4.

在上一節課我們提過了機器學習是實現人工智慧的一種手段，而現在很火的深度學習則是機器學習的一個分支，他們的關西如上面這個圖異樣。

這邊用兩個簡單的例子去說明機器學習和深度學習不同的點在哪邊，假如我們要用機器學習建立一個颱風預測系統，我們首先可以去瀏覽歷年來颱風的資料，從這些資料中得出一些模式，這些模式包含了能導致颱風的具體條件，像是溫度大於40度 濕度在70到100之間之類的，而溫度與濕度這些指標就是機器學習裡的特徵，這些都是人工設計好的，也就是說我們在做這個預測系統的時候，會先由專家通過分析知道哪些特徵是重要的，然後電腦再通過分析歷史數據中的這些特徵來找出相應的模式，因此不同的特徵組合會有不同的結果

而深度學習則不一樣，他是透過模擬人類神經元的方式去做計算，簡單的例子，如果人類要去區分下面的圖娜個是方的哪個是圓的，我們眼睛第一件事情是判斷這個圖有沒有四條邊，如果有的話再看他們是不是連在一起，是不適等長與互相垂直的，如果都符合我們就會判斷他為正方形，從上面這個問題可以看出來，我們會將一個複雜的抽象問題分辨形狀，分解成比較簡單比較不抽象的小問題判斷邊長角度之類的，深度學習基本就是在模擬這個過程

以人臉識別任務來說，raw data就是原始數據，這個機器無法去理解是甚麼，於是深度學習首先盡可能地找到與這個頭像相關的各種邊角，這些邊角就是底層特徵，接下來對這些底層特徵進行組合，就可以湊出鼻子眼睛耳朵等東西，再進行組合則可以組出各種人臉，這個時候就能去辨識是否為人頭了

5.

了解工作過程後這邊簡單總結一下兩者差別，深度學習好壞很大部分取決於資料量，這張圖上可以看出當資料量多的時候dl比較好，少的時候則是ml

然後是dl很吃gpu，因為它很多任務都是用矩陣運算

最後dl不需要人工給特徵而ml要

6.

接下來說明一下ml的工作流程，資料集的部分就是看從db或是網路上爬蟲爬下來都可以，再來步驟2資料清洗是將資料變成電腦能計算的格式，步驟3特徵工程就是前面說人工給特徵的部分，這兩個我們統稱為資料前處理，也是今天的主題，再來等數據都處理好特徵都找好後，我們就可以將資料丟給模型訓練，而為了能讓模型在沒新的沒見過的資料上也表現良好，我們會將資料切成訓練用的與測試用的，訓練用的去餵給模型訓練，然後再用測試用的資料去評估模型的效能，測試效能ok後，我們才會將實際的資料丟到模型去做預測

8.

接下來就是今天的正題資料前處理，不管事做甚麼機器學習的任務，我們的第一步驟都是前處理，下面有兩個統計圖。左邊的是資料科學家或資料分析師在專案中不同步驟所花的時間，右邊則是各個步驟中哪個最討人厭，有趣的是兩張圖的前兩名都一樣，第一名都是前處裡，第二名則都是資料收集，可以發現專案大部分時間都在處理這兩件事，反而]建模與分析只占少部分時間

9.

而這邊這兩句話算是在機器學習領域很常聽到的名言，他們也很好的說明了為什麼資料前處理這麼重要，兩者都在描述同一件事，就是資料與特徵的重要性，舉個簡單的例子，假如今天我們要做一個辨認男女的二元分類問題，你給模型1000張照片，選擇將膚色瞳孔顏色作為特徵，不管甚麼算法都無法良好的區分男女，但如果我們是選染色體為特徵，娜隨便選個算法都會比前一種好

10.

接下來簡單介紹一下我們前處理常用的工具，因為我本身是寫python比較多，這邊就用python來跟大家介紹，首先是Numpy，他是python在進行科學運算得時候最基礎的package也是最核心的，其他有許多package都是在numpy的基礎上發展出來的，它的優點有下面幾點，簡單來說就是能方便計算多維度的陣列，函有許多數學方程式可以用，以及資料型別轉換方便之類的，相關的操作語法網路上很好找，也不會很難，有興趣的可以去玩看看

11.

再來就是PANDAS，有些人會叫他PYTHON的EXCEL試算表，他沒有EXCEL對資料列數與維度的限制，並結合了NUMPY與SQL的操作能力，讓操作大量資料變得比較簡單一些，然後他比較特別的資料型態有series 和dataframe兩個，series有點像帶有index的array，而dataframe有點像函有一堆series的字典，然後結構化操作像是重組切割聚和子集和之類的都能用簡單的語法完成

12.

最後是scikit-learn簡稱sklearn，他是開源的python機器學習包，裡面基本上涵蓋了所有主流的演算法，在實際專案的執行中，手寫出一個演算法的機會比較少，除了耗時外，架構是否清晰與穩定性的強弱都是問題，因此比較多時後是根據資料的特徵選擇演算法，利用像sklearn這類的package去呼叫算法，並針對結果去調整算法的參數

剛剛講的三個package操作不難網路上都有很多資料，有興趣的可以去丸看看

13.

講完使用的工具，接下來是我們資料前處理的流程，這邊是我自己去過歸納的，可能步驟的數量或名稱與網路上的資料有些不異樣，但內容都差不多相同，清洗與轉換主要是讓演算法能夠計算，特徵篩選與抽取則以不同的方式留下有用的特徵，後面會做說明

14.

接下來是一些資料常見的問題，假如這邊是我們要清洗的資料，紅色的部分代表缺失的資料，藍色代表格式不統一的資料，黃色代表不準確的資料，像是這欄應該只有N或Y兩種格式，而綠色代表重複的資料，剛剛說的這些都是比較好解決的，replace一下或是直接刪除都可以，接下來講一些比較慣用的手法

15.

首先對缺失值的處理，最簡單的方法是刪除，這概念就是刪掉缺失的部分我們就沒有缺失的問題，當然 這個方法有比較多的限制，假如資料集中含有缺失值的部分只佔整個資料集的一小部分，直接刪掉當然沒問題，但如果現在全部資料的99%都有或多或少的缺失值，全刪掉就不可行

下面一個小範例，可以發現資料裡面有空值Nan，我們可以利用pandas內建的函數對她做刪除，像是有空值就刪除，按照行或列將資料刪除，所有特徵階為空值才刪除，依照空值出現次數刪除，或是指定刪除類別之類的

16.

這邊額外補充一下缺失數據的種類，簡單的說明在上面，這邊舉一些例子比較好理解，完全隨機缺失就像假如我們做問卷調查，但某一題我們用擲骰子來決定是否回答，是否回答與其他問題沒有相關

隨機缺失像我們問收集了1000個人的身高體重年齡，發現體重缺失值30%，一查下去發現只要是30歲以下的女生體重就不回答，這情況的缺失值就跟其他變數有關

最後非隨機缺失，一樣問卷調查年收工作之類的，發現有20%的人年收是缺失值，並且我們從其他變數看也找不出原因，因此年收的缺失與否可能與他值的高低有關，也就是只與自身有關

前兩種可根據情況刪除缺失資料，隨機缺失甚至可以通過其他便量去進行估算

而非隨機缺失若直接刪除缺失值則可能導致模型出現偏差，對缺失值進行填補也比較麻煩

16.

這邊額外補充一下缺失數據的種類，簡單的說明在上面，這邊舉一些例子比較好理解，完全隨機缺失就像假如我們做問卷調查，但某一題我們用擲骰子來決定是否回答，是否回答與其他問題沒有相關

隨機缺失像我們問收集了1000個人的身高體重年齡，發現體重缺失值30%，一查下去發現只要是30歲以下的女生體重就不回答，這情況的缺失值就跟其他變數有關

最後非隨機缺失，一樣問卷調查年收工作之類的，發現有20%的人年收是缺失值，並且我們從其他變數看也找不出原因，因此年收的缺失與否可能與他值的高低有關，也就是只與自身有關

前兩種可根據情況刪除缺失資料，隨機缺失甚至可以通過其他便量去進行估算

而非隨機缺失若直接刪除缺失值則可能導致模型出現偏差，對缺失值進行填補也比較麻煩

17.

接下來介紹缺失值處理的第二種辦法，找一個數字把缺失的地方補上，而要用甚麼數字補，可以透過一些模型攻勢或是自己決定，首先是最基本的用中位數和平均數進行填補，他在python裡有寫好的function可以直接用

這邊教大家比教簡單的方式去判斷怎麼去決定用哪個補植，若資料的分布是常態分佈的話用平均數或中位數都可以，但如果資料是偏態分布或是傾斜的話，力用中位數去填補比較適合，接下來跟大家介紹一夏資料的分布

18. https://zhuanlan.zhihu.com/p/72398933

這張圖大家以前上學的時候應該都有看過，這種圖叫正態分布也較高斯分布，在機器學習里，越簡單的模型越常用，原因就是她很好被理解與解釋

這邊幫大家簡單複習一下相關的概念，假如我們現在要做一個預測模型，要去精準的預測某個變數的值，首先要做的事情就是了解這個變數的特性，舉例來說，假如我要去預測下次丟骰子可能的值，要先知道她可能值的範圍是在1到6，再來就是去確定這1到6的選項中每個值發生的概率，像是骰子只有六個面，他出現7的機率就是0，

實際操作中就是我們去重複的丟這個骰子，去紀錄每個可能結果發生的次數，有了這些資料後我們就可以去畫出概率分布的取縣就像上面的圖異樣，知道變數的分布後，接著就可以開始去估計每個結果出現的機率

而正態分布是所有概率分布中最常使用的類型，像是人的身高 每天回家所花的時間或是線性回歸中的殘差值基本上都比較接近正態分布

而正態分布的圖繪呈現一個鐘形的曲線，而他只依賴資料集中的兩個特徵，平均值和方差，平均值應該大家都知道，方差指的是衡量樣本總體偏離平均值的程度，正態分布這個統計的特性會讓預測問題變得很簡單，任何具有正態分布的變數，基本上都可以進行高精準度的預測

正態分布很容易去解釋的原因有兩個，他的平均值眾數和中位數都異樣，再來就是用平均值和標準差就可以去理解整個分布

而為甚麼正態分布這麼常見，這就會扯到一堆數學像是中心極限定理和一些概率分布函數，這個有興趣的可以上網去找資料看一下

18.

了解為什麼許多機器學習的模型都很喜歡用正態分布後，我們就可以去理解為什麼要處理非正態分布也就是偏態分不得資料了，偏態分部的突有兩種，分為副偏態和正偏態，而偏移的程度會用偏度這個統計量去衡量，圖上的名詞MODE指的是眾數MEAN是平均MEDIAN是中位數，塗上可以發現，中位數所在的位置是最接近正態分布的中心點，因此回到前面講的，當我們遇到資料傾斜成偏態分布時，用中位數去做缺失值的填充，會讓我們資料的分布更往正態分布的方向移動

18.

另外，除了偏度之外，還有一個統計輛鋒度也可以衡量資料是否為正態分布，這也分為尖峰和低鋒兩種，圖尚可以看出像尖峰分布這種類型，他的平均值也就是他鋒的地方和他尾巴的地方也就是正副兩端發生的機率都比正態分布高，而低鋒分布就正好相反，通常我們會拿偏度和鋒度搭配去衡量資料分布的狀態

19.

這邊有用偏度查看三個不同資料分布的範例，第一張為正態分布，下面的值是偏度，可以看到正態分布的話偏度是0，然後下一個是負偏態的，偏度會小於0，最後是正偏態，偏度會大於0

20.

這邊再介紹另一種檢驗正態分布的方式ks測試，他是一種非參數檢驗的方式，用於判斷樣本與給定的分布是否一致，這邊給定的分布就是正態分布，具體操作就像下面這樣，而判斷的方式是看p value，大於0.05的話就是符合正態分布，而p value是統計學裡的重要指標，用於判斷假設檢定之類的，有興趣的網路上有很多介紹

21.

然後這邊介紹其他常用的處理資料偏態的方式，下圖右邊為還沒處理過的資料，左邊是經過處理的資料，大家可以去看一下他們的偏度和資料分布的改變，第一種方式是平方根轉換，就是取每個資料的平方根，她有個限制是資料不能是負數

第二種是倒豎轉換，取每個資料的倒數，限制是只能接受非0的樹值

最後是對數轉換，取每個資料的log，限制根平方根異樣，數值不能為0

22.

接下來講回來補缺失值的事情，另一種補植方式是拉格朗日插值法，這邊做他數學上的簡單解釋，若平面上具有n個觀測值，則一定能夠求出通過這n個觀測值的一條n-1階曲線或多項式

下面他的數學式，省略掉比較複雜的計算大概可以分成三個部分，首先通過已知的觀測值得出n-1階多項式，再來將不同的值帶進去得出不同的y值，最後綱綱前面這些不同的y值的公式可以推導成這個樣子，然後就可以透過這公式得出填補缺失值的樹值

23.

綱綱的數學式換成圖片就會像這樣，圖上的點代表已知的觀測值，經過前面那些數學計算後，我們帶入任意x皆能得出一個y值

最後拿中位數補植與拉格朗日補植比較一下，塗上最左邊式函有缺失值的，中間適用中位數補值過的資料，最右邊式用拉格朗日補植後的資料，可以發現比起中位數，拉格朗日插值法能更好的還原資料原本的分布狀態

24.

接下來式多重插值法，他的概念式重複對缺失值進行補植，每次補植的過程中會加入不同的偏差，來生成許多不同的完整資料集進行比較，步驟主要就下面三個，化成圖的話會長這樣，這範例的圖上面的function是r語言的，mice()的部分就是針對缺失值用不同偏差去補植，with()的部分就是補完後得資料集就用模型取評估，pool()的部份就是選出結果最好的資料集進行後續計算，

25.

再來是用模型填充和啞便量填充，模型的部份下次課程會介紹，所以今天就先跳過，而啞便量的部分，若我們缺失的變數是離散型的，離散型變數的意思是只能用整數單位計算或代表類別的變數，假如缺失的是離散型變數且值的類型較少，像是性別，我們就可以將缺失值變成單獨的一個類別，像下面的圖，缺失的部分用other去代替

到這裡缺失值處理常見的都說玩了

26.

接下來是離群值和噪聲的處理，離群值和噪聲的定義是，噪聲是有錯的資料， 而離群值則是與大部份資料不同的資料，上面這張圖就是兩者與資料集的關西，資料集由真實數據和噪聲組成，而離群值可能是真實數據也有可能是噪聲，

當我們的資料集裡離群值或噪聲過多的話，對模型的魯棒性會有影響，像是模型的計算速度也就是收斂速度還有結果都有不良的影響，

27.

然後這邊是實際例子，用身價來舉例的話，媽斯課再地球上就算離群值，而對統計量的影響也很好理解，就像我跟他的平均身價一人有800億美金異樣，這樣的平均值來描述資料集其實是不準確的

另外假如我們在記錄這些資訊的時後，把馬斯課的身價繼承10萬台幣，這樣這筆資料就算噪聲

噪聲與離群值在統計圖上大概就長這樣，

28.

噪聲的處理常見的有分箱處理 利用聚類或迴規模型去計算或是用人工判斷，第二根第三點模型的部分異樣下捷克會解紹，這邊主要說明分相處理

29.

分箱處理是比較簡單且常用的處理方式，主要是通過箱麟的數據去替換噪聲值，箱子的分法分為等深等寬兩種，等深就是每個箱有相同比數的資料，等髖就是每個箱的區間範圍相同，

化成圖的話長這樣，像等深的每個箱內資料筆數都相同，等寬則是值的區間相同

30.

分完箱後，常見的處理方式有三種，平均值中位數和邊界值平滑處理，這邊直接用途來說明，我們把資料等深分箱後，紅字的部分為平滑處理後的部分，平均值就是用每個箱內的平均值去替代所有的值，中位數也異樣，邊界平滑則是取箱內最小的值去替換除了頭尾的數值，

這樣我們對噪聲的處理就結束了

31.

接下來是離群值的處理，常見的方法有下面這些，第四個第五個是用模型去處理，先跳過，其他會依序介紹

32.

首先最簡單的就是把資料會突會出來用看的，若資料裡有離群值就在圖上會很好判斷

33.

接著是用3∂原則去判斷，這個方法有個條件，就是資料須府和正態分布，在正態分布下，正負三標準差內的發生機率是99.7%，這之外的數值發生機率很小，因此若數值超過3備的標準差，我們便將該該數值是為離群值。塗上紅色箭頭指的部分就是離群值的範圍

34.

再來是用吸行圖的四分為距來檢測，IQR的算法為第三四分為數減掉第一四分為數，檢視的方法就是超過上四分為加上1.5備IOR或下四分為減掉1.5倍IQR，C化成箱行圖就像旁變得範例異樣

35.

接下來是基於密布的DBSCAN無監督聚類算法，這邊的無監督的意思是沒有事先標記答案的資料，答案指的是我們要用模型預測的目標，這部分在下節課會有更詳細的說明，再來聚類算法的意思是將相似的數據份組到相同的組別或CLUSTER中，

要使用DBSCAN首先要定義兩個參數，搜尋半徑與最小點數，這邊我們直接用途來解釋，搜尋半徑就是我們每個群的大小，最小點數則是指在搜尋半徑內要有多少筆資料才能算一個群，塗上的半徑就是綠色點下面那條線，而最小點數為4筆資料，另外圖上的點分為三個種類，core point核心點代表群的核心，而其他在群裡面的非核心點較邊界點border point，而兩者之外的就剩離群值outlier

DBSCAN值行的方式，首先會先選擇一個半徑內至少大於最小點的隨機資料當作核心點，接著對該群內的每個點進行評估，確定他是否在半徑內符合最小點標準，如果符合 該點會成為新的核心點，如果不符合則成為邊界點，並以此類推，

最後當某個群已經完全搜索完，在半徑內沒有其他符合最小點的資料，DBSCAN就會依照前面的步驟重新選個點進行聚類

而這種聚類方式聯不規律的分布也能區分出來，像下面這兩種分布

36.

再來是one class svm無監督式的單分類算法，跟他的名字異樣，這個算法訓練出來只分一個類別，透過正常的樣本特徵去學習決策邊界，再透過這個決策邊界去判斷新的資料是否和訓練數據相似，若超出決策邊階就是為異常值

舉個實際應用例子，假如銀行要去偵測銀行大廳內的異常行為，正常情況大部分民眾都式坐著等叫號 去取號 填寫資料或在櫃檯接受服務，這些行為雖然都會有些差異但大體上都相同，都是幅度較小的動作，因此我們將這類資料丟給分類器，他便能找出判斷動作是否為正常行為的決策邊界，因此當有人在銀行內做出與正常行為特徵不符合的動作如揮拳之類的，我們便可認定他是異常行為並發出警報

37.

下圖是他的執行方式，圖片的左邊式原來的特徵，透過kernel核函數去將這些特徵投影到高維度空間中，並且再高為空間中找出一個離原典最遠的超平面，向右圖異樣，實際分類出來的圖長得像這樣

38.

再來是獨立森林，他跟前面的DBSCAN或ONE CLASS SVM部異樣，前面的是將正常資料的範圍找出來，範圍外的就是異常值，而獨立森林則是直接去找異常值，

他的核心思想有點像切蛋糕，這邊用圖片演示，假如我們現在有個隨機的超平面對數據進行切割，切一次可以產生兩個子空間 如圖，接下來我們繼續隨機選取超平面來切割第一步驟所得到的兩個子空間，這樣一值切到每個子空間裡都只含有一個數據為止，塗上可以發現，密度較高的群裡的資料，要倍多是切割後才能分類出來，而那些分布稀疏的點則很快就能切出來，這就是這個算法找異常值的方式

39.

而他的切割步驟分為四步，從樣本裡隨機選擇N個點，然後隨機指定一個維度，並用該維度裡的某個值P去做切割，這邊為度指的就是我們資料的特徵，再來將小於P的放在節點的左邊，大於的再右邊，重複步驟2核3值到葉節點只剩一個資料或達到設定的樹的高度，

以上是單科獨立樹的訓練過程，這個算法跟他名字異樣，獨立森麟的森林代表會有很多這樣的樹去做訓練，最後再將所有樹的結果進行綜合評估來判斷異常值

40.

最後就是發現異常值怎麼去處理，處理就比發現簡單很多，像是直接刪除，或用前面介紹的處理缺失值的方法處理，或用平均值代替，或是不處理都可

處不處理離群值要看值行的任務決定，如果是一般的預測任務或分類任務，離群值就會對結果有不好的影響，但如果是做異常值偵測像信用卡盜刷這種異常偵測的任務，離群值反而會變成我們主要關注的重點

41.

到現在把基本的特徵清洗都講完了，接下來是資料前處理的第二步驟資料轉換，一般的模型都有兩個假設，一是透過訓練來使模型的準確率達到最高，另外一個是則是我們用來訓練模型的資料，他的資料分布要和測試用或者是實際上線後的資料有相同的資料分布，這邊第二點指的資料不平衡的問題，所謂的資料不平衡就像旁邊這個範例異樣，假如我們要去預測性別，但資料裡男生的量占了97%而女生只有3%，拿這種資料去訓練會有一個嚴重的問題，就是我們的模型訓練出來，在測試的時候可能準確率會高達97%，但在實際的資料上使用的話可能準確率連50%都步道，而造成這狀況的原因是，因為訓練資料裡男生佔多數，因此模型在測試時只要將結果都判斷成男性，他的準確率就會很高，但在實際上的資料男女的比例可能是1比1，因次模型全預測男的結果基本上就是亂猜，因此為了避免這種事情發生，我們需要用一些手段去平衡我們的資料

42.

首先是針對數據本身做改變的重採樣方法，分為減少多數類樣本的欠採樣與增加少數類樣本的過採樣，

下面優缺點的方式藍色是優點黑色是缺點，重採樣方法能平衡數據，增加某些分類氣的效能，其中欠採樣方法可以減少我們的數據量，在丟失重要特徵的情況下，可以縮短我們模型的計算時間

缺點則是在增加或刪減的過程，原始資料越大計算速度越慢，另外若資料是極端不平衡且躁聲過多的情況，重採樣的效果會不好，最後此類方法大部分都是用計算距離的方式去計算，因此像類別特徵或有缺失值的資料可能會不適用

接下來介紹其中比較常用得過採樣方法SMOTE

43.

SMOTE可以簡單分為四個步驟，這邊我們搭配圖片去看，首先我們將所有少數類樣本找出，並在這之中選定一個點，像圖中紅十字就是選定的點，再來找出離這個點最近的K個樣本，這邊假設K=5，途中圈起來的部分就是鄰居，再來從這5個鄰居中隨機選出一個點連線，接著在這條線上隨機找個點，而這個點就是我們生成的樣本，然後再重複1到4步驟就可以生成多個少數類樣本

基本上SOMTE的運作方式就是這樣

44.

接下來是算法及的方法，主要是去修改現有的機器學習算法來修正他們對多數類的偏好，比較常用的修改方式是代價敏改學習，代價敏改學習的概念是給少數類樣本分配較高的錯誤分類代價，像是若將少數類分類錯誤，則在下次蝶帶訓練的過程中提高少數類別的權重之類的，通過這種人為的方式在訓練過程中提高少數類別樣本的重要性，來減少模型只預測多數類的機率

優點是修改完的算法通常效果會比較好且部會增加計算時間，並且如果套用在多酚類問題上只要修改錯誤分類代價就可以用了

缺點是如何去決定錯誤分類代價需要一些DOMAIN KNOWLEDGE，不然只能TRY AND ERROR，另外就是針對不同的分類任務都需要重新設計，不能泛化到不同任務上

45.

最後是集成學習法，就是把前兩種方法去做組合，像是剛剛介紹的SMOTE加上BOOSTING算法就變成SMOTEBOOST，BOOSTING算法在下節課會跟大家介紹

優點就是這種方法大多數時間前兩者好，再來就是有些組合算法可以在碟代過程中進行動態調整，像是在碟代的過程中會自行丟棄那些模型已經能很好是分類的多數類樣本，最直接的影響就是加快模型收斂的速度，節省運算時間，

缺點的部分是，組合的算法中原本的缺點都還在，像是剛剛的SMOTE增加樣本的方式就會降低計算的速度

46.

介紹完不平衡的資料處理方式後，接下來是講如何將我們的變數變成模型比較好計算的型態，首先是針對類別行特徵，所有機器學習模型的原理都來自於數學，因此類別型的特徵像A A+ A-這種模型無法用來計算，因此我們要將這些特徵轉換成數字，轉換的情況有兩種，第一種是定序資料，指的是類別特徵有大小或好壞之類的區別，這種有順序性的我們通常會轉換成有序的數列如123這種方式，維持這類特徵有順序的特性，另一種情況是類別特徵沒有順序的特性，像是紅色綠色藍色，這種定序型的資料我們通常會將每個類別獨立成一個特徵，有該類特徵的用1表示沒有的用0表示，像範例上這樣

47.(特徵縮放補上)

接下來是數值型特徵的轉換，這類轉換也叫特徵縮放，縮放的目的主要有兩個，第一是使不同量剛的特徵處於同一數值量級，減少方差的影響讓模型更準確，第二個目的是加快模型訓練的速度，就像99\*99不好算但3\*3就比較好算的意思

常用的轉換方式有歸一化和標準化兩種，歸依化指的是將數值縮放到固定區間，像是將數值等比例所小的0到1的區間中，而標準化則是將所有述職的均質轉換為0和標準差為1的分布，他理想是轉換到正態分布但是通常實際操作上資料的分布還是會和轉換前相似

48.

接著是兩種轉換方式的差別，相同的地方就是兩者的計算方式都是用線性轉換的方式，因此部會改變原始數據的排列順序

差異的部分主要有兩點，第一點 從定義上可以明顯發現歸一化後的數值有議定的區間範圍，而標準化後的數據沒有範圍的限制，只是對他的統計特徵做出改變，

第二點就是歸一化的縮放比例僅受極值的影響，但標準化會受原始數據的平均值和標準差影響，也就是說假如我們將資料中除了最大最小值以外的數值都更換過一輪，他歸一化後的縮放比比例一樣不變，但如果是歸依化，因為原始數據改變他的平均值和標準差有很大的機率會不相同，因此縮放比例也會受到改變

49.

而用者兩種方式轉換數值的原因主要有三種，第一種 在統計建模中，不同量剛的數值會導致模型誤判，像是假如我們有兩個特徵，一個數值的區間範圍是1到10，另一個是1到1000萬，很大機率模型會認為區間範圍較大的特徵會有更大的重要性，第二點跟第一點意思差不多，用計算距離相關的算法，結果比較依賴區件較大的特徵，第三就是縮小區間範圍後計算速度會變快

50.

接下來是如何決定用哪種方法，首先 如果你對怎麼處理沒有想法，可以先用標準化處理，他比較通用，再來就是你的模型算法跟距離計算有關的話也可用標準化，然後就是對數值的範圍有要求時用歸依化，或是模型算法跟距離計算或斜方差計算無關也可用歸依化，最後就是前面講的，有極端的最大最小值存在時，因為會影響平均值與標準差，不建議用歸依化

51.

使用的時機，這邊直接記簿用轉換的模型的類型會比較老去分辨，當我們使用的模型裡沒有關於對距離的衡量或是對變數之間標準差的衡量，像決策樹模型，或是對變數的值不關心，只在意變數的分布和變數之間的條件機率的概率模型，這類的算法都不需要做處理

52.

資料轉換的部分就紹完後，資料前處理最後兩個步驟是特徵篩選與特徵抽取，這兩者的目的都是在減少無用的特徵，盡可能地只保留重要的特徵拿給模型訓練，為什麼要做這種事，這就會牽扯到機器學習裡最常見的問題維度災難，

維度災難通常會有兩個現象，第一 當模型有涉及到像量的計算文題時，維度越多，也就是特徵越多，計算量會成指數暴增

再來就是，在建模的過程中，模型的性能會隨著特徵數量的增加，先上升後下降，就像圖片這樣

53.

接下來用簡單的範例說明為神甚麼會這樣，假設我們要去做貓狗的分類，兩者照片各五張，首先我們用單一特徵像是毛色是否為紅色進行訓練，可以發現分類的效果不理想

再來我們將維度擴大到二為，增加一個例如特徵像體型大小，圖上可以發現我們依然無法用一條簡單的直線去區分兩者

接著我們再增加一個特徵像是叫生的大小之類的形成三維特徵空間，再三維特徵空間中，我們比起前面兩次的結果，更能找出一個分類平面去區分貓跟狗，這樣從一維到三維的過程中，大家應該能感受到維度越大，分類的結果越好

不過如果我們再繼續增加特徵，這邊大家可能要用意下想像力，因為圖片表現不出來三維以上的空間，從前面三個圖可以發現，隨著維度的增加，樣本之間的距離會越大，也就是樣本之間會越稀疏，這種情況很容易找到一個超平面去區分他們，但是如果我們將高維空間像低維空間投影，會發現透過高維空間訊連出的模型，相當於低維空間的一個複雜的非線性分類氣，這種模型會過於強調訓練資料的準確率甚至對一些錯誤或異常的資料也進行了學習，而正確的資料卻無法覆蓋到整個特徵空間中，因此在對新的資料進行預測的時候會出現錯誤，以圖片來說的話就是，當如果我們丟了一個貓咪的新資料進去這個模型中，如果他的特徵沒有完全符合訓練資料中貓咪的特徵，他就會被判斷成狗，這情況就是維度災難的體現

下面這張圖適用二維特徵分類的結果，可以發現雖然訓練時的準確度沒有剛剛高維度的來的好，但只要新資料與訓練資料中大部分貓咪的特徵類似，他就有機會被判斷成貓

54.