Wine Type and Quality Classification

Content

- Abstract
- Data Preprocessing
- Wine Type Classification
    - Logistic Regression
    - LDA
    - QDA
    - K Nearest Neighbors
    - Naive Bayes
    - SVC
    - XGBoost
- Wine Quality Classification
- Result and Discussion
- Reference

Abstract

本文所使用的資料來自 UCI Machine Learning Repository 中的 **Wine Quality Data Set**，這個葡萄酒資料集的來源是來自於葡萄牙西北部的綠酒 Vinho Verde。資料集內加上酒的品質共包含了 12 個變量 (Fig. 1)，因為資料集多個變數存在右偏的特性，所以在比較 Log 和 Yeo-Johnson 轉換後選擇了後者 (Table 1 & 2)，Fig. 2 中顯示了在轉換後的資料分布，並用顏色區別紅白酒 (酒紅色為紅酒，米白色為白酒)。

本次做分類學習的步驟為先使用 Cross Validation 選出最適合這個資料集和演算法的參數，我採用的是 5-Folds CV 加上 GridsearchCV 以調整超參數的方法比較每個參數組合的表現並選出每個演算法的最佳參數，再以 Sequential Feature Selector (SFS) 和 Lasso 共同選出資料集中重要的 Features 後並用這組 Feature Group 去做學習，並以 Confusion Matrix 和 ROC curve 評估模型表現，最後再以改變 Model Complexity 的方式去看其是否會有 Overfitting 的問題。

在做葡萄酒種類分類時，我比較了七個模型，最後以 XGBoost 的準確率 99.46% 為最高，Logistic Regression, LDA, QDA 的準確率也達到 99% 以上，且在預測白酒的部分 Accuracy 均較紅酒高，在比較 Feature Importance 後發現 "total sulfur dioxide" , "chlorides" & "volatile acidity" 三個為各模型中最重要的三個變量。

在做葡萄酒品質分類時，我參考了 P. Cortez *et al.* [1] 在 2009 年發表的 "Modeling wine preferences by data mining from physicochemical properties" 中的結果顯示 SVC 在他的結果中具有最高的 Accuracy，不同於 P. Cortez *et al.* 改變 Error tolerence 的做法，我嘗試將資料集 Upsample 以解決其集中在 Quality 5~6 的部分，並成功使 Classification Accuracy 提高到 82.39% 和 78.22%。

Cross Validation → Feature Selection → Confusion Matrix & ROC → Model Complexity Test

Attribute (units)	Red wine			White wine		
	Min	Max	Mean	Min	Max	Mean
Fixed acidity (g(tartaric acid)/dm^3)	4.6	15.9	8.3	3.8	14.2	6.9
Volatile acidity (g(acetic acid)/dm^3)	0.1	1.6	0.5	0.1	1.1	0.3
Citric acid (g/dm^3)	0.0	1.0	0.3	0.0	1.7	0.3
Residual sugar (g/dm^3)	0.9	15.5	2.5	0.6	65.8	6.4
Chlorides (g(sodium chloride)/dm^3)	0.01	0.61	0.08	0.01	0.35	0.05
Free sulfur dioxide (mg/dm^3)	1	72	14	2	289	35
Total sulfur dioxide (mg/dm^3)	6	289	46	9	440	138
Density (g/cm^3)	0.990	1.004	0.996	0.987	1.039	0.994
pH	2.7	4.0	3.3	2.7	3.8	3.1
Sulphates (g(potassium sulphate)/dm^3)	0.3	2.0	0.7	0.2	1.1	0.5
Alcohol (vol.%)	8.4	14.9	10.4	8.0	14.2	10.4

Fig. 1 [1]

Data Preprocessing

從 Table 1 & 2 可以觀察到資料在轉換前存在著很大的 Skewness 和 Kurtosis，特別是 “chlorides” 的在紅白酒的偏度都超過 5，紅酒的峰度更是高達 41，在 Log 轉換後，變數普遍表現得不錯，但 “residual sugar”, “chlorides” , “sulphates” 仍不夠接近於常態分佈，但在經過 Yeo-Johnson 轉換後，紅白酒的偏度絕對值都能落在 0.15以下，峰度除了少數變量以外也幾乎都能落在 1 以下，相較 Log 的表現更為出色，故本文將選擇 Yeo-Johnson 做為資料的轉換模式 (Fig. 2-1~11)。

Red Wine

	Column	Skewness	Kurtosis	Log Transform Skewness	Log Transform Kurtosis	Yeo-Johnson Skewness	Yeo-Johnson Kurtosis
0	fixed acidity	0.98	1.13	0.46	0.14	0.00	0.04
1	volatile acidity	0.67	1.23	0.27	0.18	0.00	-0.14
2	citric acid	0.32	-0.79	0.09	-1.04	0.02	-1.08
3	residual sugar	4.54	28.62	2.26	7.18	-0.02	0.94
4	chlorides	5.68	41.72	5.07	33.61	-0.15	3.28
5	free sulfur dioxide	1.25	2.02	-0.10	-0.66	-0.01	-0.68
6	total sulfur dioxide	1.52	3.81	-0.04	-0.69	-0.00	-0.69
7	pH	0.19	0.81	0.05	0.69	-0.00	0.66
8	sulphates	2.43	11.72	1.61	5.38	0.01	0.08
9	alcohol	0.86	0.20	0.68	-0.25	0.11	-0.93

Table 1

White Wine

	Column	Skewness	Kurtosis	Log Transform Skewness	Log Transform Kurtosis	Yeo-Johnson Skewness	Yeo-Johnson Kurtosis
0	fixed acidity	0.65	2.17	0.15	0.98	-0.01	0.91
1	volatile acidity	1.58	5.09	1.14	2.72	0.01	-0.06
2	citric acid	1.28	6.17	0.61	2.90	-0.07	2.13
3	residual sugar	1.08	3.47	0.00	-1.37	0.00	-1.37
4	chlorides	5.02	37.56	4.63	32.05	-0.07	1.09
5	free sulfur dioxide	1.41	11.47	-0.83	1.12	0.03	0.74
6	total sulfur dioxide	0.39	0.57	-0.95	3.23	0.02	0.27
7	pH	0.46	0.53	0.34	0.35	-0.00	0.11
8	sulphates	0.98	1.59	0.71	0.81	0.01	-0.10
9	alcohol	0.49	-0.70	0.33	-0.88	0.06	-1.03

Table 2 另外，在轉換後的變數相關性測試中 (Fig. 3) “total sulfur dioxide” 和 “free sulfur dioxide” 兩者有最高的 0.72，似乎不太意外，因為 “total sulfur dioxide” 是指紅白酒中所含有的二氧化硫 SO_2 總量，包含未鍵結的 “free sulfur dioxide”，所以兩者具有高度正相關這點是在預料之中的。做 Variance Inflation Factor 後可以發現

“density”的值高達 15 (Table 3-1)，表示有共線性問題 [2]，故將其刪除後即可使全部變量的 VIF 都低於 5 (Table 3-2)。

Correlation test

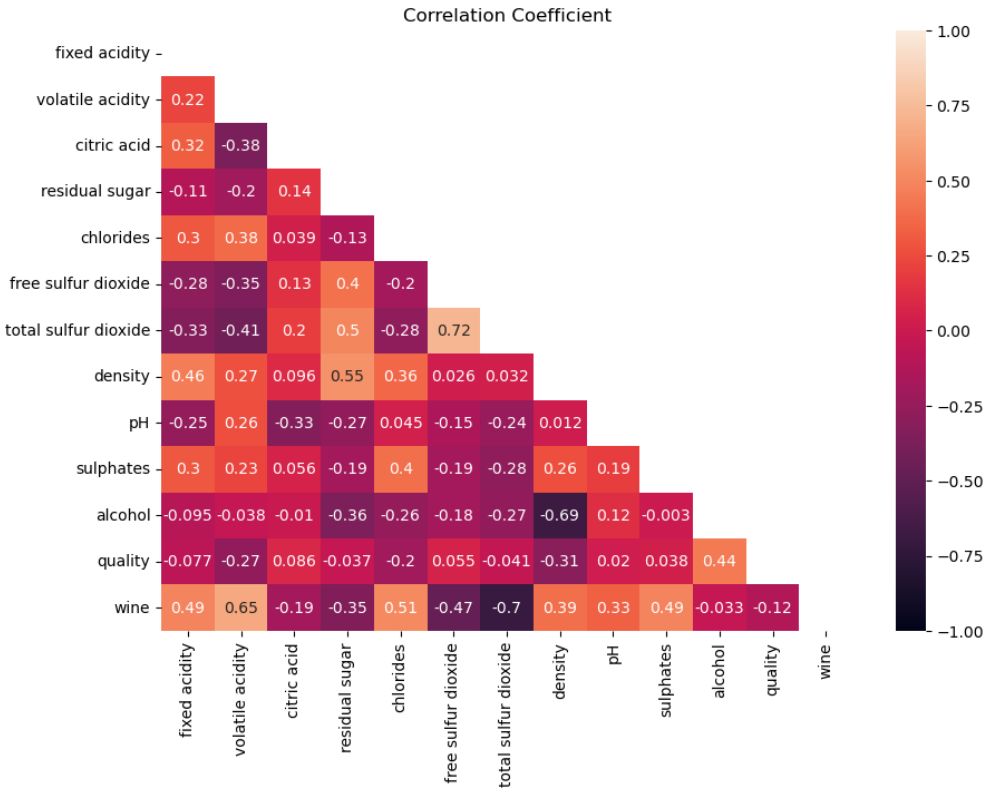


Fig. 3-1

Variance Inflation Factor

	Variable	VIF
7	density	15.964831
3	residual sugar	7.308546
10	alcohol	4.970044
0	fixed acidity	4.911189
6	total sulfur dioxide	2.974040
8	pH	2.545764
5	free sulfur dioxide	2.156281
1	volatile acidity	2.037955
4	chlorides	1.632490
2	citric acid	1.608690
9	sulphates	1.565737
11	quality	1.412703

	Variable	VIF
6	total sulfur dioxide	2.873884
5	free sulfur dioxide	2.156276
1	volatile acidity	1.809241
0	fixed acidity	1.783922
9	alcohol	1.697222
2	citric acid	1.608526
4	chlorides	1.565547
3	residual sugar	1.532924
7	pH	1.416984
10	quality	1.408249
8	sulphates	1.365039

Table 3-1 & Table 3-2

Wine Type Classification

先說結論，在 Wine type Classification 用來分類的七個演算法當中，所有的 Accuracy 都達到了 95% 以上，甚至 Logistic Regression, LDA, QDA, XGBoost 都達到 99% 以上，而最高的是 XGBoost 的 99.46%。且在所有演算法中，預測白酒的精準度都比預測紅酒來得高，我認為這非常合理，因為兩種類的資料集大小本身存在差異 (白酒: 4898 ; 紅酒: 1599)，有更多的資料可以針對白酒做學習，自然精準度也會較高。

另外，在所有模型中，"total sulfur dioxide" , "chlorides" & "volatile acidity" 均為 Feature Importance 的前三名，其分別對葡萄酒種類在 95% Confidence interval 下的機率分佈如 Fig. 4-1~3 (Wine = 1 紅酒 ; Wine = 0 白酒)，而 Feature Importance 相對非常低的 "residual sugar" , "alcohol" , "free sulfur dioxide" 分佈如 Fig. 4-4 ~ 6。可以觀察到重要的變量其分佈都是非常鮮明的，處在 0 < P < 1 的區間非常小，且由 Confidence interval 可以看到其標準差都很小。反觀三組較不重要的變數，"residual sugar" 和 "alcohol" 均只有分佈在白酒的區域，也可以看到這兩個變量在紅酒的部分幾乎沒有一個明顯的分佈，且峰度分別低達 -1.08 和 -0.93 (Table 2)。而 "free sulfur dioxide" 則是約有 1/3 的區間在 0 < P < 1 的區域，且從 Confidence interval 可以發現其標準差非常的大。

以下為各演算法的個別表現，內容主要會包含四個部分

1. Grid Search Cross Validation: 以參數調整的方式找出該演算法表現最佳的參數
2. Feature Selection: 利用 Stepwise Selection 選出重要性較高的變量，以減少變量的方式降低 Overfitting 的可能性和雜訊的產生
3. Model Performance: 在選完變數後即開始以該演算法進行學習，表現評估的部分包括 Confusion Matrix, Accuracy, Precision, Recall, F1 score, ROC curve 和 AUC，最後以該模型最重要的兩個變量繪製 Decision Region 以觀察模型的分類方法
4. Model Complexity: 最後利用改變模型複雜度以驗證是否有 Overfitting 的現象，若有 Overfitting 的現象產生時，會出現如 KNN model 在減少 n_neighbors 時出現的 test error 突然高起的狀況，但這個狀況在這次的學習中幾乎沒有發生。

Table 3	Logistic Regression	LDA	QDA	KNN	Naive Bayes	SVC	XGBoost
Accuracy	0.9905	0.9906	0.9931	0.9517	0.9868	0.9834	0.9946

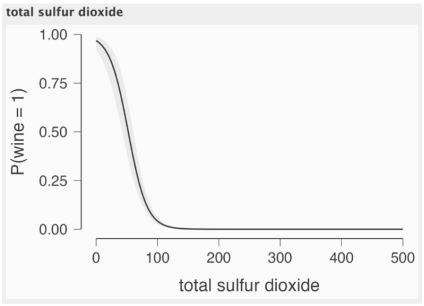


Fig. 4-1

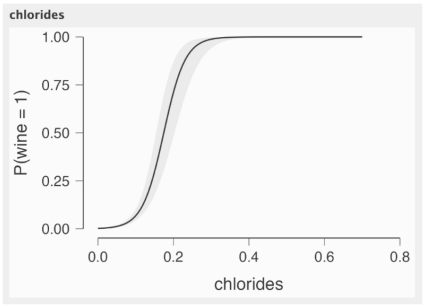


Fig. 4-2

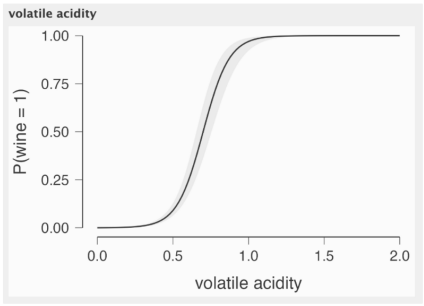


Fig. 4-3

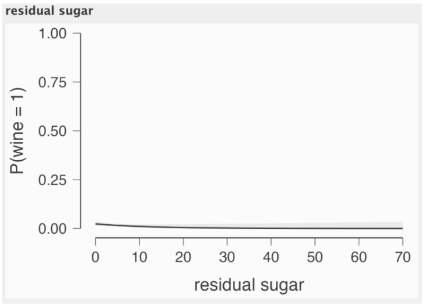


Fig. 4-4

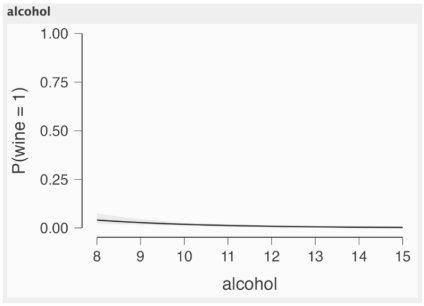


Fig. 4-5

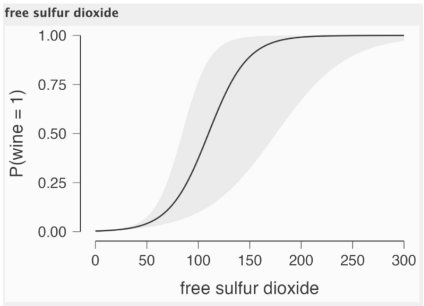


Fig. 4-6

Lasso Regression as Feature Selection

```
importance > 0.01

['volatile acidity', 'free sulfur dioxide', 'sulphates', 'quality']
```

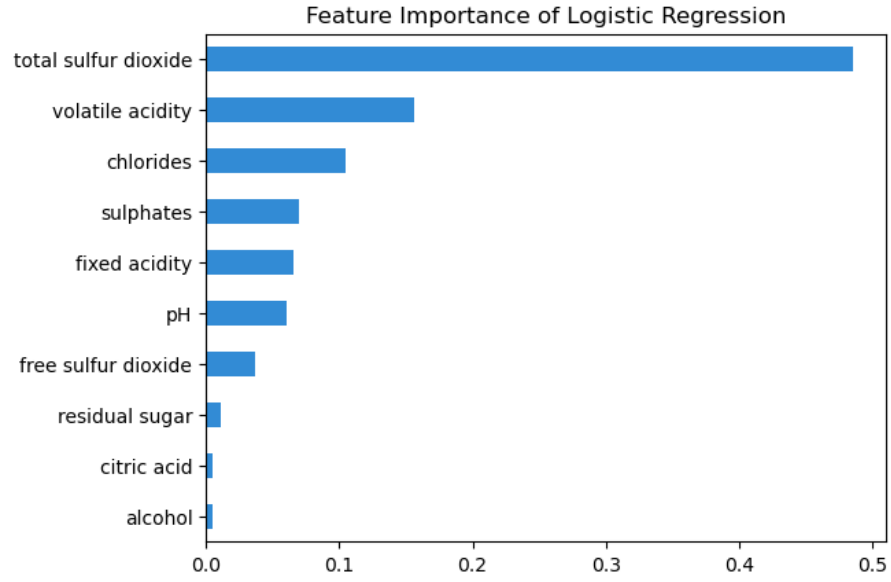
Logistic Regression

Cross Validation

```
params = {
    'C': [ 10**i for i in range(-4, 5) ],
    'penalty': ['l1', 'l2', 'elasticnet', 'None'],
    'solver': ['lbfgs', 'liblinear', 'saga']
}

Best params: {'C': 10000, 'penalty': 'l2', 'solver': 'lbfgs'}
Best scores: 0.9905
```

Feature Selection

```
SFS = ['volatile acidity', 'chlorides', 'free sulfur dioxide', 'total
sulfur dioxide', 'sulphates']
Lasso = ['volatile acidity', 'free sulfur dioxide', 'sulphates',
'quality']

Feature Group = ['fixed acidity', 'volatile acidity', 'citric acid',
'residual sugar', 'chlorides', 'free sulfur dioxide', 'total sulfur
dioxide', 'sulphates', 'quality']
```



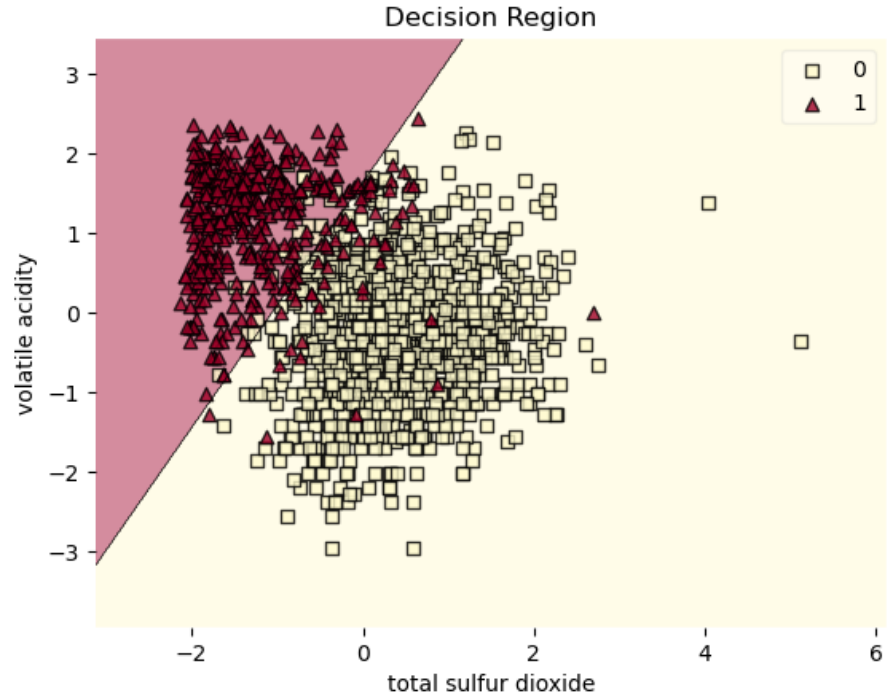


Fig. 6-1 & Fig. 6-2

Model Performance

	Predict White	Predict Red
True White	0.9949	0.0050
True Red	0.0161	0.9839

Acurracy : 99.2000%

Precision : 99.1995%

Recall : 99.2000%

f1 : 99.1997%

Model Complexity

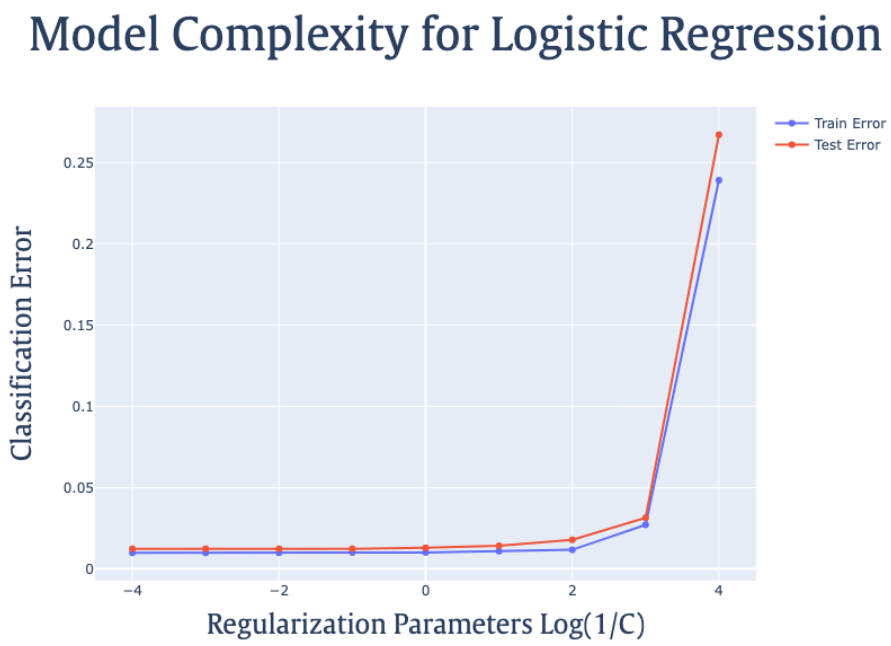


Fig. 6-3

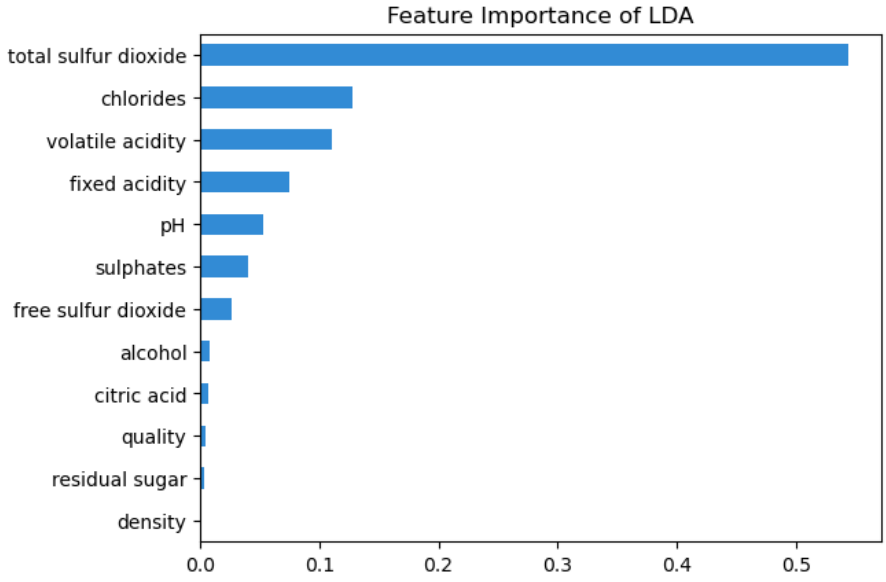
Linear Discriminant Analysis

Cross Validation

```
shrinkage_range = [10**i for i in range(-10, 1)]

params = {
    'solver': ['svd', 'lsqr', 'eigen'],
    'shrinkage': ['auto', None, shrinkage_range],
}

Best params: {'shrinkage': 'auto', 'solver': 'lsqr'}
Best scores: 0.9906
```

Feature Selection

```
SFS = ['fixed acidity', 'volatile acidity', 'citric acid', 'residual
sugar', 'chlorides', 'free sulfur dioxide', 'total sulfur dioxide']
Lasso = ['fixed acidity', 'volatile acidity', 'chlorides', 'pH',
'sulphates', 'alcohol']

Feature Group = ['fixed acidity', 'volatile acidity', 'citric acid',
'residual sugar', 'chlorides', 'free sulfur dioxide', 'total sulfur
dioxide', 'pH', 'sulphates', 'alcohol']
```



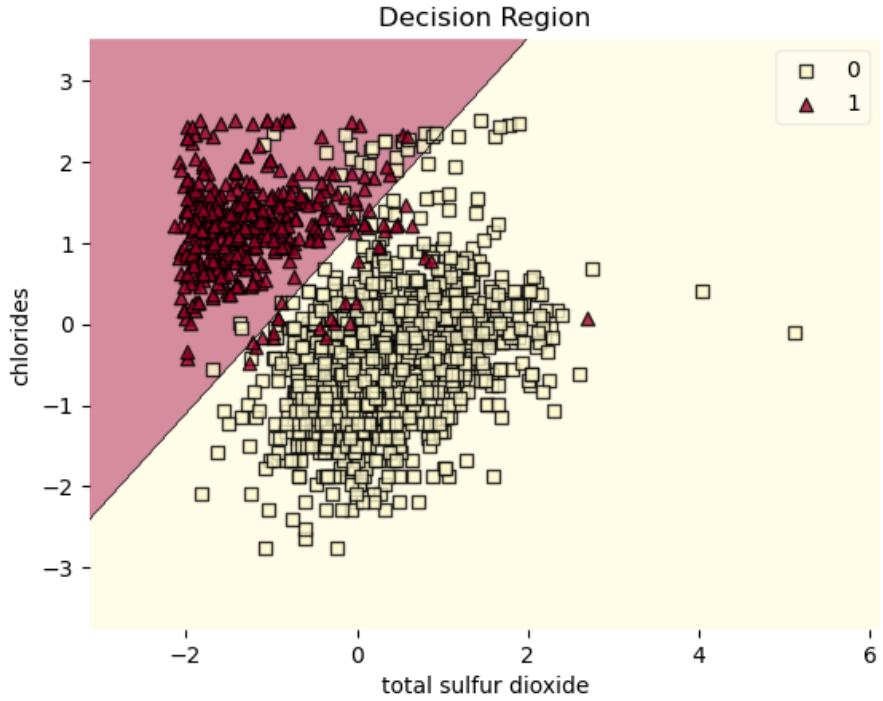


Fig. 7-1 & Fig. 7-2

	Predict White	Predict Red
True White	0.9924	0.0076
True Red	0.0230	0.9970

Acurracy : 98.8308%

Precision : 98.8300%

Recall : 98.8308%

f1 : 98.8308%

Model Complexity

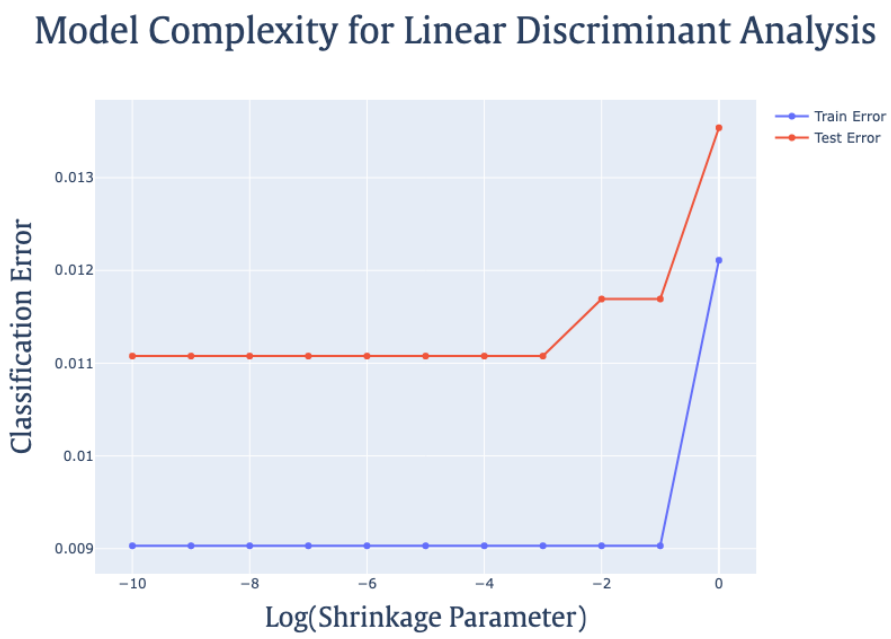


Fig. 7-3

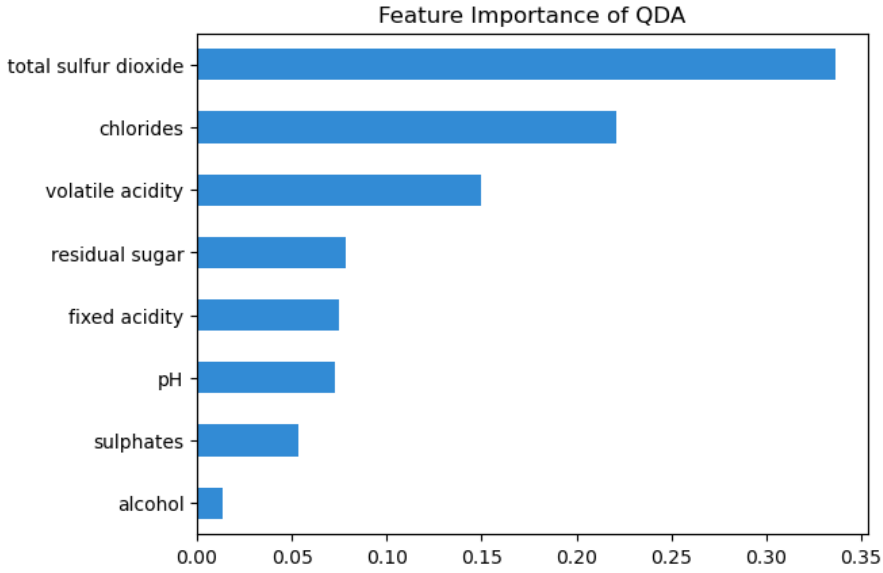
Quadratic Discriminant Analysis

Cross Validation

```
reg_param = [10**i for i in range(0, -10, -1)]

params = {
    'reg_param': reg_param,
}

Best params: {'reg_param': 1e-07}
Best scores: 0.9931
```

Feature Selection

```
SFS = ['volatile acidity', 'residual sugar', 'chlorides', 'total sulfur
dioxide', 'sulphates']
Lasso = ['fixed acidity', 'volatile acidity', 'chlorides', 'pH',
'sulphates', 'alcohol']

Feature Group = ['volatile acidity', 'residual sugar', 'chlorides', 'total
sulfur dioxide', 'sulphates', 'fixed acidity', 'pH', 'alcohol']
```



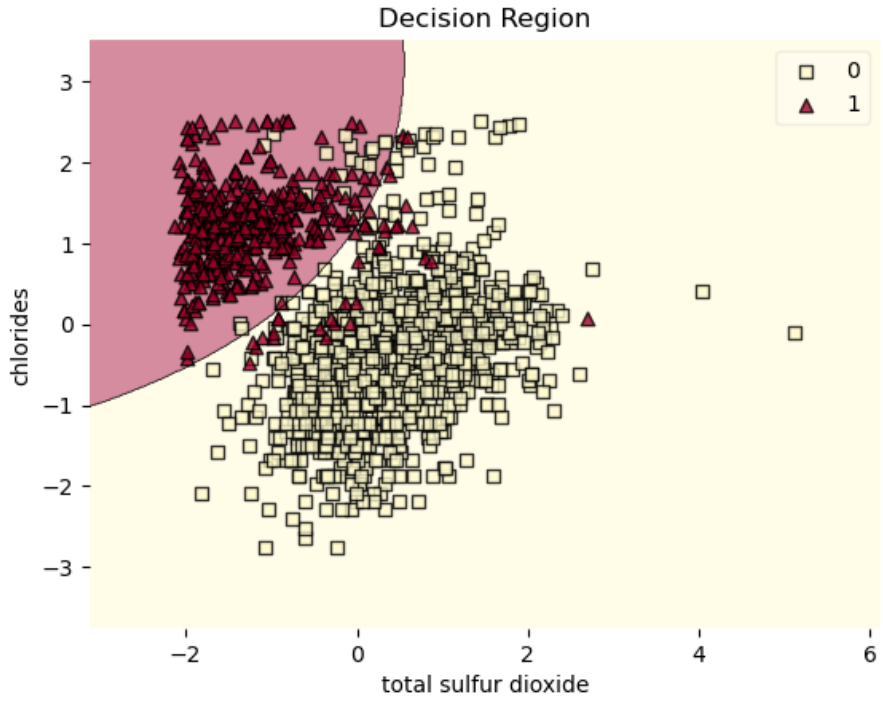


Fig. 8-1 & Fig. 8-2

Model Performance

	Predict White	Predict Red
True White	0.9966	0.0034
True Red	0.0138	0.9862

Acurracy : 99.3846%

Precision : 99.3841%

Recall : 99.3846%

f1 : 99.3842%

Model Complexity

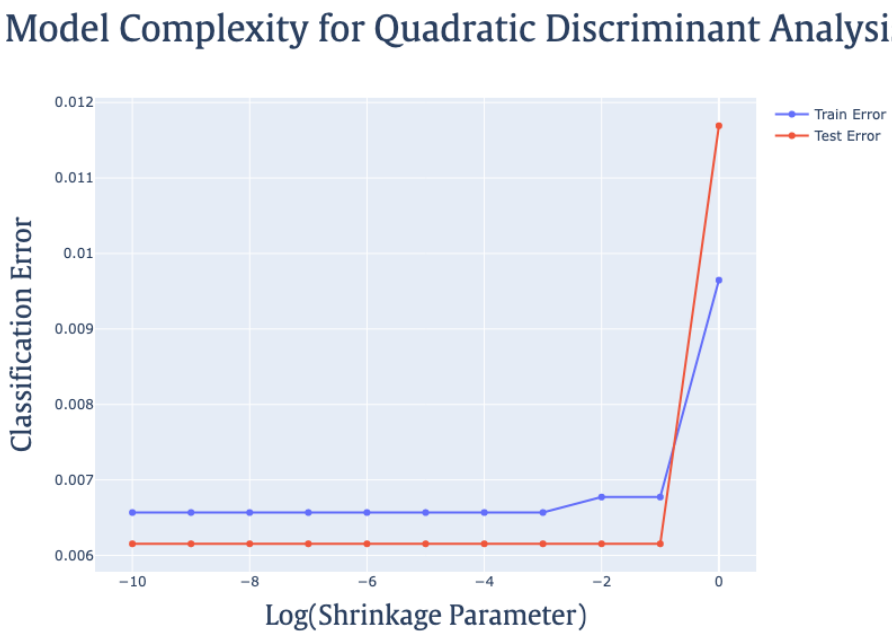


Fig. 8-3

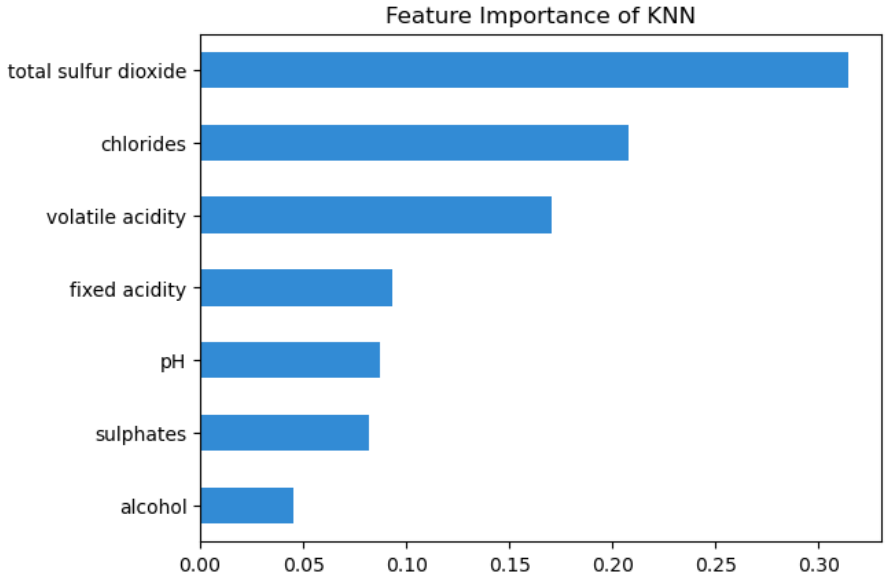
K Nearest Nieghbors

Cross Validation

```
params = {
    'n_neighbors': list(range(1, 11)),
    'leaf_size': list(range(10, 41, 5)),
    'p': [1, 2],
    'weights': ['uniform', 'distance'],
    'metric': ['euclidean', 'manhattan', 'minkowski']
}

Best params: {'leaf_size': 10, 'metric': 'manhattan', 'n_neighbors': 9,
'p': 1, 'weights': 'distance'}
Best scores: 0.9517
```

Feature Selection

```
SFS = ['chlorides', 'total sulfur dioxide', 'alcohol']
Lasso = ['fixed acidity', 'volatile acidity', 'chlorides', 'pH',
'sulphates', 'alcohol']

Feature Group = ['fixed acidity', 'volatile acidity', 'chlorides', 'total
sulfur dioxide', 'pH', 'sulphates', 'alcohol']
```



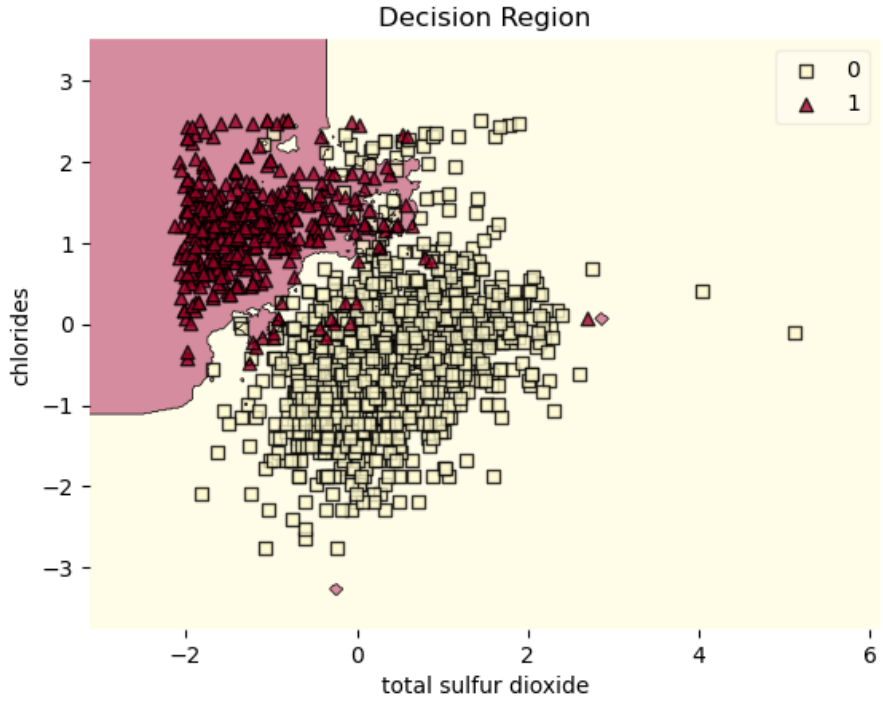


Fig. 9-1 & Fig. 9-2

Model Performance

	Predict White	Predict Red
True White	0.9966	0.0034
True Red	0.0115	0.9885

Acurracy : 99.4462%

Precision : 99.4458%

Recall : 99.4462%

f1 : 99.4460%

Model Complexity

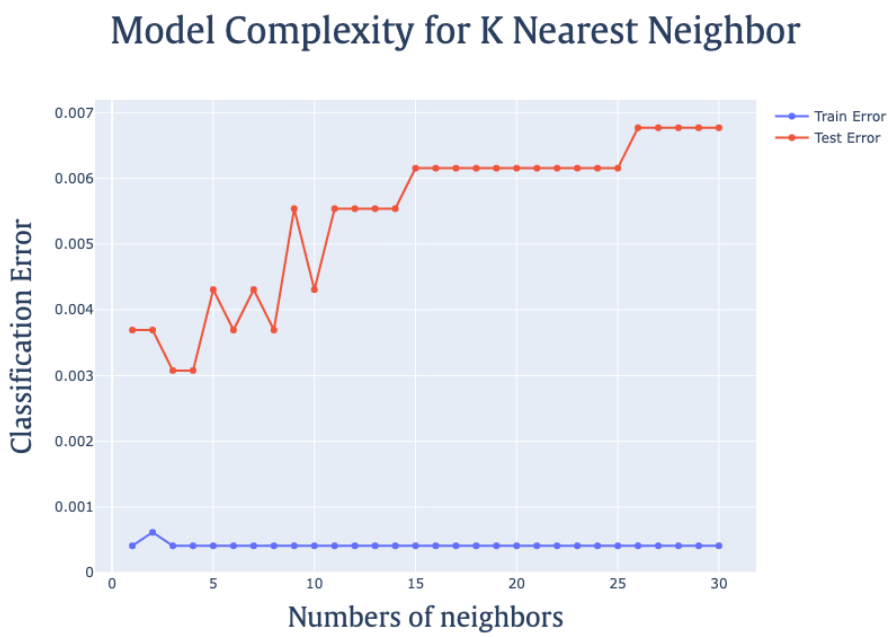
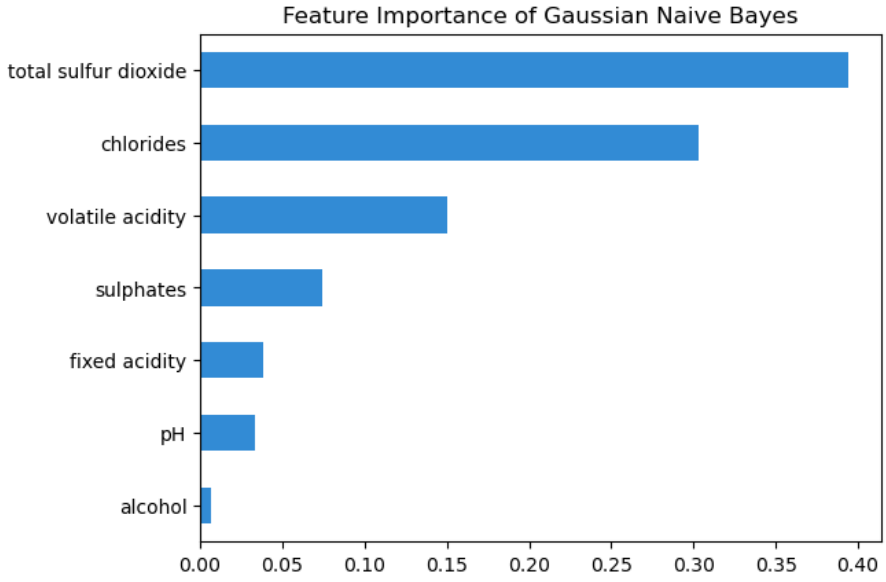


Fig. 9-3

Gaussian Naive Bayes

Cross Validation

```
var_smoothing = [10**i for i in range(0, -15, -1)]

params = {
    'var_smoothing': var_smoothing
}

Best params: {'var_smoothing': 1e-10}
Best scores: 0.9868
```



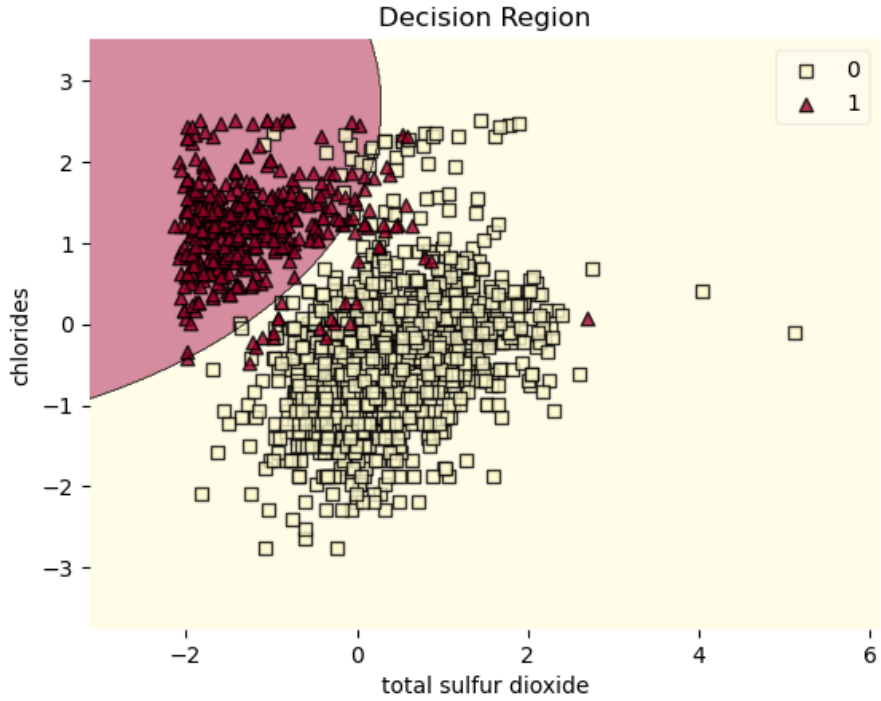


Fig. 10-1 & Fig. 10-2

Model Performance

	Predict White	Predict Red
True White	0.9966	0.0034
True Red	0.0277	0.9723

Acurracy : 99.0154%

Precision : 99.0157%

Recall : 99.0154%

f1 : 99.0125%

Model Complexity

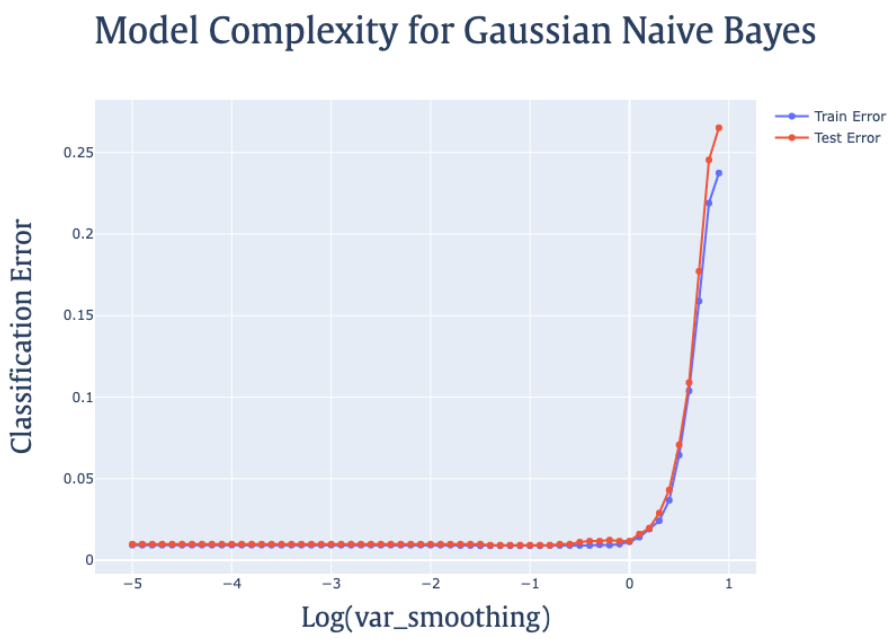


Fig. 10-3

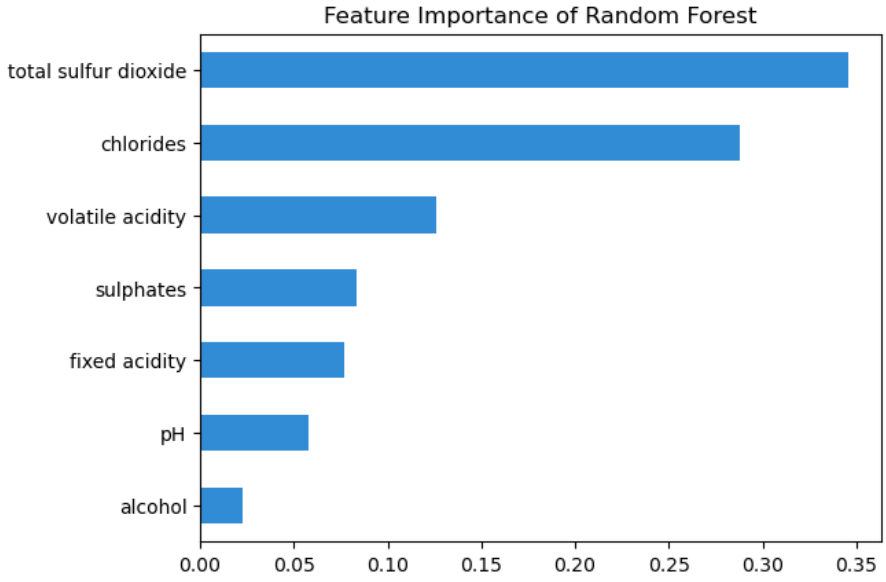
Random Forest

Cross Validation

```
params = {
    'n_estimators': list(range(100, 501, 50)),
    'max_depth': [1, 3, 5, 7, 9],
    'max_features': ['sqrt', 'log2'],
    'criterion': ['gini', 'entropy']
}

Best params: {'criterion': 'entropy', 'max_depth': 9, 'max_features':
'log2', 'n_estimators': 250}
Best scores: 0.9923
```

Feature Selection

```
SFS = ['fixed acidity', 'volatile acidity', 'chlorides', 'total sulfur
dioxide', 'sulphates']
Lasso = ['fixed acidity', 'volatile acidity', 'chlorides', 'pH',
'sulphates', 'alcohol']

Feature Group = ['fixed acidity', 'volatile acidity', 'chlorides', 'total
sulfur dioxide', 'sulphates', 'pH', 'alcohol']
```



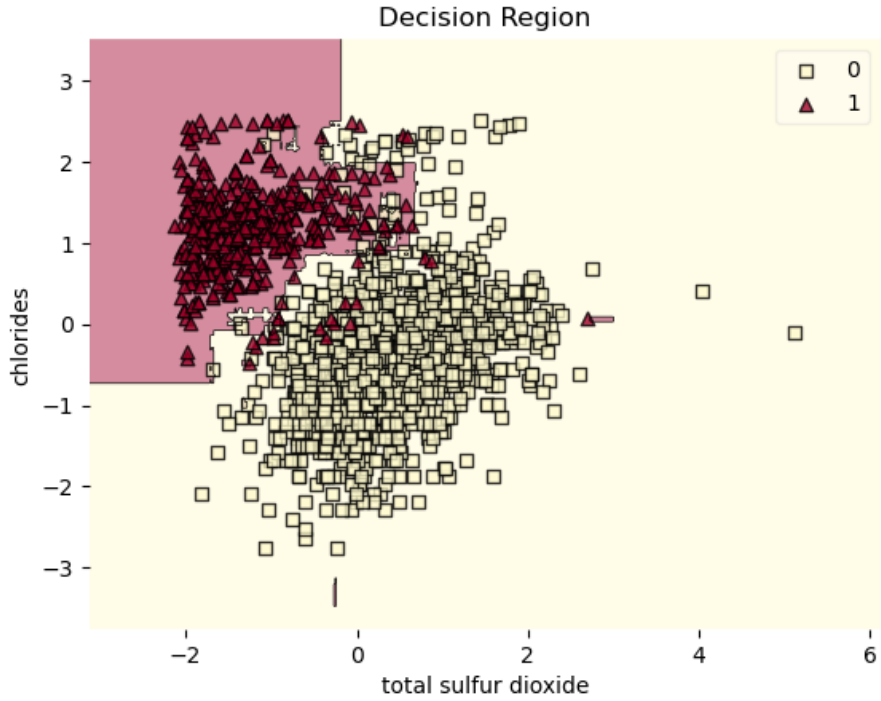


Fig. 11-1 & Fig. 11-2

Model Performance

	Predict White	Predict Red
True White	0.9991	0.0008
True Red	0.0161	0.9839

Acurracy : 99.5077%

Precision : 99.5090%

Recall : 99.5077%

f1 : 99.5066%

Model Complexity

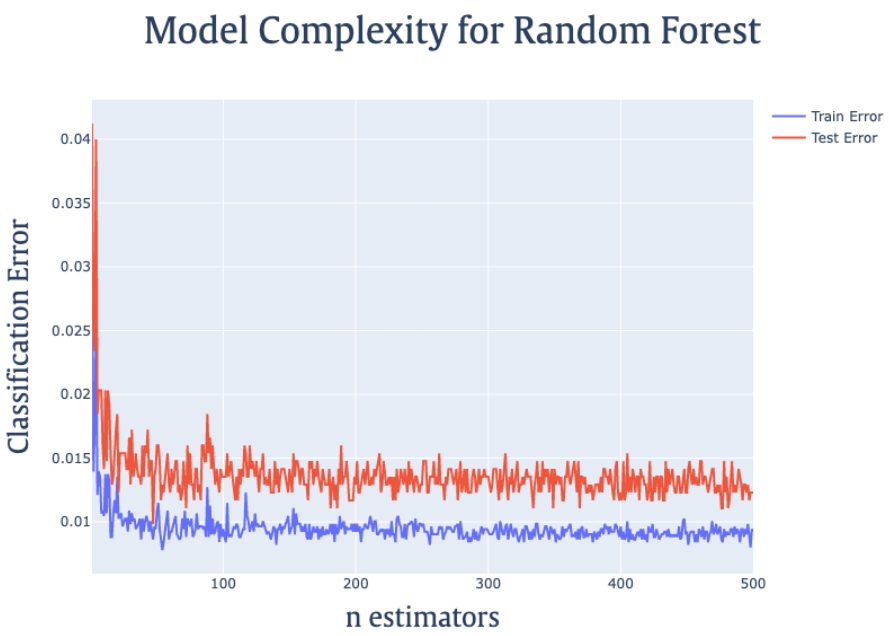


Fig. 11-3

Decision Tree

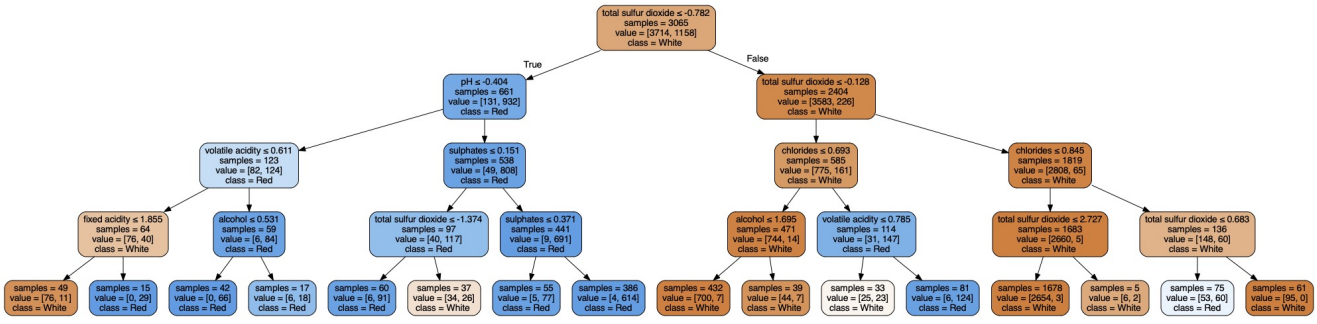


Fig. 11-4 越藍表示被分類到 Red Wine 的機率越大；越橘表示被分到 White Wine 的機率越大

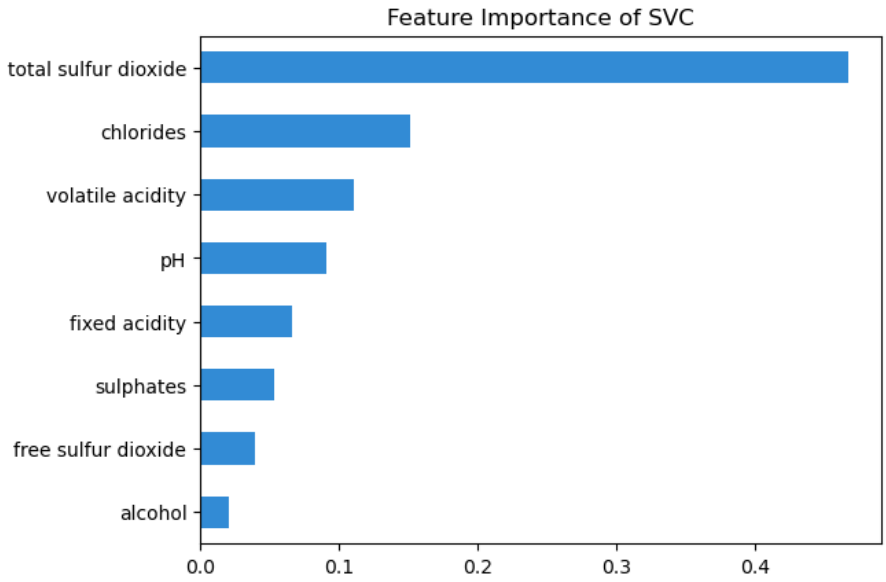
Support Vector Classifier

Cross Validation

```
params = {
    'C': np.logspace(-3, 3, num = 7, base = 10),
    'kernel': ['rbf', 'sigmoid'],
    'gamma': np.logspace(-3, 3, num = 7, base = 10),
}

Best params: {'C': 1000.0, 'gamma': 0.01, 'kernel': 'rbf'}
Best scores: 0.9834
```

Feature Selection

```
SFS = ['volatile acidity', 'chlorides', 'total sulfur dioxide',
'sulphates', 'free sulfur dioxide']
Lasso = ['fixed acidity', 'volatile acidity', 'chlorides', 'pH',
'sulphates', 'alcohol']

Feature Group = ['fixed acidity', 'volatile acidity', 'chlorides', 'total
sulfur dioxide', 'sulphates', 'pH', 'alcohol', 'free sulfur dioxide']
```



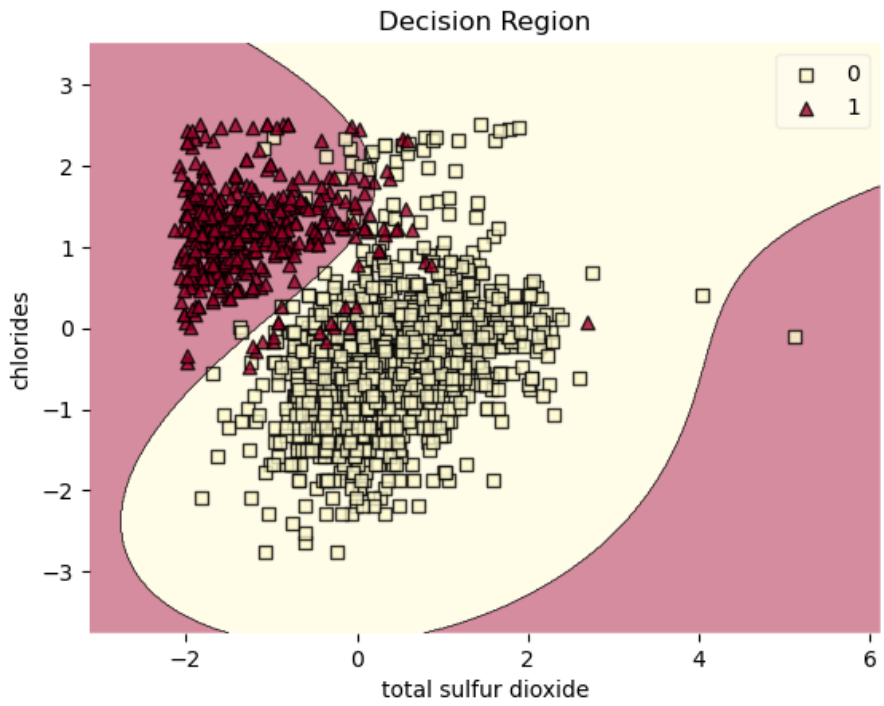


Fig. 12-1 & Fig. 12-2

Model Performance

	Predict White	Predict Red

	Predict White	Predict Red
True White	0.9983	0.0017
True Red	0.0184	0.9816

Acurracy : 99.3846%

Precision : 99.3854%

Recall : 99.3846%

f1 : 99.3832%

Model Complexity

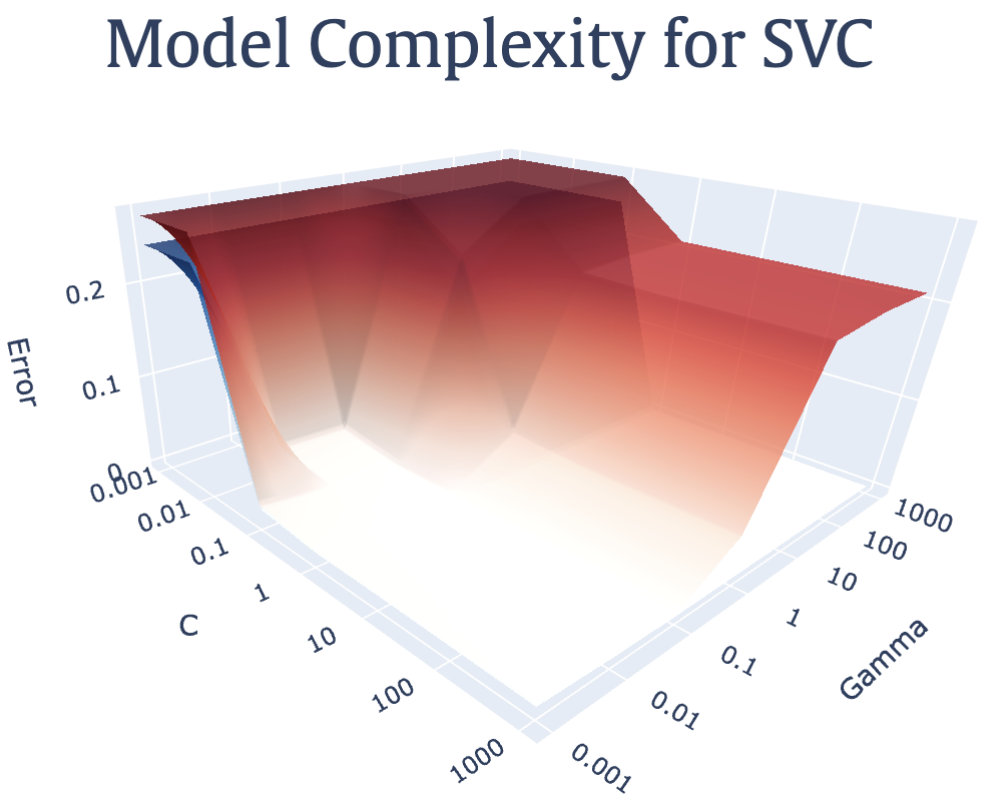


Fig. 12-3

XGBoost

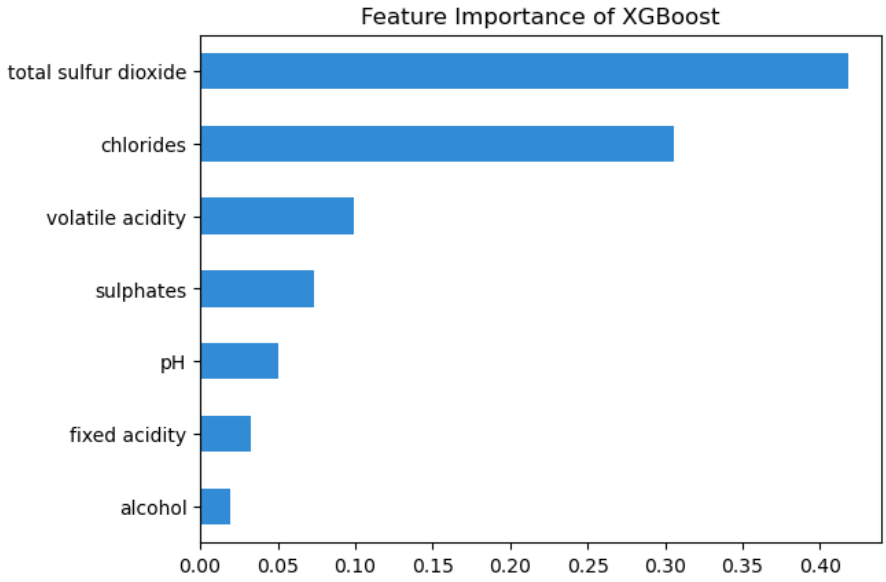
Cross Validation

```
params = {
    'max_depth': [1, 3, 5, 7, 9, 11],
    'n_estimators': [50, 100, 200, 250, 300],
}
```

```
Best params: {'max_depth': 3, 'n_estimators': 250}
Best scores: 0.9890
```

Feature Selection

```
SFS = ['fixed acidity', 'volatile acidity', 'chlorides', 'total sulfur
dioxide', 'pH', 'sulphates']
Lasso = ['fixed acidity', 'volatile acidity', 'chlorides', 'pH',
'sulphates', 'alcohol']

Feature Group = ['fixed acidity', 'volatile acidity', 'chlorides', 'total
sulfur dioxide', 'pH', 'sulphates', 'alcohol']
```



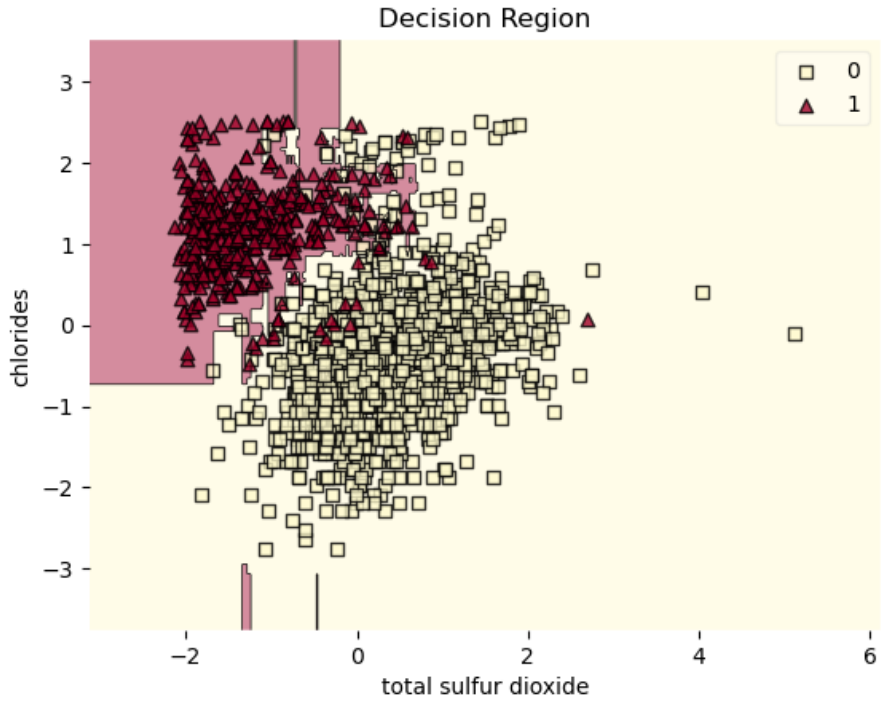


Fig. 13-1 & Fig. 13-2

Model Performance

	Predict White	Predict Red
True White	0.9983	0.0017
True Red	0.0161	0.9839

Acurracy : 99.4462%

Precision : 99.4465%

Recall : 99.4462%

f1 : 99.4451%

Model Complexity

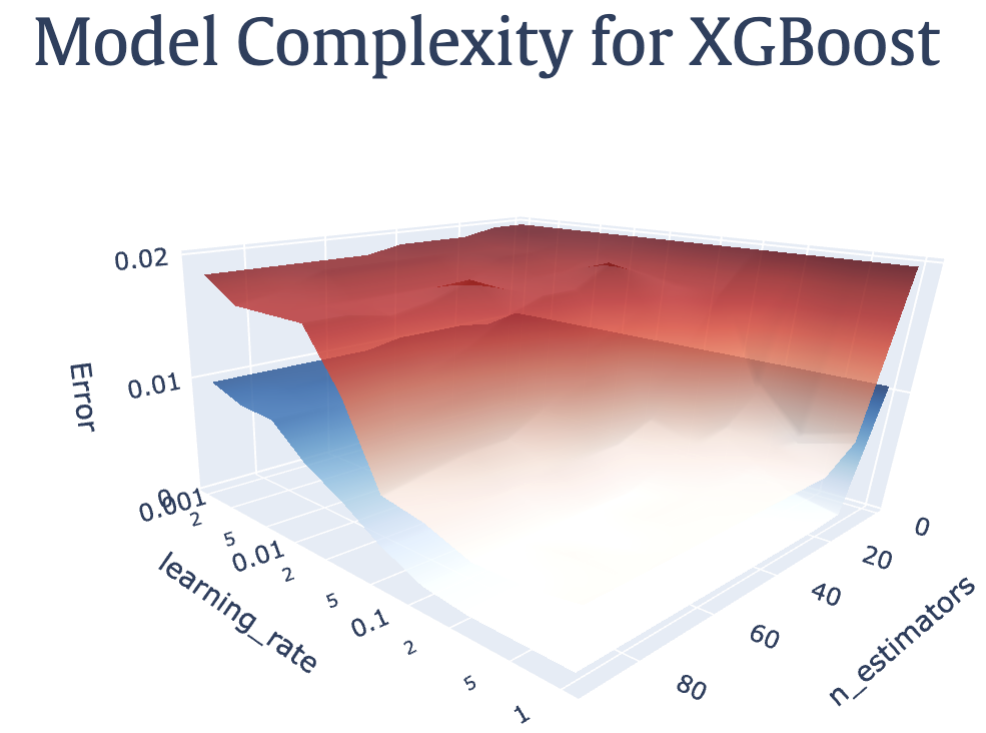


Fig. 13-3

Decision Tree

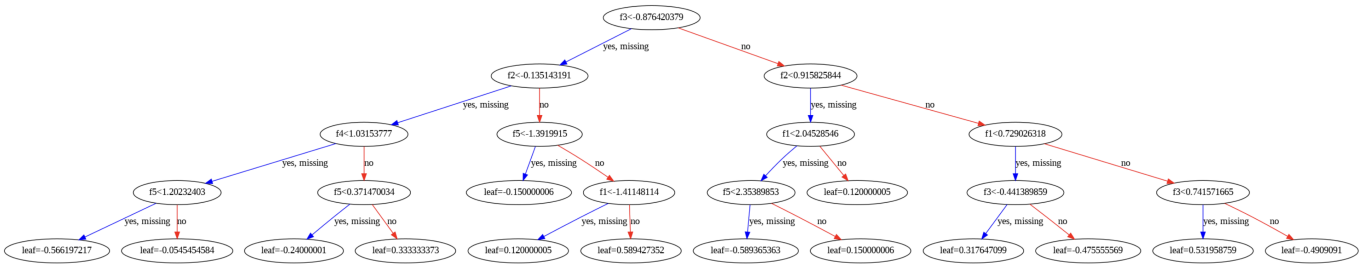


Fig. 13-4

{'fixed acidity'} {0} ; {'volatile acidity'} {1} ; {'chlorides'} {2} ; {'total sulfur dioxide'} {3} ; {'pH'} {4} ; {'sulphates'} {5} ; {'alcohol'} {6}

Wine Quality Classification

參考了 P. Cortez *et al.* 的文章結論顯示，SVC 在預測葡萄酒品質的表現是最好的 (相較於 Multiple Regression 和 Neural Networks)，故本文也以 SVC model 做 wine quality 的學習。文章中是以增加 Error tolerance 的方式使 Classification accuracy 從 43.2% 增加到 89.0%，SVC 中的誤差項通過設置誤差上限 ϵ 來限制，使得預測值和實際值之間的絕對差值小於或等於 ϵ，通過調整 ϵ 的值，我們可以控制 SVC 模型的準確性。較小的 ϵ 值意味著模型對誤差的容忍度更嚴格，這容易導致 Overfitting，而較大的 ϵ 值會導致 Underfitting。簡單來說，SVC 使我們能夠靈活地定義我們的模型可以接受多少誤差，並找到合適的直線或更高維度的超平面來擬合數據 [3], [4] 。有鑒於 Wine Quality 呈非常漂亮的常態分佈 (Fig. 14)，但也代表著將在預測等級低的 3, 4 或 等級高的 8, 9 時，將因為資料數量不夠模型去做訓練的關係而導致預測失準，所以我的方法是用 SMOTE (Synthesized Minority Oversampling Technique, sampling strategy = “auto”) 來增加少數類別樣本，SMOTE 的採樣模式是通過在現有的少數類別樣本之間進行插值，為少數類別生成合成樣本。具體來說，它隨機選擇一個少數群體的樣本，並在特徵空間 (feature space) 中找到它的 k 個最近的鄰居 (k nearest neighbors)，然後隨機選擇這些鄰居中的一個，並在連接原始少數群體樣本和所選鄰居的線段上隨機選擇一個點，生成一個新的合成樣本 [5], [6]

不同於文章中的做法，我利用 Grid Search CV 找出 SVC 對此資料集的最佳 Error tolerence (倒數正比於 SVC 中的 C) 和 gamma 值再比較有無使用 SMOTE 方法來上採樣的結果比較，包括 Confusion Matrix, Accuracy, Precision, Recall, F1 score, ROC curve 和 AUC，最後觀察其 Model Complexity 判斷是否有 Overfitting 的狀況。

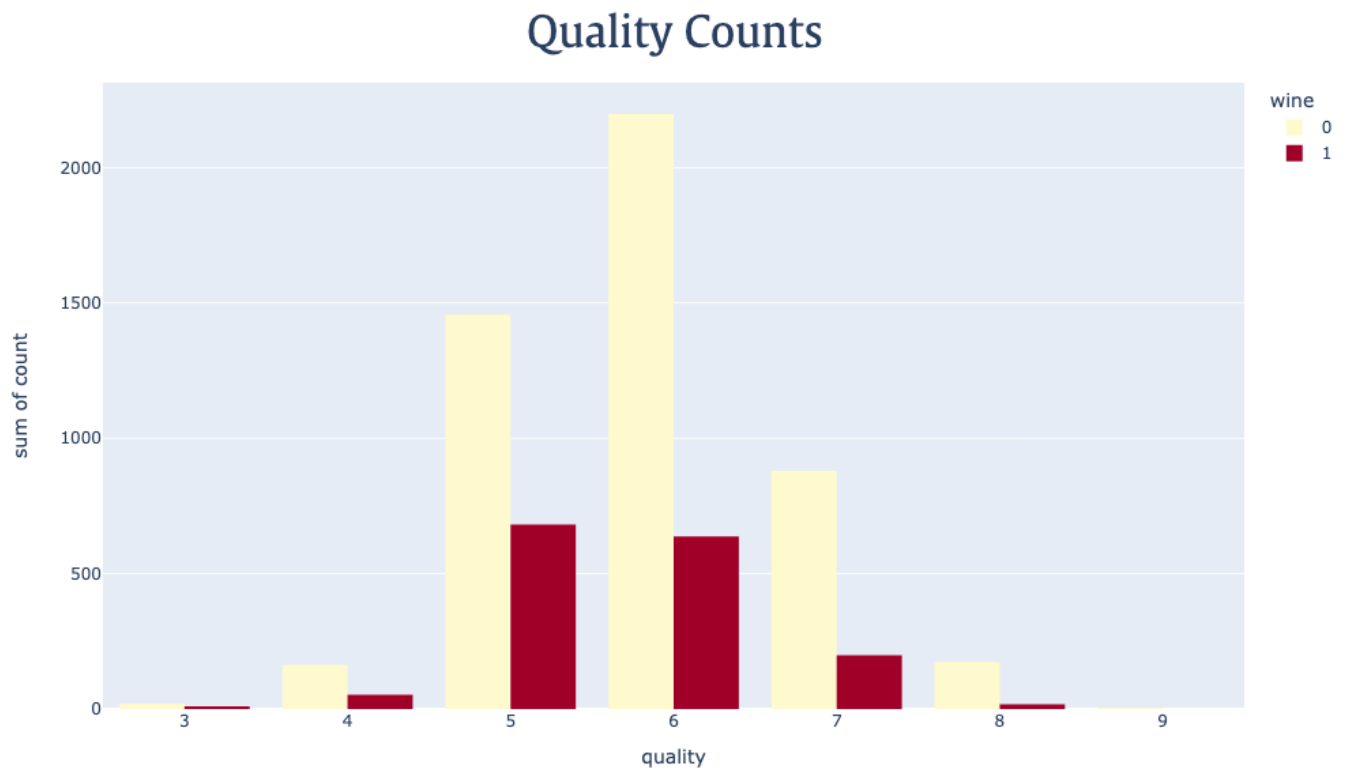


Fig. 14

Support Vector Classifier for Wine Quality

可以從 Table 4 這張簡單的圖表中看到，在 SMOTE upsample 之前 Accuracy 分別只有 52.5% / 61.14%，但在 SMOTE upsample 後 Accuracy 分別提升了 1.57 和 1.28倍 來到 82.39% / 78.22%，且從 Fig. 15-3, 4 , Fig. 16-3, 4 的 Confusion Matrix 可以看到，在 SMOTE 前因為資料集中在 Quality 5 ~ 6，導致 Red wine 和 White wine 的分類幾乎也集中在 Quality 5 ~ 6 的部分，但在 SMOTE 過後，可以看到分類結果呈現了很好的對角線，即 True Positive。

在觀察 Model Complexity 的部分 (Fig. 15-7, 16-7)可以看到 Training error 大致隨著 gamma 和 C 值增加而降低，但紅白酒的 Testing error 都在 gamma 和 C 太大時會有一個上升趨勢，和 Training error 有一個背離的現象，代表有 Overfitting 的狀況。除此之外，在比較 Before / After SMOTE 的參數選擇和 Model Complexity 的部分 (Fig. 15-7, 16-7) 也可以看到 SVC 模型在 C 和 gamma 之間的 Trade-off 行為，我認為十分有趣 ！

Table 4	**Red Wine**		**White**	
	Before SMOTE	After SMOTE	Before SMOTE	After SMOTE
Accuracy	52.50%	82.39%	61.14%	78.22%
Precision	73.24%	81.84%	78.71%	89.88%
Recall	52.50%	82.39%	61.14%	78.22%
F1 score	44.88%	81.87%	56.30%	80.47%

TOP 3 Features	**1**	**2**	**3**
Red wine	“total sulfur dioxide”	“free sulfur dioxide”	“citric acid”
White wine	“total sulfur dioxide”	“free sulfur dioxide”	“residual sugar”

“total sulfur dioxide”

ANOVA

ANOVA - total sulfur dioxide

Cases	Sum of Squares	df	Mean Square	F	p
quality	63.626	5	12.725	24.671	< .001
Residuals	821.671	1593	0.516		

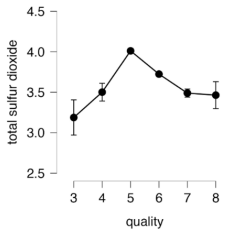
Note. Type III Sum of Squares

Descriptives

Descriptives - total sulfur dioxide

quality	N	Mean	SD	SE	Coefficient of variation
3	10	3.188	0.686	0.217	0.215
4	53	3.501	0.801	0.110	0.229
5	681	4.011	0.773	0.030	0.193
6	638	3.724	0.650	0.026	0.174
7	199	3.489	0.711	0.050	0.204
8	18	3.464	0.705	0.166	0.204

Descriptives plots



“free sulfur dioxide”

ANOVA

ANOVA - free sulfur dioxide

Cases	Sum of Squares	df	Mean Square	F	p
quality	17.406	5	3.481	6.281	< .001
Residuals	882.951	1593	0.554		

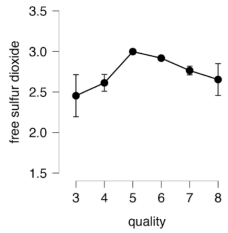
Note. Type III Sum of Squares

Descriptives

Descriptives - free sulfur dioxide

quality	N	Mean	SD	SE	Coefficient of variation
3	10	2.454	0.820	0.259	0.334
4	53	2.613	0.752	0.103	0.288
5	681	2.998	0.748	0.029	0.250
6	638	2.918	0.734	0.029	0.251
7	199	2.765	0.753	0.053	0.272
8	18	2.654	0.831	0.196	0.313

Descriptives plots



“citric acid”

ANOVA

ANOVA - citric acid

Cases	Sum of Squares	df	Mean Square	F	p
quality	1.809	5	0.362	19.135	< .001
Residuals	30.129	1593	0.019		

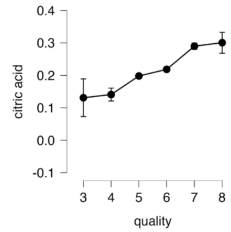
Note. Type III Sum of Squares

Descriptives

Descriptives - citric acid

quality	N	Mean	SD	SE	Coefficient of variation
3	10	0.131	0.183	0.058	1.400
4	53	0.141	0.144	0.020	1.024
5	681	0.198	0.132	0.005	0.666
6	638	0.218	0.142	0.006	0.652
7	199	0.290	0.137	0.010	0.471
8	18	0.301	0.137	0.032	0.456

Descriptives plots



Red

Grid Search CV

```
params = {
        'C': np.logspace(-3, 3, num = 7, base = 10),
        'kernel': ['rbf', 'sigmoid'],
        'gamma': np.logspace(-3, 3, num = 7, base = 10),
    }

Before SMOTE
    Best params: {'C': 10, gamma: 100, kernel: 'rbf'}
    Best scores: 0.6035

After SMOTE
    Best params: {'C': 1000, gamma: 10, kernel: 'rbf'}
    Best scores: 0.8404
```

Feature Importance

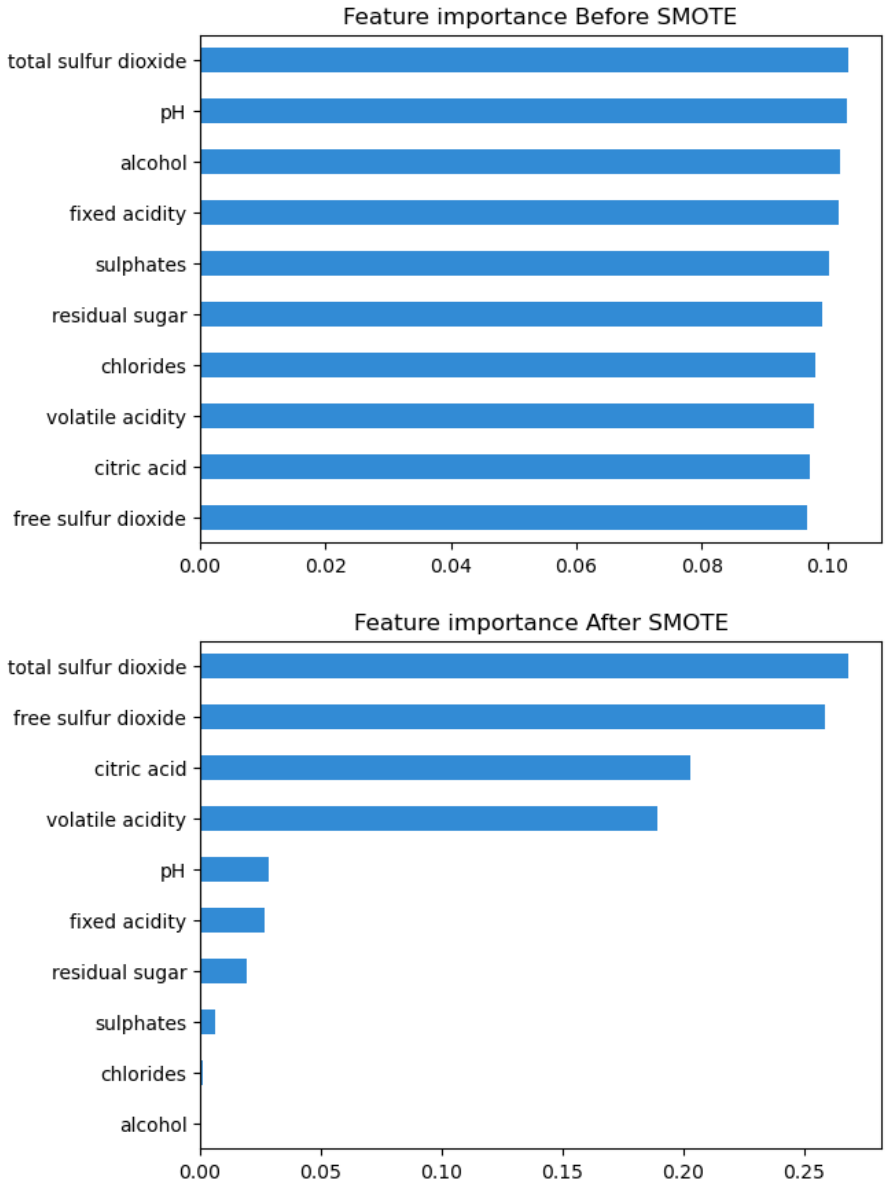
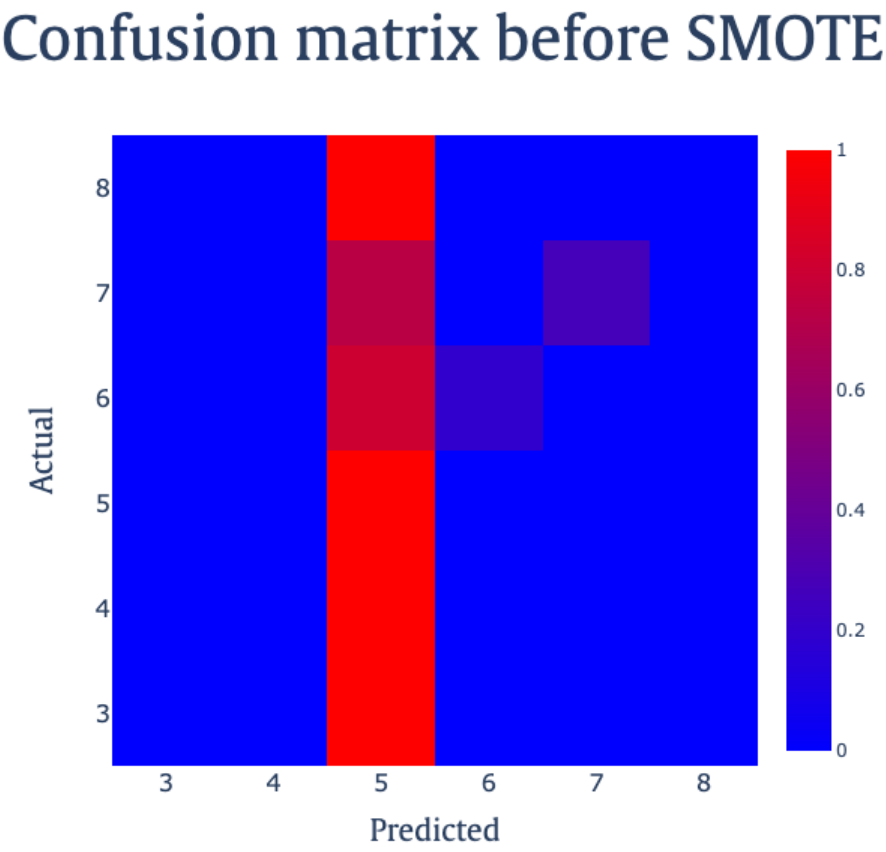


Fig. 15-1 & Fig. 15-2

Model Performance



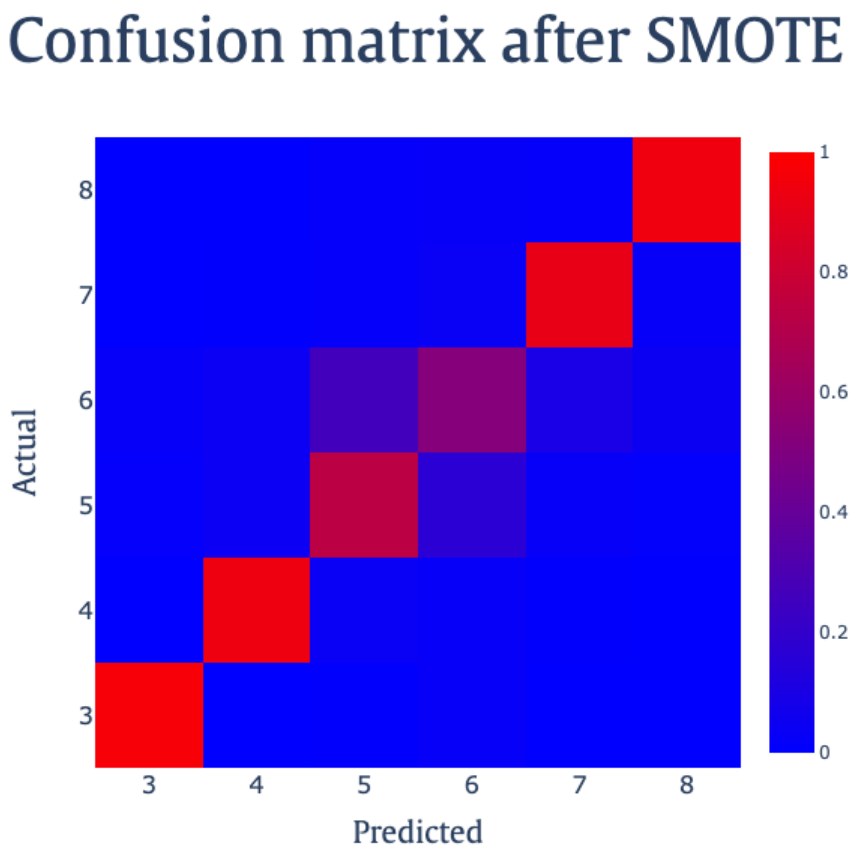


Fig. 15-3 & Fig. 15-4

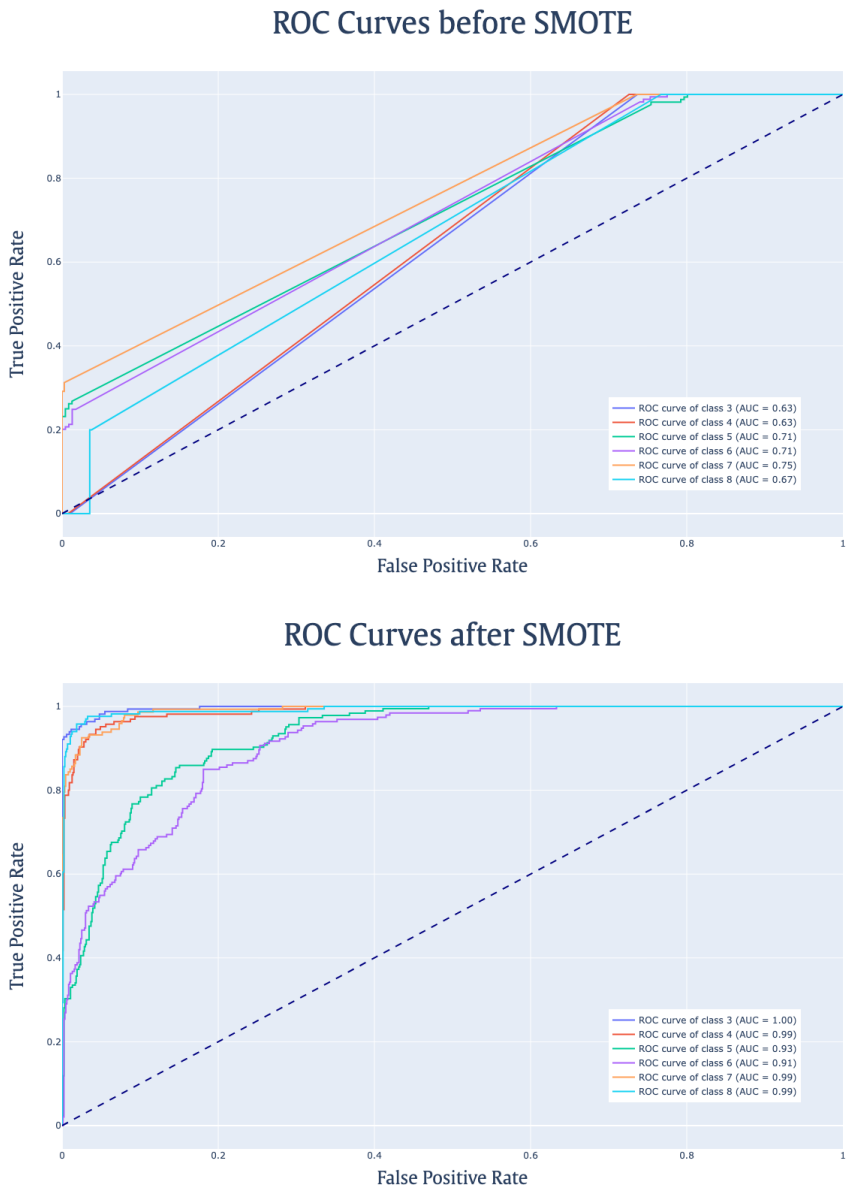


Fig. 15-5 & Fig. 15-6

White

Grid Search CV

```
params = {
        'C': np.logspace(-3, 3, num = 7, base = 10),
        'kernel': ['rbf', 'sigmoid'],
        'gamma': np.logspace(-3, 3, num = 7, base = 10),
    }

Before SMOTE
    Best params: {'C': 1, gamma: 100, kernel: 'rbf'}
    Best scores: 0.6192

After SMOTE
```

```
Best params: {'C': 1000, gamma: 1, kernel: 'rbf'}
Best scores: 0.8794
```

Feature Selection

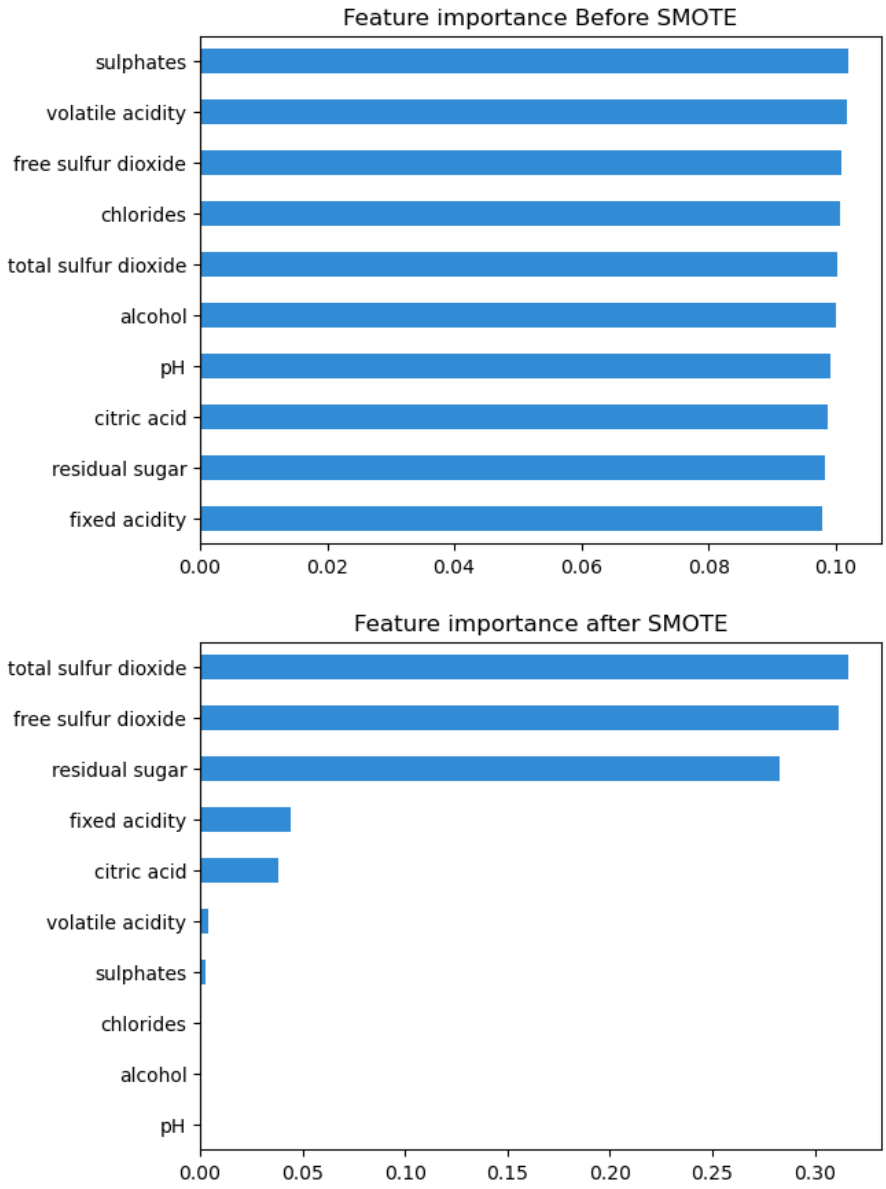
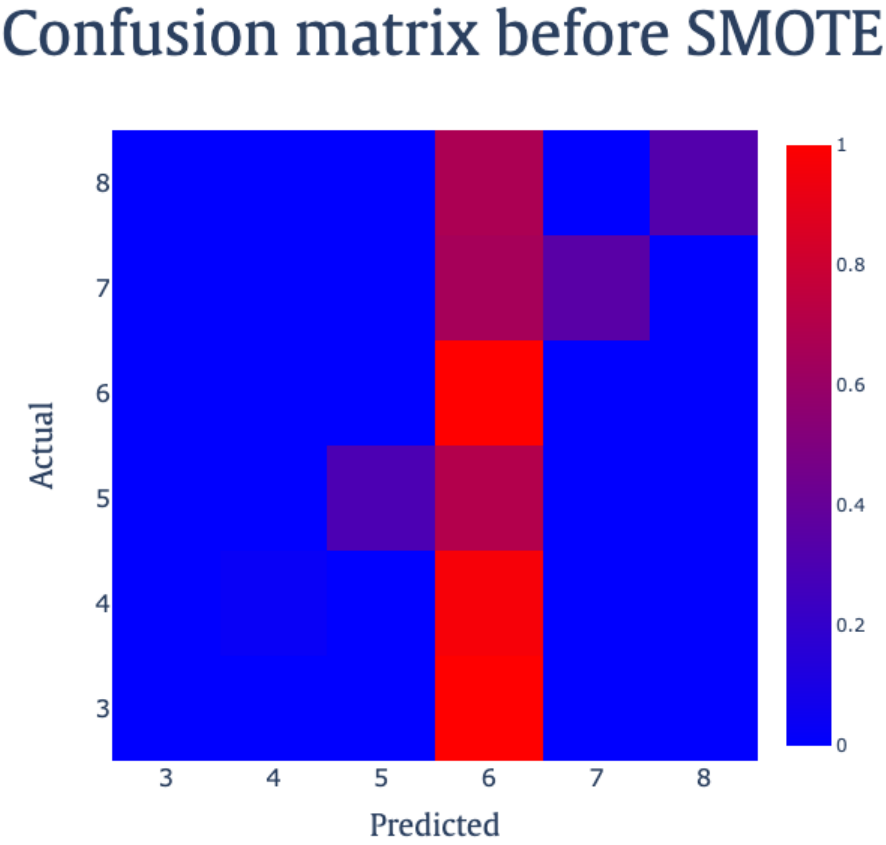
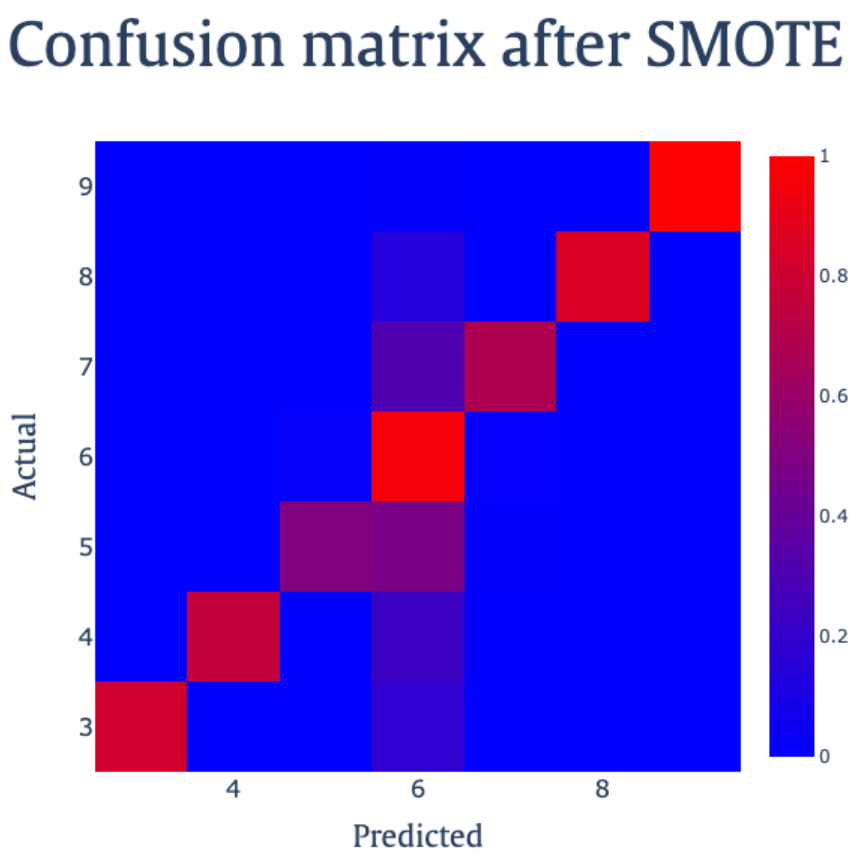


Fig. 16-1 & Fig. 16-2

Model Performance





![newplot]
(https://github.com/scfengv/Wine-Type-and-Quality-Classification/assets/123567363/ffccffc2-210c-40e4-99c0-2e19b42bf70f)

Fig. 16-3 & Fig. 16-4

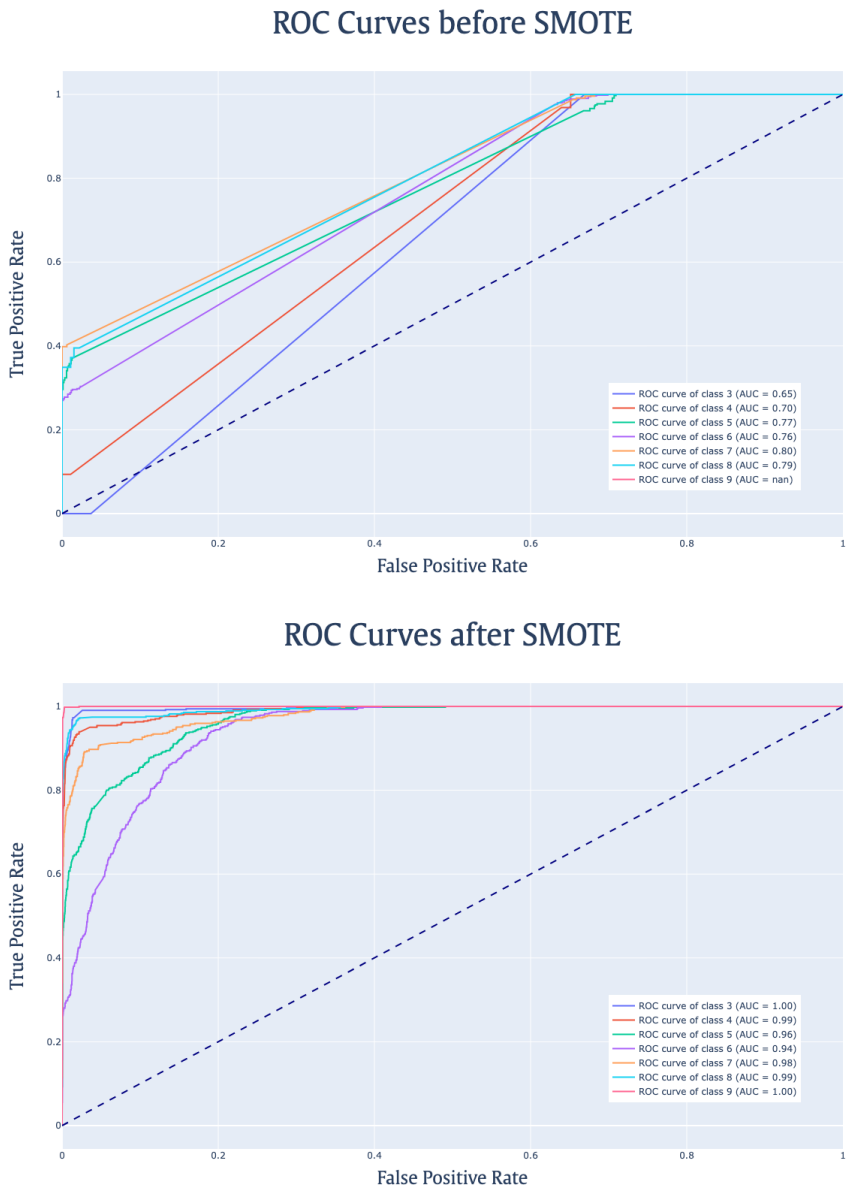


Fig. 16-5 & Fig. 16-6

Results and Discussion

最後，我想針對 P. Cortez *et al.* 的文章做一些討論，從其文章中的 Table 2 可以看到，他用來衡量模型的標準是 Accuracy，但正如我的 Fig. 14 顯示，此資料集的 Quality 是集中在 5~6 這個區間，而作者也有畫出他們的 Confusion matrix，可以看到他們並無做 Over-sampling 等處理，表示其 Accuracy 數值會嚴重因為資料集中的問題而提高，我嘗試復刻他們的實驗（沒有轉換, 沒有 Over-sampling）而得到的 Classification Report 大致如下，雖然因為參數設定不同等原因不會完全一樣，但仍可以作為參考：

Red wine

```
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.00      0.00      0.00        13
           5       0.46      1.00      0.63       164
           6       1.00      0.20      0.33       169
           7       1.00      0.25      0.40        48
           8       0.00      0.00      0.00         5

    accuracy                           0.52       400
   macro avg       0.41      0.24      0.23       400
weighted avg       0.73      0.52      0.45       400

[[  0   0   1   0   0   0]
 [  0   0  13   0   0   0]
 [  0   0 164   0   0   0]
 [  0   0 136  33   0   0]
 [  0   0  36   0  12   0]
 [  0   0   5   0   0   0]]
```

White wine

```
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         7
           4       1.00      0.03      0.06        32
           5       1.00      0.30      0.47       358
           6       0.54      1.00      0.70       544
           7       1.00      0.35      0.52       241
           8       1.00      0.33      0.49        43

    accuracy                           0.61      1225
   macro avg       0.76      0.34      0.37      1225
weighted avg       0.79      0.61      0.57      1225

[[  0   0   0   7   0   0]
 [  0   1   0  31   0   0]
 [  0   0 109 249   0   0]
 [  0   0   0 544   0   0]
 [  0   0   0 156  85   0]
 [  0   0   0  29   0  14]]
```

從上述公式可以將 Precision 和 Recall 理解成，Precision 是計算被分類到此類的所有數據中，真的屬於這個類別的機率，而 Recall 則是所有屬於這個類別的數據中，真的被分到這類的機率。可以在 Red wine 的 Quality = 5 和 White wine 的 Quality = 6 中看到 Low Precision & High Recall 的現象，即表示被分到這個類別的資料分

別有 54% 和 46% 不屬於這個類別，但屬於這個類別的資料都有確實被分類到此處。而在 Red wine 的 Quality = 6, 7 和 White wine 的 Quality = 4, 5, 7, 8 都有看到 High Precision & Low Recall 的現象，表示雖然被分到這個類別的資料確實都屬於此處，但該類別仍有很多資料四散在其他類別 (即前面提到的 Low Precision & High Recall 處)。

基於以上理由我認為這篇論文使用 Accuracy 作為衡量模型的標準是不恰當的，而應該是使用綜合評估 Precision 和 Recall 的 F1 score，以處理不平衡資料的分佈問題，而我前面之所以用 Accuracy 來做是因為我有對資料集做 Over-sampling 來使資料集變為平衡，故使用 Accuracy 可以綜合評估分類的正確性。

Reference:

[1] Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, José Reis, Modeling wine preferences by data mining from physicochemical properties, Decision Support Systems, Volume 47, Issue 4, 2009

[2] James G, Witten D, Hastie T, Tibshirani R. An Introduction to Statistical Learning: With Applications in R. 1st ed. 2013, Corr. 7th printing 2017 edition. Springer; 2013.

[3] An Introduction to Support Vector Regression https://towardsdatascience.com/an-introduction-to-support-vector-regression-svr-a3ebc1672c2

[4] R筆記 – (14)Support Vector Machine/Regression(支持向量機SVM) https://rpubs.com/skydome20/R-Note14-SVM-SVR

[5] 5 SMOTE Techniques for Oversampling your Imbalance Data. https://towardsdatascience.com/5-smote-techniques-for-oversampling-your-imbalance-data-b8155bdbe2b5

[6] SMOTE API https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.SMOTE.html