Rapport final Wine Quality
Jean-Claude Faber

Table des matières

1. Choix du DataSet : Wine Quality

L'évaluation de la qualité du vin dans ce jeux de donnée repose sur des dégustations sensorielles réalisées par des experts. Chaque vin a été noté par au moins trois dégustateurs, puis la médiane de leurs évaluations a été utilisée comme score final. La note varie de 0 (très mauvais) à 10 (excellent). Je vais travailler sur le jeux de vin rouge

Input variables (based on physicochemical tests): min et max des valeurs
1 - fixed acidity (4,6/15,9)
2 - volatile acidity (0,12/1,58)
3 - citric acid (0/1)
4 - residual sugar (0,9/15,5)
5 – chlorides (0,012/0,611)
6 - free sulfur dioxide (1/72)
7 - total sulfur dioxide (6/289)
8 – density (0,99007/1,00369)
9 – pH (2,74/4,01)
10 – sulphates (0,33/2,00)
11 – alcohol (8,4/14,9)
Output variable (based on sensory data):
12 - quality (score between 0 and 10) (3/8)

2. Nettoyage des données, caractérisation et transformation :

Le nombre de valeur est de 1599 lignes et 12 colonnes.
Il n'y a pas de valeur null
Il y a 240 doublons

Après visualisation des doublons et confirmation que se sont bien des doublons, je vais les supprimer pour qu'il ne fausse pas les calculs
Je passe ainsi à 1359 lignes et 12 colonnes

Pour visualiser les données je vais faire un boxplot des variables physico-chimiques.
Les boites à moustache permettent de visualiser les outliers et la dispersion.
La quasi-totalité des variables présentent de nombreuses valeurs extrême.
Les variables particulièrement touchées sont les variables fixed acidity, residual sugar, free sulfur dioxide et total sulfur dioxide.

Les variables ont des échelles très différentes, celles qui ont les plus grandes valeurs vont dominer la distance dans le modèle, rendant les autres variables moins influentes, même si elles sont pertinentes.
Je vais donc appliquer une technique de mise à l'échelle (scaling)

Réalisation d'une heat map pour voir les corrélations avec quality

Influence sur la Qualité Binaire : Corrélations Positives Clés par ordre d'importance
améliorant la qualité : alcohol (+0,48), sulphates (+0,25), citric acid(+0,23), fixed acidity(+0,12).

Corrélations Négatives Clés par ordre d'importance dégradant la qualité:volatile acidity (-0,40), total sulfur dioxide (-0,18), density(-0,18), chlorides (-0,13), free sulfur dioxide (-0,05)

3. Modèle de régression linéaire multiple :

Modèle régression linéaire multiple sélection descendante (backward selection) :
Construction du modèle, utilisation de la bibliothèque stasmodels pour inclure une constante Beta0 et calcul des coefficients.

Analyse de la Variance (ANOVA) et Test de Fisher :
On vérifier si la variabilité de la qualité s'explique réellement par les variables explicatives.
On teste l'hypothèse H0: $\beta 1 = \beta 2 = \cdots = \beta p = 0$ contre l'hypothèse
$H1$: au moins un des paramètres $\beta 1, \beta 2, \ldots, \beta p$ est non nul.
Si la p-value associée à la statistique de Fisher (F) est inférieure à votre seuil (généralement 0,05), on rejet H0, ce qui signifie qu'au moins une variable contribue significativement au modèle.

Pour Prob (F-statistic) on obtient 5,83 x10^-124, ce qui est extrêmement inférieur au seuil de 0,05.
On rejette l'hypothèse nulle H0: $\beta 1 = \beta 2 = \cdots = \beta p = 0$
Au moins une des variables explicatives contribue de manière significative à expliquer la note de qualité

Évaluation de la Qualité et Sélection :
Coefficient de détermination $R2$, il permet d'évaluer la qualité du modèle
Si $R2 = 1$, l'ajustement est parfait.
R-squared=0,364 soit 36,4 % qui s'explique par les régresseurs

Coefficient de détermination ajusté :
Permet de sélectionner les variables qui contribuent de manière significative
Adj.R-squared : 0,359 soit 35,9 %

Analyse de la Significativité des Variables (Test de Student) :
L'examen des p-values permet d'identifier les régresseurs réellement influents (ceux dont la p-value est < 0,05)
Les variables significatives avec leur impact via le coefficient beta sont les suivantes :

Impact positif majeur :
l'alcohol P=0,000 / coefficient Beta 0,3132
les sulphates P=0,000 / coefficient Beta 0,1561

Impact négatif majeur :
la volatile acidity P=0,000 / coefficient Beta -0,2050
les total sulfur dioxide P=0,001 / coefficient Beta -0,0904
le pH P=0,031 / coefficient Beta -0,0711
les chlorides P=0,000 / coefficient Beta -0,0953

Sélection du sous-modèle avec les variables significatives
Le modèle reste significatif avec une Prob (F-statistic) de 3,78 x 10^-128, ce qui est extrêmement inférieur au seuil de 0,05.
On rejette l'hypothèse nulle $H0: \beta 1 = \beta 2 = \cdots = \beta p = 0$
Au moins une des variables explicatives contribue de manière significative à expliquer la note de qualité

Comparaison des Critères AIC/BIC :
Le meilleur modèle au sens du critère AIC est celui pour lequel la valeur AIC est minimale.
Le meilleur modèle au sens du critère BIC est celui pour lequel la valeur BIC est minimale.
AIC : Est passé de 2738 à 2732.
BIC : Est passé de 2800 à 2768.
Ce sous-modèle est donc statistiquement "meilleur"
Le coefficient de détermination R2 est resté stable à 0,359 (35,9%).
Cela signifie que 5 variables inutiles ont été supprimées sans perdre de pouvoir explicatif.

Conclusion pour la régression linéaire multiple optimisée :
Malgré l'optimisation, plusieurs éléments indiquent que la Régression Linéaire Multiple (OLS) n'est pas l'outil final idéal pour traiter la "qualité" d'un vin car :
Faiblesse du R2 qui est seulement de 36%
La Normalité des résidus : Tests Omnibus et Jarque-Bera tests qui vérifient si les erreurs suivent une distribution gaussienne (loi normale).
Omnibus (Prob: 0.000) et Jarque-Bera (Prob: 1.92e-08), dans les deux cas, la "Probabilité" (p-value) est extrêmement faible (bien inférieure à 0,05).
On rejette l'hypothèse nulle de normalité, les erreurs du modèle ne sont pas normalement distribuées

Confirmation graphique du rejet de l'hypothèse de normalité des résidus :

L'histogramme montre une distribution des erreurs qui n'est pas parfaitement symétrique.
Les points sur le QQ-Plot ne suivent pas parfaitement la ligne droite rouge, surtout aux extrémités .
Résidus vs Qualité Prédites, les résidus ne sont pas distribués de manière aléatoire et uniforme (hétéroscédasticité)
Qualité Réelle vs Prédite, le modèle a un faible pouvoir de discrimination. Il ne peut pas prédire la "catégorie" exacte du vin.

4. Métrique de comparaison utilisée pour les différents modèles :

Source : https://scikit-learn.org/stable/modules/model_evaluation.html#accuracy-score

Accuracy (exactitude) : elle représente la proportion totale de prédictions correctes (vrais bons vins et vrais mauvais vins) parmi toutes les bouteilles testées.

$$\text{Accuracy} = \frac{\text{nombre de prédiction correctes}}{\text{nombre total de prédictions}}$$

source : https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html

F1-score : F1 peut être interprété comme une moyenne harmonique de la précision et rappel, un score en F1 atteint sa meilleure valeur à 1 et le pire score à 0

$$\text{F1-score} = 2 \times \frac{\text{Précision x Rappel}}{\text{Précision + rappel}}$$

Source https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_score.html

Précision : La précision mesure la fiabilité des prédictions positives du modèle.

$$\text{Précision} = \frac{\text{Vrai positif}}{\text{vrai positif + Faux positif}}$$

source : https://scikit-learn.org/stable/modules/generated/sklearn.metrics.recall_score.html

Rappel : la capacité du classificateur à trouver tous les échantillons positifs

$$\text{Rappel} = \frac{\text{Vrai positif}}{\text{vrai positif + Faux négatif}}$$

source : https://scikit-learn.org/stable/modules/model_evaluation.html#roc-metrics

Courbe ROC : Elle est créée la fraction de vrais positifs parmi les positifs (TPR = vrai positif taux) vs. la fraction de faux positifs sur les négatifs (FPR = faux taux positif), à différents seuils.

AUC : représente la probabilité que le modèle, s'il reçoit un exemple positif et un exemple négatif choisis aléatoirement, classe l'exemple positif plus haut que l'exemple négatif

5. Régression logistique :

Transformation en binaire de la variable quality pour prédire le succès ou l'échec ici la bonne qualité ou mauvaise qualité du vin. Car pour la régression logistique on se place toujours dans le cadre de la classification binaire.

Distribution de quality_binary :
Le jeu de donnée est relativement équilibré 53 % de classe 1 et 47 % de classe 0.

Pré-traitement et séparation des données :
Séparation des données : 80% Entraînement, 20% Test
Modèle aux 6 variables significatives identifiées lors de la phase de la régression linéaire multiple
Entraînement uniquement sur l'ensemble d'entraînement
Application de la même transformation à l'ensemble de test
Données standardisées dans le DataFrame
Entraînement du Modèle Initialisation du modèle de Régression Logistique
Entraînement du modèle sur les données standardisées

Interprétation des Coefficients_poids (w) :

```
Biais (Intercept - b) : 0.2287
Coefficients (poids w) :
alcohol                1.0188
sulphates              0.4711
pH                    -0.1152
chlorides             -0.2537
total sulfur dioxide  -0.4096
volatile acidity      -0.4453
dtype: float64
```

Facteur influençant positivement la qualité du vin :
Alcool (1.0188) : Est la variable la plus influente.
Sulphates (0.4711) : Ils contribuent positivement à la qualité

Facteur de dégradation de la qualité du vin :
Volatile Acidity (-0.4453) et total sulfur dioxide (-0,4096) sont les facteurs principaux qui diminuent fortement la qualité du vin.

Interprétation des coefficients (Odds Ratio)

Odds : Les "chances" représentent la probabilité qu'un événement se produise divisée par la probabilité qu'il ne se produise pas

Odds Ratio : C'est le rapport des chances entre deux groupes ici d'avoir un bon vin ou un vin mauvais

OR = 1 : Aucune association entre les deux groupes.

OR > 1 : L'événement est plus probable dans le premier groupe.

OR < 1 : L'événement est moins probable dans le premier groupe.

```
 Analyse des Odds Ratios (Impact)
alcohol                2.77
sulphates              1.60
pH                     0.89
chlorides              0.78
total sulfur dioxide   0.66
volatile acidity       0.64
dtype: float64
```

Chaque augmentation standardisée de l'alcool multiplie les chances que le vin soit classé comme bon.

Évaluation des résultats :

L'évaluation finale sur l'ensemble de test valide la pertinence du modèle de régression logistique avec une précision globale de 73,16 % et un F1-score de 0,7439

Rapport de classification :

Le support indique que j'ai 128 vins "Mauvais" (0) et 144 vins "Bons" (1) dans l'échantillon de test.

Le Recall est la métrique qui mesure la capacité d'un modèle à débusquer tous les individus d'une classe donnée.

Précision = Vrais positifs/(Vrai Positifs+Faux positifs)

Pour les "Bons Vins" (Classe 1) : la précision est de 0.75, cela veut dire que lorsque le modèle prédit qu'un vin est bon, il a raison dans 75 % des cas. Le modèle parvient à détecter 74 % de tous les bons vins.

Pour les "Mauvais Vins" (Classe 0) Le modèle est légèrement moins précis il a raison dans 71 % des cas. Le modèle parvient à détecter 73 % de tous les mauvais vins.

Courbe ROC :

Le score de 0.81 correspond à l'Aire Sous la Courbe (AUC) il donne la capacité globale du modèle à distinguer les bons vins des mauvais, le score de 0.81 indique que dans 81 % des cas, le modèle classera correctement un vin de qualité au-dessus d'un vin médiocre.

Importance des variables (coefficients w) :

Cela confirme que l'alcohol est la variable dominante pour que le vin soit jugé bon à l'opposé l'acidité volatile tend a donner un vin de mauvaise qualité

Distribution des probabilités prédites selon la qualité réelle :

On observe deux populations distinctes : la majorité des vins médiocres reçoivent une probabilité faible, tandis que les vins de qualité reçoivent une probabilité élevée. Le faible chevauchement entre les deux groupes illustre visuellement que le modèle sépare efficacement les classes tout en identifiant clairement les échantillons ambigus situés autour du seuil de 0.5.

6. SVM à marge souple :

La marge dure ne peut pas fonctionner pour le jeu de donnée du vin car les classes se chevauchent. Transformation en binaire de la variable quality pour prédire le succès ou l'échec ici la bonne qualité ou mauvaise qualité du vin. Car **yi appartient à l'intervale {1,-1}**, On transforme le 0 (mauvais) en -1 pour coller à la formule mathématique.

Division Entraînement / Test

Paramètre de la marge souple :

$$\min \|w\|^2 + C \sum_i \xi_i$$

Utilisation de Linear SVC avec C=1 qui correspond à la marge souple

Référence :

https://www.geeksforgeeks.org/machine-learning/hinge-loss-relationship-with-support-vector-machines/

Relation entre la perte de charnière et la SVM :

SGDClassifier (**loss='hinge'**) configure une SVM linéaire à l'aide de la fonction de perte de charnière, tout comme les SVM traditionnelles.

max_iter=1000 garantit suffisamment d'étapes d'apprentissage pour que l'optimiseur puisse potentiellement converger vers une bonne solution

Résultat obtenus :

Le biais, décalage par rapport à l'origine est de b=0,1461 hyperplan passe presque par le centre du nuage de points.

Selon le cours il faut trouver min $\|w\|^2$, le vecteur w est la normale à l'hyperplan la valeur obtenue est 1,1735 c'est la valeur que l'algorithme a réussi à atteindre tout en classant correctement les vins. Ce qui donne une largeur de la marge de 1,7042 ce qui très large car les valeur son normalisées.

Comparaison des performances obtenues :

Métrique	**Régression Logistique**	**SVMP (marge souple)**	**Amélioration**
Précision (Accuracy)	0,7316	0,7647	+4,3 %
F1-Score	0,7439	0,7647	+2,7 %

Le SVM à marge souple est plus performant car minimise simultanément deux termes : $\|w^2\|$ (pour la marge) et $\Sigma\xi i$ (pour les erreurs).

Amélioration de la Précision Globale (Matrice de Confusion) :

- Régression Logistique : 93 + 106 = 199 bonnes prédictions.
- SVM à marge souple : 104 + 104 = 208 bonnes prédictions.

Capacité de Séparation (Courbe ROC) :

- Régression Logistique : AUC = 0.81
- SVM à marge souple : AUC = 0.85

La fonction de décision $w^T x + b$ du SVM est un prédicteur plus robuste pour le jeux de données.

Stabilité des Variables (Vecteur w) :

- L'Alcool reste le moteur principal de qualité (poids positif fort).
- L'Acidité Volatile et le Soufre Total restent les freins majeurs (poids négatifs).

7. Kernel trick :

Principe : transposer les données dans un autre espace (en général de plus grande dimension), appelé espace de redescription, dans lequel elles sont linéairement séparables (ou presque) et ensuite appliquer l'algorithme SVM sur les données transposées.

Etapes :

La binarisation et le remplacement par {-1, 1} pour la qualité du vin, les SVM sont conçus pour la classification binaire.

Division 20% des données totales pour l'ensemble de Test, 80% restants sont alloués à l'ensemble d'Entraînement

La Standardisation remet toutes les caractéristiques sur une échelle commune, elle permet au noyau de mesurer la "similarité" de manière équitable.

Création du modèle avec le noyau RBF, C=1 correspond à la marge souple kernel='rbf' implémente l'astuce du noyau.

Entraînement l'algorithme cherche à maximiser les coefficients αi (multiplicateurs de Lagrange) du problème dual :

$$\max \sum_i \alpha_i - \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

l'algorithme utilise la fonction RBF pour mesurer la similarité entre les vins dans un espace de Hilbert de dimension infinie.

Il doit respecter : $\text{t.q. } \alpha_i \geq 0 \text{ et } \sum_i \alpha_i y_i = 0$

Prédiction : L'objectif de la prédiction est de générer la variable y_pred_rbf, qui contient les étiquettes prédites par l'IA, ces prédictions seront comparées aux vraies étiquettes (y_test).

Comparaison des performances obtenues :

Métrique	**SVMP (marge souple)**	**SVM kernel Trick**	**Amélioration**
Précision (Accuracy)	0,7647	0,7757	+1,4 %
F1-Score	0,7647	0,7875	+2,9 %
Précision mauvais (-1)	0,76	0,80	+5,0 %
Précision bon (1)	0,77	0,75	-2,7 %
Recall mauvais (-1)	0,77	0,73	-5,5 %
Recall bon (1)	0,76	0,82	+7,3 %

Conclusion :
L'astuce du noyau permet au modèle d'être plus performant par rapport au SVMP (marge souple). Cela indique que la séparation entre les "bons" et les "mauvais" vins n'est pas parfaitement droite, mais présente des courbures que seul le SVM kernel Trick peut capturer en projetant les données dans un espace de plus grande dimension.
On obtient une meilleure détection des "Bons Vins" (Rappel : +7,3 %), le modèle non linéaire est beaucoup plus efficace pour dissocié les bouteilles de qualité.
Cependant le modèle est devenu moins sévère. Il rate environ 5,5 % de mauvais vins par rapport au modèle linéaire en les classant par erreur comme "bons". Ce qui a pour conséquence de faire diminuer la précision des bons vins.

8. Perceptron multicouches :

Etapes :

Binarisation : passage en binaire de la qualité car le réseau de neurone apprend en minimisant l'erreur entre sa prédiction $\hat{y}$ et la cible y* , la formule de l'ntropie Croisée est :

$L = -[y^* \log(\hat{y}) + (1-y^*)\log(1-\hat{y})]$

Cette formule ne fonctionne que si y* vaut 0 ou 1.

Division 20% des données totales pour l'ensemble de Test, 80% restants sont alloués à l'ensemble d'Entraînement.

Standardisation : permet que toutes les caractéristiques chimiques du vin contribuent équitablement au calcul du gradient.

Encodage : pour calculer l'erreur (le "Loss"), le réseau compare son vecteur de probabilités avec le vecteur One-hot c'est pour cette raison que les étiquettes de qualité sont transformée en vecteurs binaires de dimension 2, mauvais (0) donne le vecteur [1,0] et bon (1) donne le vecteur [0,1]

Architecture du réseau de neurones : 16 neurones pour être légèrement supérieur aux 11 variables du vin, activation de la fonction ReLU Rectified Linear Unit pour la non-linéarité

Dropout : avec un taux de 20 % a été insérée après la couche cachée. Pendant chaque étape de l'entraînement, le Dropout va "désactiver" aléatoirement 20 % des neurones de la couche précédente, à chaque passage (itération), ce ne sont pas les mêmes neurones qui sont éteints.
Le réseau doit apprendre à prédire si le vin est bon en utilisant seulement 80 % de ses "capacités" à un instant T, cette technique améliore ainsi la précision du modèle sur des données de test inédites."

Couche de sortie : 2 neurones car la couche de sortie doit avoir autant de neurones qu'il y a de catégories à prédire, ici nous avons deux catégorie vin bon (0) ou mauvais (-1).La fonction Softmax transforme les scores en probabilités.

Compilation et entraînement : La fonction de perte : 'categorical_crossentropy', mesure l'erreur entre la prédiction du modèle et la réalité.
L'optimiseur : SGD (Descente de Gradient Stochastique) modifie les poids pour réduire l'erreur.

Le pas d’apprentissage est de 0,05 (learning_rate=0.05)

- si le pas est trop grand : le modèle risque de sauter par-dessus le minimum et de ne jamais stabiliser.
- si le pas est trop petit : l'entraînement sera extrêmement lent

l'Accuracy est le pourcentage de bonnes réponses (vins bien classés) par rapport au total, metrics=['accuracy']

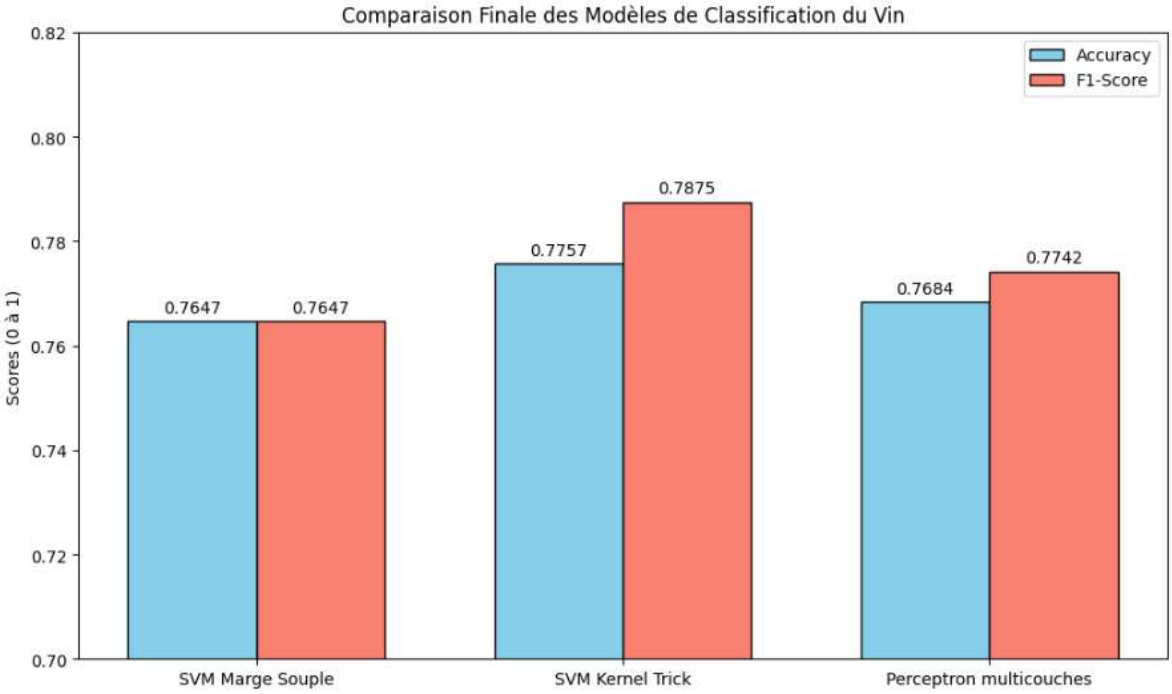
Apprentissage par rétropropagation du gradient :

- Passe Avant (Forward Pass) : Le vin entre dans le réseau, traverse les 16 neurones ReLU, puis la sortie Softmax donne une prédiction
- Calcul de l'Erreur : On compare avec la réalité (One-Hot)
- Rétropropagation (Backward Pass) : On repart de la sortie vers l'entrée

En 60 époques, le modèle va affiner 60 fois les données (epochs=60)
Au lieu de donner les 1359 vins d'un coup, on les donne par petits groupes de 32. (batch_size=32)
Pour afficher les résultats à chaque époque. (verbose=1)

9. Conclusion :

Métrique	SVMP (marge souple)	SVM kernel Trick	Perceptron multicouches
Précision (Accuracy)	0,7647	**0,7757**	0,7684
F1-Score	0,7647	**0,7875**	0,7742
Précision mauvais (-1)	0,76	**0,80**	0,78
Précision bon (1)	0,77	0,75	0,76
Recall mauvais (-1)	0,77	0,73	0,75
Recall bon (1)	0,76	**0,82**	0,79



Le SVM kernel Trick RBF reste le meilleur modèle pour ce problème. Il obtient la meilleure Accuracy (0,7757) et le meilleur F1-score(0,7875).
L'étude montre que la classification de la qualité du vin rouge est un **problème non-linéaire**, car les modèles complexes (SVM kernel Trick et Perceptron multicouches) surpassent le modèle SVM marge Souple.