HW4

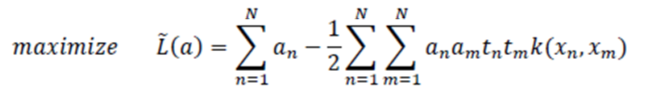
0550230 賴銘賢

0550262 陳志弘

0556171 吳耿輝

* SVM & Kernel function

不同於之前章節或作業的方式要求model的參數weight，SVM的想法是直接由training data 做linear combination直接得到weight

而SVM的dual form

其中的為kernel function，定義為

可以理解為將**原data的feature經 做non-linearity轉換至一另一空間後做內積**；另一個較直接理解kernel function方式為：假設training data為 ，testing data為 ，而如上所說，我們要直接由training data來求結果，因此直觀上理解，**假如離 越近，應該受到的label 影響越多**，因此可以把kernel function理解為跟data間距離有關的函數。

再來針對函數來做討論

* 從Polynomial二階的情況開始討論

以這次作業來說MNIST為784維度feature，所以上式的d為784，因此可以預期是一非常複雜的運算；之後將其在非線性轉換後的空間做內積可以得到kernel function

之後再加入一scaling factor ，經化簡後可以得到

再加入一項coefficient ，可以得到general form的polynomial形式

如同libsvm中調用polynomial的形式，其中Q為polynomial的階數

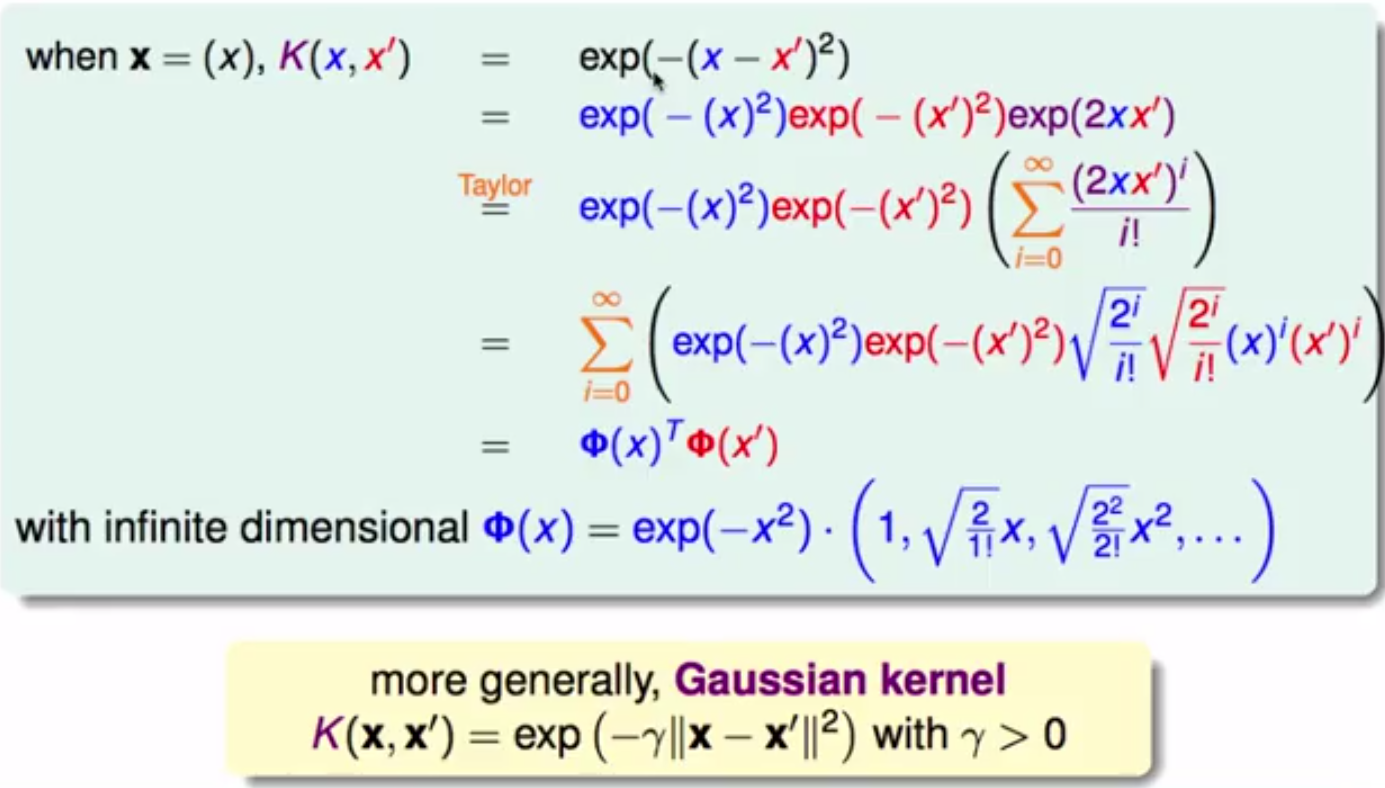
而linear kernel為polynomial中的一個特例，當

Linear kernel 使用上來說比較簡單，可以理解為沒有經過的非線性轉換，而是直接在原feature space中做處理，並且只能線性做分割。

* Radial Kernel (Gaussian Kernel)

Gaussian Kernel 可以視為 做了一個轉到**無限多維度**的空間中做處理。

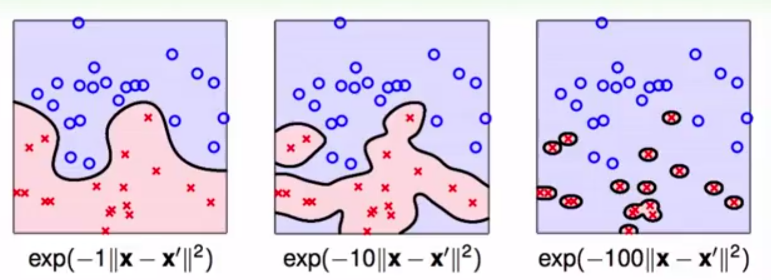
細節推導：



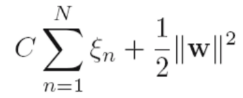
1. Gaussian Kernel與Polynomial Kernel相比，除了共同的Penalty項 C 需要調之外，Gaussian Kernel額外需調的參數只有，因此調整上較方便

2.由於Gaussian Kernel 是轉換到無限大的維度中做處理，因此可以預期其邊界可以較複雜

3. Gaussian Kernel中的會影響Gaussian Curve的形狀，當越大表示Curve越窄，因此每個data可以造成的影響範圍很小，所以只有當**x**和**x’**很近時，**x**才會影響到**x’**，但因為每個影響都很大所以可能切出形狀較奇怪的邊界，所以training error可以很低，但是有可能會Overfitting；反之越小表示Curve較寬，表示每個data的影響範圍很大，所以當**x**和**x’**很遠時，**x**也會影響到**x’**，切出來的邊界也可能較Smooth。下面為示意圖



* C-svm



C-svm中的C表示Penalty的加權大小，C越大表示Penalty越大，因此會希望每個點都能分對且距離要>=1；反之C越小就不那麼嚴苛，可以容許較多點被分錯，或是分對但是距離<=1。

而SVM中，data的假如不等於0就會被選為support vector影響邊界，support vector成分有兩種，一是分類錯誤，另一是分類正確但margin1的

**Linear Kernel**

|  |  |  |  |
| --- | --- | --- | --- |
| c | Training accuracy | Testing accuracy | **# of SV** |
| 0.0001 | 90.92 | 89.92 | **4479** |
| 0.001 | 95.98 | 94.68 | 2138 |
| 0.01 | **97.86** | **95.96** | 1024 |
| 0.1 | **99.2** | **95.8** | 692 |
| 1 | **100** | **95.08** | 701 |
| 10 | 100 | 95 | 702 |
| 100 | 100 | 95 | 702 |
| 1000 | 100 | 95 | 702 |
| 10000 | 100 | 95 | 702 |
| 100000 | 100 | 95 | 702 |

1. 可以發現當C越來越大時，penalty加重，因此training accuracy越來越高，但過度要求training data要分類正確，反而導致有些overfitting，因此testing accuracy有下降的情形

2. 另一點則是當C較小時，因為允許較多分類錯誤的點，即在svm算式中 不為0的點會變多，因此support vector數量也會較多。

**Polynomial Kernel**

C固定為1，coeff0固定為0的情況

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=2**, C=1, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.0001 | 45.12 | 46.6 | 5000 |
| 0.001 | 83.3 | 83.08 | 4316 |
| 0.01 | 98.74 | 97.28 | 1016 |
| 0.1 | 100 | 97.68 | 770 |
| 1 | 100 | 97.68 | 770 |
| 10 | 100 | 97.68 | 770 |
| 100 | 100 | 97.68 | 770 |
| 1000 | 100 | 97.68 | 770 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=3**, C=1, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.0001 | 28.02 | 28.88 | 5000 |
| 0.001 | 28.02 | 28.88 | 5000 |
| 0.01 | 98.32 | 97.04 | 1187 |
| 0.1 | 100 | 97.48 | 736 |
| 1 | 100 | 97.48 | 736 |
| 10 | 100 | 97.48 | 736 |
| 100 | 100 | 97.48 | 736 |
| 1000 | 100 | 97.48 | 736 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=4**, C=1, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.0001 | 23.44 | 23.72 | 5000 |
| 0.001 | 23.44 | 23.72 | 5000 |
| 0.01 | 95.68 | 94.04 | 1497 |
| 0.1 | 1000 | 96.36 | 775 |
| 1 | 1000 | 96.36 | 775 |
| 10 | 1000 | 96.36 | 775 |
| 100 | 1000 | 96.36 | 775 |
| 1000 | 1000 | 96.36 | 775 |

在上面固定C和coeff0的情況下，可以發現當越來越大時，training accuracy越來越好，推測原因可能是當變大時，Polynomial的變化可以越大，因此切出來的邊界可以變化更多，所以效果越來越好。

固定為0.01，coeff0固定為0的情況

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=2**, =0.01, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.001 | 45.12 | 46.6 | 5000 |
| 0.01 | 83.3 | 83.08 | 4316 |
| 0.1 | 95.82 | 94.36 | 2105 |
| 1 | 98.74 | 97.28 | 1016 |
| 10 | 99.96 | 97.76 | 771 |
| 100 | 100 | 97.68 | 770 |
| 1000 | 100 | 97.68 | 770 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=3**, =0.01, coeff0 = 0 | | | |
| c | Training accuracy | Testing accuracy | # of sv |
| 0.001 | 28.02 | 28.88 | 5000 |
| 0.01 | 67.86 | 68.04 | 4419 |
| 0.1 | 91.84 | 91.64 | 2640 |
| 1 | 98.32 | 97.04 | 1187 |
| 10 | 99.9 | 97.76 | 782 |
| 100 | 100 | 97.48 | 736 |
| 1000 | 100 | 97.48 | 736 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=4**, =0.01, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.001 | 24.75 | 25 | 4994 |
| 0.01 | 53.98 | 54.44 | 4510 |
| 0.1 | 84.28 | 84.16 | 2864 |
| 1 | 95.68 | 94.04 | 1497 |
| 10 | 99.42 | 96.84 | 909 |
| 100 | 100 | 96.6 | 779 |
| 1000 | 100 | 96.6 | 775 |

可以看出如linear kernel時的現象，C越大，Penalty越大，因此training分錯的點越來越少，train accuracy越來越高；test的accuracy也越來越高，因此推測還沒有發生overfitting的現象。

**Radial Kernel**

C固定為1的情況

|  |  |  |  |
| --- | --- | --- | --- |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.001 | **96.7** | **95.24** | 1730 |
| 0.01 | **99.08** | **97.52** | 1087 |
| 0.1 | **100** | **90.4** | 4087 |
| 1 | **100** | **30.4** | 5000 |
| 10 | **100** | **20.24** | 5000 |
| 100 | **100** | **36.6** | 5000 |
| 1000 | **100** | **20** | 5000 |

可以發現當變大時，如同前面所討論，有可能會造成**overfitting**的現象，在這邊我們可以發現當變大時，support vector的數量越來越多，到最後更是全部5000筆training data都當作support vector，因此可以想像此時的Decision boundary是繞過每個資料點，且離資料點的距離都滿近的(因為Margin1，才會被選為support vector)，因此推論此時的Decision boundary應該是非常奇怪的形狀，發生了overfitting，所training accuracy越來越高但testing accuracy卻越來越低。

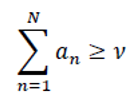
固定為0.01的情況

|  |  |  |  |
| --- | --- | --- | --- |
| **c** | **Training accuracy** | Testing accuracy | # of sv |
| **0.001** | **90.04** | 89.16 | 5000 |
| **0.01** | **93.12** | 91.8 | 4569 |
| **0.1** | **97** | 95.6 | 2146 |
| **1** | **99.08** | 97.52 | 1087 |
| **10** | **99.98** | 98.2 | 1019 |
| **100** | **100** | 98.16 | 1021 |
| **1000** | **100** | 98.16 | 1021 |

可以看出如linear kernel時的現象，C越大，Penalty越大，因此training分錯的點越來越少，train accuracy越來越高；test的accuracy也越來越高，因此推測此情況還沒有發生overfitting的現象。

* Nu – svm

與c-svm的差別在，nu-svm是調整nu來控制penalty和margin的影響，根據



不為0的data point會被選為support vector，因此當nu上升代表可以有更多點是support vector(分類錯誤，或是分類正確但margin1)；反之nu下降，則support vector數量會下降，會要求有更多的點要被分類正確且margin大於1

另外c-svm中的c可以從0到無限大，但nu-svm中的nu範圍只有0~1

**Linear Kernel**

|  |  |  |  |
| --- | --- | --- | --- |
| **nu** | **Training accuracy** | Testing accuracy | # of sv |
| 0.00001 | 100 | 94.92 | 687 |
| 0.0005 | 100 | 95 | 708 |
| 0.001 | 100 | 95 | 705 |
| 0.005 | 99.98 | 94.96 | 709 |
| 0.01 | 99.94 | 95.12 | 691 |
| 0.05 | 98.84 | 95.64 | 834 |
| 0.1 | 95.84 | 97.8 | 1179 |
| 0.3 | 95.68 | 94.4 | 2615 |
| 0.6 | 92.56 | 91.48 | 4143 |
| 0.9 | 87.64 | 87 | 4950 |
| 0.99 | 81.76 | 81.72 | 5000 |

可以看到如上述說明，當nu變大時，允許更多點被分錯或是分對但是margin1，因此support vector數量增加，而且train, test的accuracy較低；反之當nu變小時，要求分類要正確且margin都要盡量大於1，因此support vector數量降低，而且train, test的accuracy較高

**Polynomial Kernel**

nu固定為0.5，coeff0固定為0的情況

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=2**, nu=0.5, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.00001 | 71.38 | 70.96 | 2500 |
| 0.0001 | 91.88 | 90.8 | 3302 |
| 0.001 | 91.6 | 90.64 | 3317 |
| 0.01 | 91.6 | 90.64 | 3321 |
| 0.1 | 91.6 | 90.64 | 3321 |
| 1 | 100 | 97.68 | 3321 |
| 10 | 100 | 97.68 | 3321 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=3**, nu=0.5, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.00001 | 49.24 | 49.8 | 2500 |
| 0.0001 | 49.24 | 49.8 | 2500 |
| 0.001 | 87.98 | 87.32 | 3083 |
| 0.01 | 87.86 | 87.24 | 3106 |
| 0.1 | 87.86 | 87.24 | 3106 |
| 1 | 87.86 | 87.24 | 3106 |
| 10 | 87.86 | 87.24 | 3106 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=4**, nu=0.5, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.00001 | 36.44 | 38.44 | 2500 |
| 0.0001 | 36.44 | 38.44 | 2500 |
| 0.001 | 86.88 | 85.56 | 2951 |
| 0.01 | 84.1 | 83.36 | 2964 |
| 0.1 | 84.1 | 83.36 | 2964 |
| 1 | 84.1 | 83.36 | 2964 |

可以看到如同c-svm時，固定C增的情形

當越來越大時，training accuracy越來越好，推測原因可能是當變大時，Polynomial的變化會越大，因此切出來的邊界可以變化更多，所以效果越來越好。

固定為0.01，coeff0固定為0的情況

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=2**, =0.01, coeff0 = 0 | | | |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.00001 | 72.74 | 71.76 | 91 |
| 0.0005 | 100 | 97.68 | 651 |
| 0.001 | 100 | 97.64 | 703 |
| 0.005 | 100 | 97.68 | 754 |
| 0.01 | 100 | 97.68 | 764 |
| 0.05 | 99.76 | 97.96 | 825 |
| 0.1 | 98.98 | 97.48 | 1055 |
| 0.3 | 95.62 | 94.32 | 2223 |
| 0.6 | 88.8 | 88.04 | 3793 |
| 0.9 | 74.52 | 73.56 | 4854 |
| 0.99 | 54.26 | 55 | 4996 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=3**, =0.01, coeff0 = 0 | | | |
| nu | Training accuracy | Testing accuracy | # of sv |
| 0.00001 | 54.22 | 54.08 | 53 |
| 0.0005 | 100 | 97.52 | 522 |
| 0.001 | 100 | 97.76 | 598 |
| 0.005 | 100 | 97.56 | 706 |
| 0.01 | 100 | 97.56 | 721 |
| 0.05 | 99.8 | 97.8 | 826 |
| 0.1 | 99.3 | 97.56 | 993 |
| 0.3 | 93.82 | 92.8 | 2024 |
| 0.6 | 83.54 | 82.28 | 3570 |
| 0.9 | 60.74 | 60.64 | 4787 |
| 0.99 | 36.88 | 38.44 | 4993 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Degree=4**, =0.01, coeff0 = 0 | | | |
| nu | Training accuracy | Testing accuracy | # of sv |
| 0.00001 | 38.34 | 38.64 | 35 |
| 0.0005 | 98.08 | 95.12 | 430 |
| 0.001 | 99.92 | 96.64 | 525 |
| 0.005 | 100 | 96.52 | 713 |
| 0.01 | 100 | 96.6 | 744 |
| 0.05 | 99.64 | 96.92 | 870 |
| 0.1 | 98.84 | 96.72 | 1007 |
| 0.3 | 91.98 | 91.16 | 1949 |
| 0.6 | 78.36 | 77.52 | 3445 |
| 0.9 | 51.68 | 52.72 | 4744 |
| 0.99 | 90.58 | 89.72 | 4999 |

可以看出如linear kernel時的現象，nu越小，要求分類要越正確，train accuracy越來越高；test的accuracy也越來越高，因此推測還沒有發生overfitting的現象。

這邊還有發現當nu為0.00001很小的時候，Train和Test的accuracy突然變很差，推測可能是在nu-svm中是由nu直接影響support vector數量，然後此時nu太低導致他無法找到合適的解，因此得到較低的accuracy。

**Radial Kernel**

固定為0.01的情況

|  |  |  |  |
| --- | --- | --- | --- |
| **nu** | **Training accuracy** | Testing accuracy | # of sv |
| 0.00001 | 83.28 | 81.68 | 105 |
| 0.0005 | 100 | 98.08 | 876 |
| 0.001 | 100 | 98.12 | 935 |
| 0.005 | 100 | 98.16 | 1003 |
| 0.01 | 100 | 98.12 | 1010 |
| 0.05 | 99.62 | 98.04 | 1047 |
| 0.1 | 98.76 | 97.4 | 1258 |
| 0.3 | 96.74 | 95.72 | 2559 |
| 0.6 | 94.62 | 93.32 | 4050 |
| 0.9 | 91.86 | 90.76 | 4909 |
| 0.99 | 90.58 | 89.72 | 4999 |

nu固定為0.5的情況

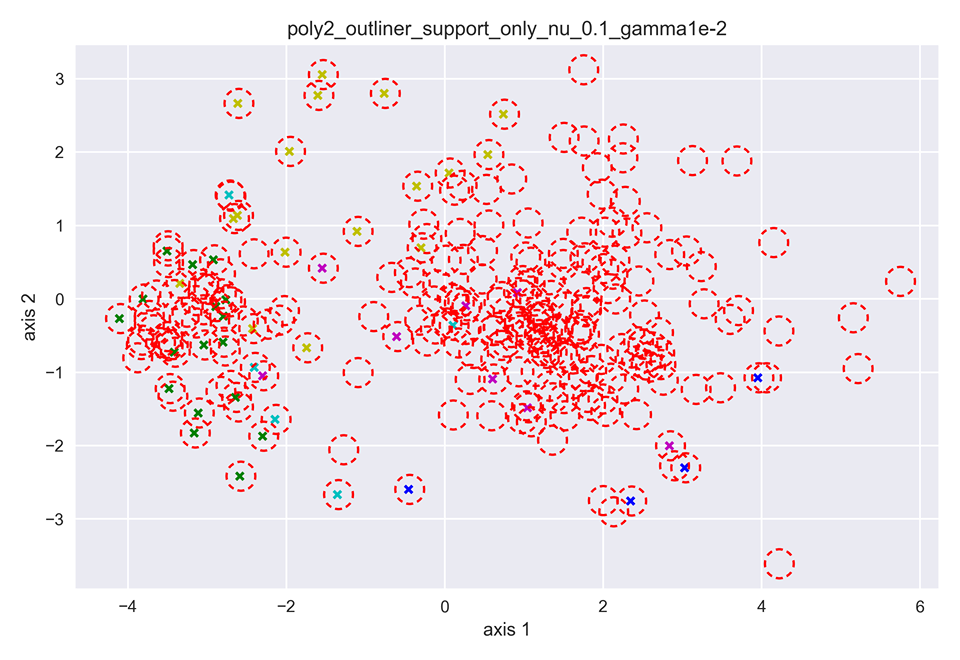
|  |  |  |  |
| --- | --- | --- | --- |
|  | Training accuracy | Testing accuracy | # of sv |
| 0.00001 | 93.84 | 92.84 | 3708 |
| 0.0001 | 93.84 | 92.84 | 3709 |
| 0.001 | 94 | 92.92 | 3702 |
| 0.01 | 95.26 | 94.32 | 3622 |
| 0.1 | **98.48** | **90.04** | 4368 |
| 1 | **100** | **31.32** | 5000 |
| 10 | **100** | **79.08** | 5000 |
| 100 | **100** | **36.6** | 5000 |
| 1000 | **100** | **20** | 5000 |

可以發現如同c-svm中固定C增加的情況，**同樣造成了overfitting**，training accuracy很高，但是testing accuracy很低。

* Support Vector vs. Outlier

這邊Outlier是指被分錯的點，在svm中被分錯的點的會不等於0，因此就會被選為Support vector，因此可以預期所有的Outlier都會是Support Vector的一部份(另一部分是分類正確但Margin1的，也會被選為Support vector)

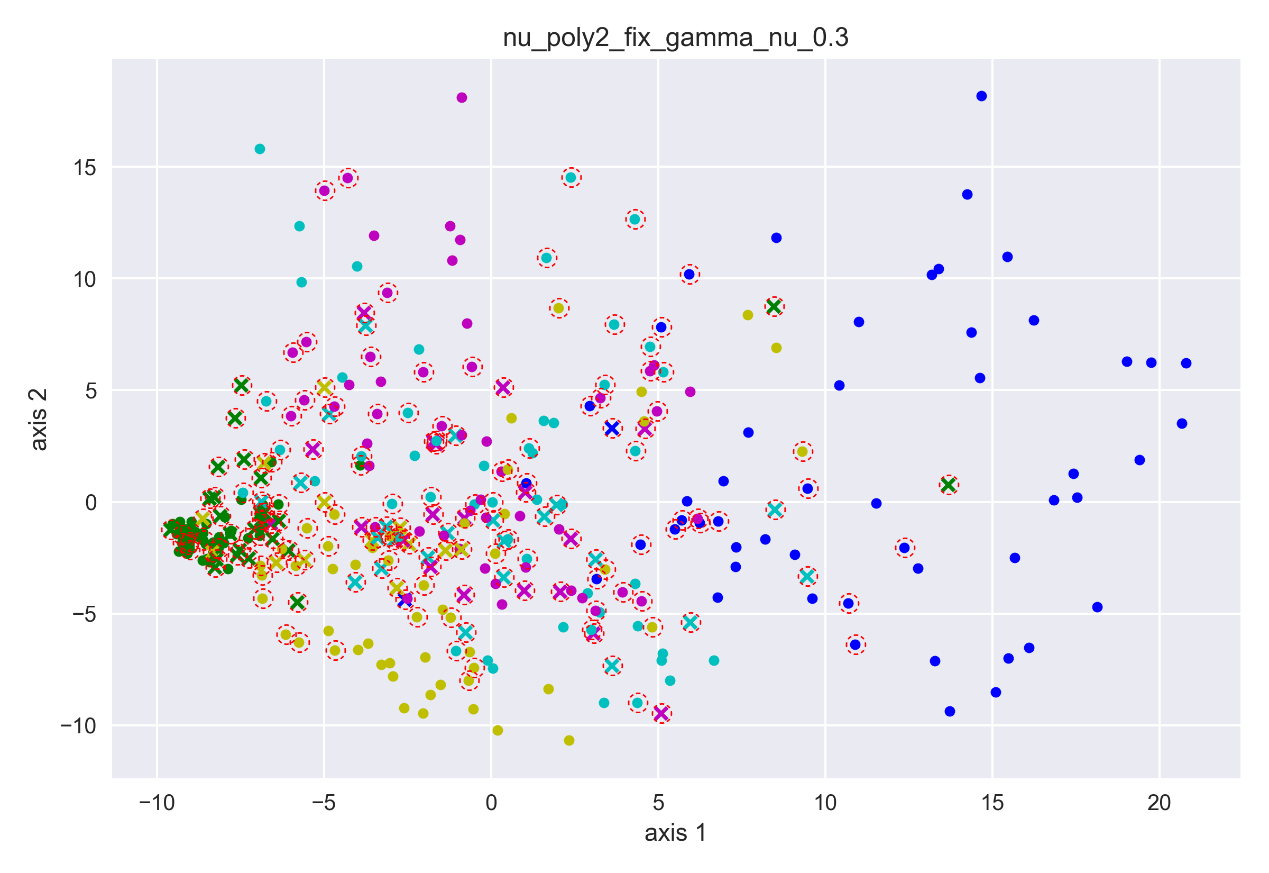
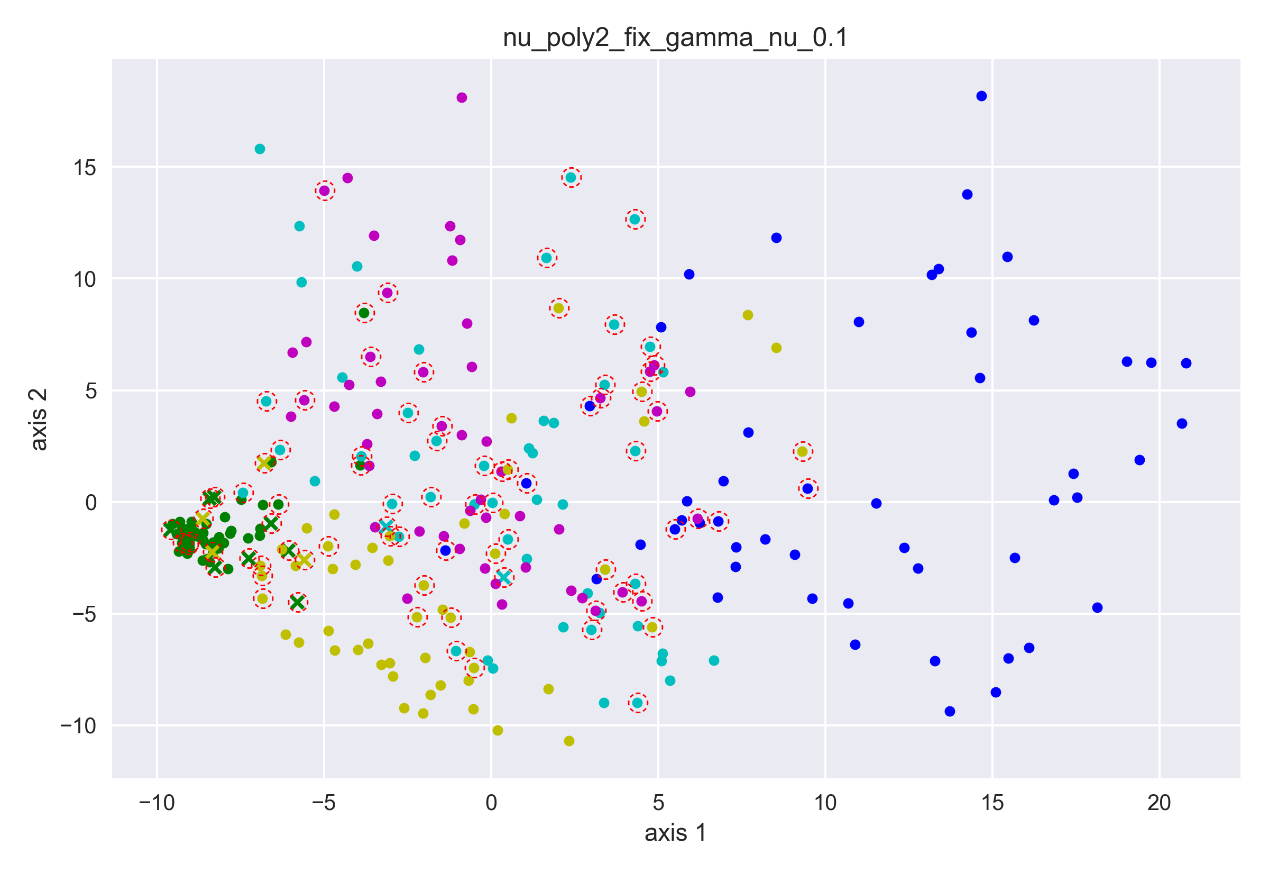
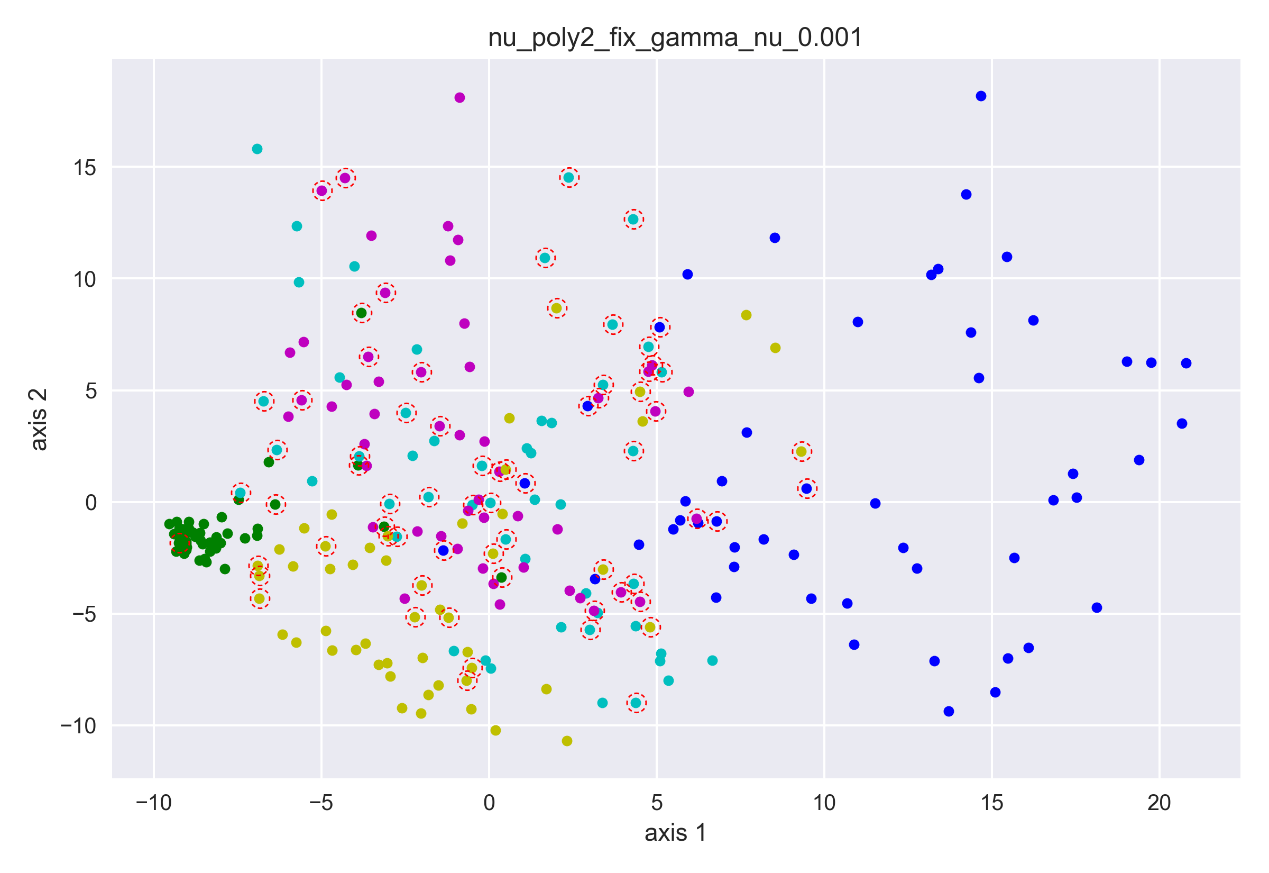
此圖為nu-svm，radial kernel，gamma=0.01的情形，其中x代表Outlier，虛線大圓圈代表support vector



可以發現x 這些outlier全都被虛線圓圈圈起，表示被選為support vector

附上nu-svm，使用二階polynomial，gamma=0.01, coeff=0

Nu從0.001🡪0.1🡪0.3 的情形，預期被分錯的點會越來越多，且都選成support vector (圓點會其他資料點，x為outlier，虛線圓圈為support vector)



* **總結&Kernel討論：**

這次作業使用Linear Kernel在合適的C值或nu值時，就能夠達到training accuracy 100%的解，因此表示training data在784維度空間時其實是線性可分得，因此Linear Kernel已經能夠達到很好的performance了；因此在使用Polynomial和Radial帶有更複雜的非線性轉換時，在合適的參數條件下也能夠達到一樣的100% training accuracy，Radial Kernel需要特別注意是否有overfitting的情形；而實際情況要選用何種Kernel和參數也需要經由Cross-Validation，或在testing data上面驗證，挑出最合適該組data特性的Kernel。

Ref : 林軒田 – 機器學習技法(<https://www.youtube.com/playlist?list=PLXVfgk9fNX2IQOYPmqjqWsNUFl2kpk1U2>)