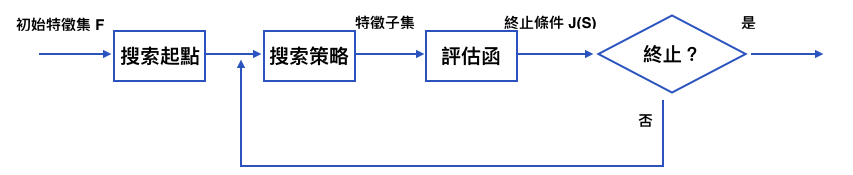
作業請提交Day\_029\_HW.ipynb

### 參考資料

### 機器學習 - 特徵選擇算法流程、分類、優化與發展綜述

掘金 [**網頁連結**](https://juejin.im/post/5a1f7903f265da431c70144c)

有關特徵選擇的優化流程，在這邊有更完整的說明，不過這篇文章與其說是說明，不如說更像一份索引，我們可以在這篇文章中找到相當多的名稱與論文選錄，建議同學在專題 / 競賽當中遇到瓶頸時，不妨來逛逛這篇，尋找一下靈感。

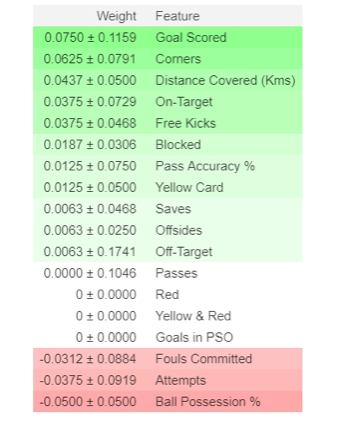


### Permutation Importance

Kaggle Dan B. [**網頁連結**](https://www.kaggle.com/dansbecker/permutation-importance?utm_medium=email&utm_source=mailchimp&utm_campaign=ml4insights)

這裡是 Kaggle 上 Dan B. 提供的課程網頁，介紹我們課程中提到的排列重要性，雖然在樹狀模型上，其精準度略遜於特徵重要性，但是這個方法在非樹狀模型上也適用，泛用性不差。

Dan B. 在這系列課程中還有其他有趣的內容，例如 SHAP Value，能將樹狀模型預測的各個特徵影響性都可解釋化，在某些應用上，這個會比精準度還要有用。



## D30：分類型特徵優化 - 葉編碼

### Sample Code & 作業內容

請參考檢視範例Day\_030\_Leaf\_Encoding.ipynb

作業1：請對照範例，完成隨機森林的鐵達尼生存率預測，以及對應的葉編碼+邏輯斯迴歸

作業2：上述的結果，葉編碼是否有提高預測的正確性呢?

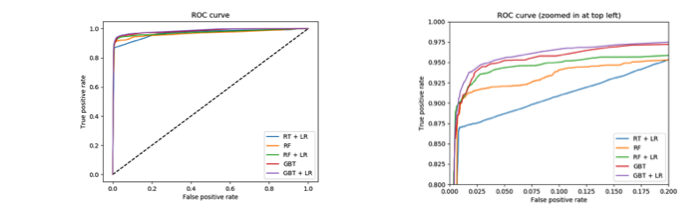
作業請提交Day\_030\_HW.ipynb

### 參考資料

### Feature transformations with ensembles of trees

sklearn官網範例 [**網頁連結**](https://scikit-learn.org/stable/auto_examples/ensemble/plot_feature_transformation.html#example-ensemble-plot-feature-transformation-py)

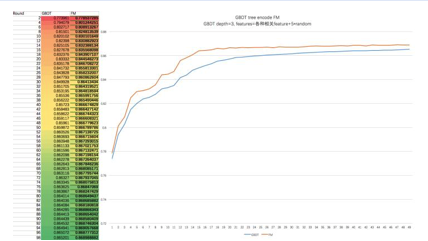
這是講義內線面這張圖的出處， 裡面有完整的葉編碼程式，因為裡面的一些細節需要多做解釋，因此我們加以註解改寫後放作為今天的範例當中，同學若是有興趣也可以直接參考這篇原文，裡面有一些原文解說。

圖片來源：scikit Learn

### CTR 預估[十一]： Algorithm-GBDT Encoder

知乎 算法那些事兒 [**網頁連結**](https://zhuanlan.zhihu.com/p/31734283)

這個網頁將葉編碼的應用，做了很詳盡的說明 : 包含使用的原因，包含的原理，以及葉編碼的戰史，如果很想弄清楚葉編碼，一定要看看這篇文章。



### 三分鐘了解推薦系統中的分解機方法（Factorization Machine, FM）

每日頭條 [**網頁連結**](https://kknews.cc/zh-tw/other/62k4rml.html)

最後是有關分解機的解說，因為這部分稍微有點複雜，需要先了解矩陣分解 ( Matrix Factorization ) 與推薦系統，如果對FM沒有興趣，可以跳過此連結，但很好奇FM到底是什麼的同學，可以由此入門。

