# Zweck und Erfolgskriterien

## Zweck des Projektes

Eine Lebensmittelkette möchte die Qualität ihrer Weine verbessern, um ihren Kunden ein besseres Einkaufserlebnis zu bieten.

Primärauftrag: Dazu soll ein maschinelles Lernmodell entwickelt werden, das aus chemischen und physikalischen messbaren Merkmalen die Qualität der Weißweine bestimmt.

Bonusauftrag: Dem Auftraggeber ist es wichtig gute Weißweine richtig vorhersagen zu können. Dabei sollen auch möglichst keine schlechten Weine als von hoher Qualität / teuer angeboten, weil sonst wichtige enttäuscht werden können.

## Angewandte Methode

Klassifikation.

Zielspalte ist die Spalte „quality“ mit sieben Qualitätsstufen (von 3 bis 9)

## Scoring

Es ist ein Modell zu entwickeln, das mit einem score von mindestens 80 % gute Weißweine von anderen Weißweinen unterscheiden kann.

Dafür bietet sich der Score „precision“ an, der definiert ist als TP/(TP+FP) und sicherstellt, dass die Anzahl der FP geringer bleibt.

Zusätzlich wäre es wichtig, das das Modell schlechte Weine gut vorhersagt damit sie nicht in den Verkauf kommen.

# Daten Analyse

Die Daten haben:

* 4898 Zeilen und 12 Spalten
* Datentyp: float64 bis auf die Zielspalte ‚quality‘, die int64 ist.
* Datensatzgröße als csv-Datei: 259 kB

1. Fixed Acidity (fester Säuregehalt)

2. Volatile Acidity (flüchtiger Säuregehalt)

3. Citric Acid (Zitronensäure)

4. Residual Sugar (Restzucker)

5. Chlorides (Chloride)

6. Free Sulfur Dioxide (freies Schwefeldioxid)

7. Total Sulfur Dioxide (gesamtes Schwefeldioxid)

8. Density (Dichte)

9. pH-Wert

10. Sulphates (Sulfite)

11. Alcohol (Alkoholgehalt)

12. Quality (Qualität)

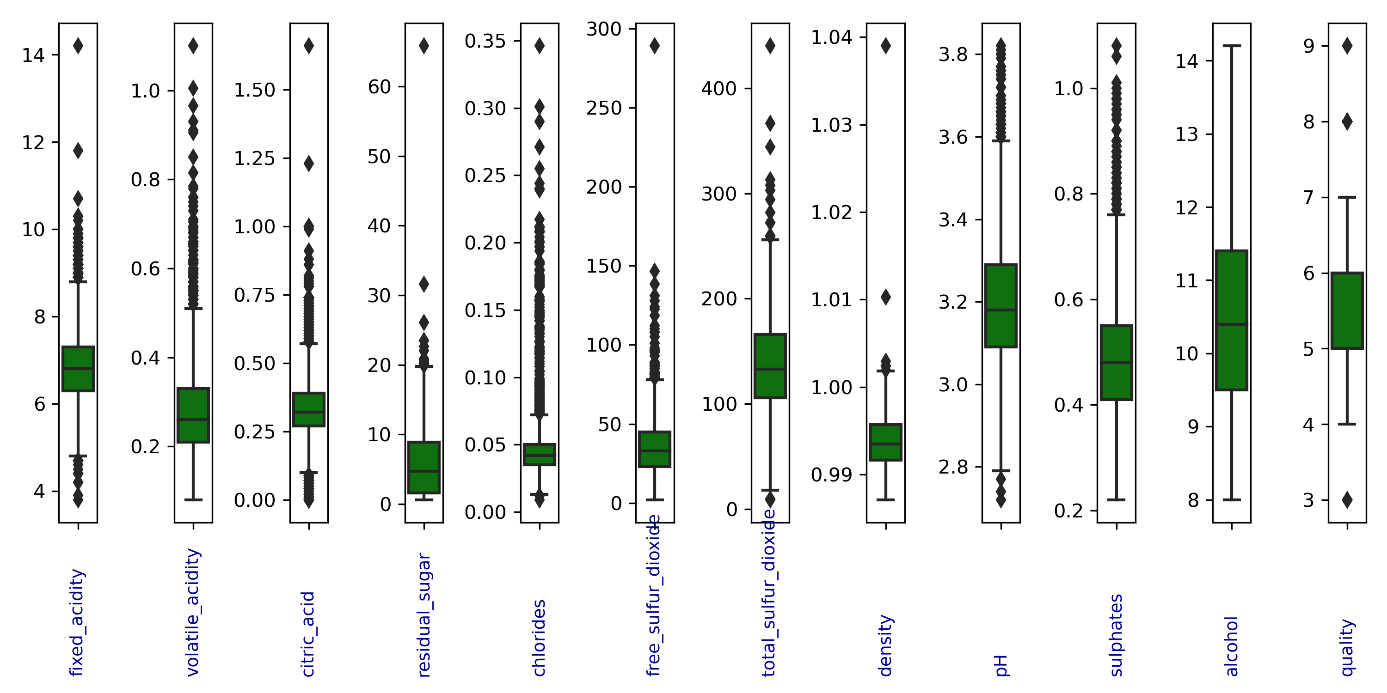
Wir untersuchen die Daten mit verschiedenen Methoden und stellen folgendes fest:

a) es gibt keine fehlenden Werte (Null Werte)

b) es gibt Ausreißer.

Die Überprüfung des Datensatzes hat ergeben, dass es einige sehr starke Ausreißer gibt

Es wurde entschieden die extremsten dieser Ausreißer zu entfernen nach der folgenden Regel: Es werden die Werte entfernt, die größer als das Dreifache des oberen Quartils sind (> 3\*Q3)



c) es gibt Korrelationen zwischen mehreren Datenspalten

Sehr hoch für:

density vs. residual sugar --> 0.82

alcohol vs. density --> -0.76

Hoch für:

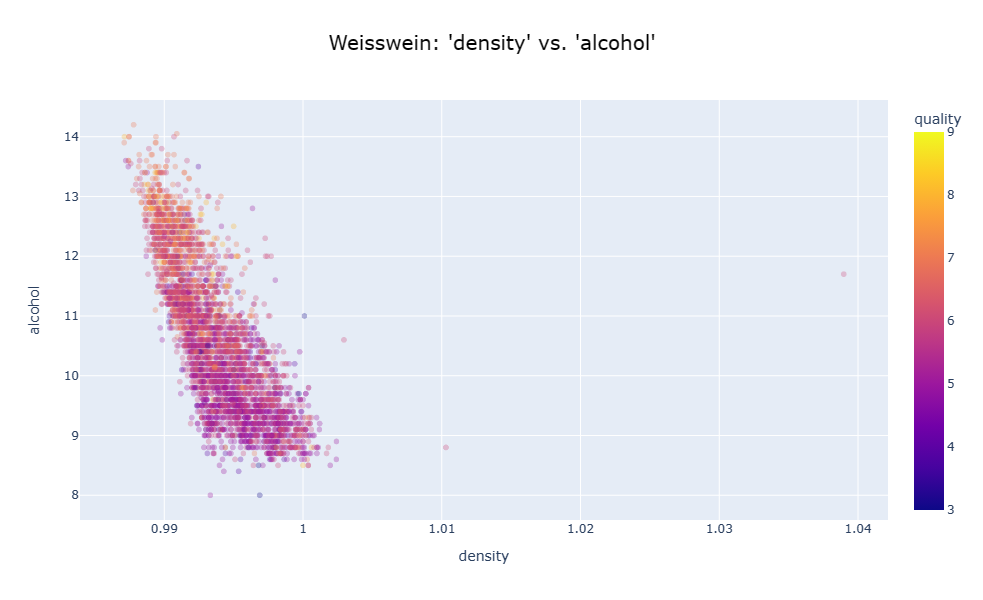
total\_sulfur\_dioxide vs. free\_sulfur\_dioxide --> 0.62

total\_sulfur\_dioxide vs. density --> 0.54

Wichtig:

Alcohol vs. quality (Zielspalte) 🡪 0.46

Beispiel:



d) Es wurden Duplikate gefunden

Hintergrund: sind vermutlich verschiedene Weine bei denen im vorliegenden Datensatz einige Spalten weggelassen wurden, daher könnte man sie löschen oder auch behalten. Es wurde entschieden sie zu löschen.

…

# Datenvorbereitung

1. Die Ausreißer wurden entfernt (Begründung wie in der Daten Analyse)
2. Duplikate wurden entfernt
3. Scaler: Es wurde der StandardScaler benutzt, bei Algorithmen die einen Scaler benötigen.
4. Es wurde für die weitere Analyse eine PCA durchgeführt, mit dem Ergebnis der Spaltenreduktion auf 9 Hauptkomponenten mit >97% "kumulative erklärte Varianz" erreicht wird.  
   Begründung: Es gibt mehrere starke Korrelationen zwischen den Spalten, wodurch eine Reduzierung der Spaltenanzahl ohne relevante Verluste möglich wird. Dadurch können einige Algorithmen, die mit vielen Spalten schlechter umgehen können, zu besseren Ergebnissen kommen.
5. Für das weitere Vorgehen werden mehrere Algorithmen auf Eignung geprüft. Hierfür kann der Datensatz mit 7 Qualitätsstufen verwendet werden, oder es können diese sieben zu einer geringereren zusammengefasst werden. Um einheitlich bei den Algorithmus-prüfungen vorzugehen wurde ein Datensatz mit drei Zielspalten erstellt. Diese wurden wie folgt definiert: „schlecht“ für [3,4], „mittel“ für [5,6] und „gut“ für [7,8,9].  
   Falls zwei Qualitätsstufen geprüft werden dann ist die Aufteilung: „schlecht“ für [3,4,5] und „gut“ für [6,7,8,9]  
     
   Warum ist die Verwendung von allen Kategorien problematisch?  
   - Die Verteilung über die sieben Qualitätsstufen ist nicht gleichmäßig.

- Wir haben zu wenige Daten in einigen Qualitätsstufen. Wir haben z.B. nur 5 Weine mit Stufe 9. Qualitätsstufen mit wenigen Daten können nur sehr schlecht vorhergesagt werden.

# Algorithmen auswählen

Einige der Algothmen die geprüft werden sind:

* SVM  
  Begründung: wir haben nur wenige (11) Spalten, nach PCA nur (9) Spalten.  
  SVM kann sehr gut mit Overfitting und Ausreißern umgehen.
* Decision Tree  
  Begründung: Einfach zu verstehen und interpretieren. Der Kunde kann Entscheidungsbäume schnell verstehen. Kann mit kategoriellen und nicht kategoriellen Werten gut umgehen.
* Logistic Regression  
  Begründung: Effizienz, geringer Speicherbedarf, Interpretierbarkeit
* Neural Networks  
  Begründung: Kann mit komplexen Daten gut umgehen. Leistungsfähiger Algorithmus, der aber auch Nachteile hat, z.B. bei wenigen Daten.  
  Kann theoretisch jede Funktion approximieren aber es ist schwierig die richtigen Hyperparameter zu wählen

Nach einer ersten Prüfung mehrerer Algorithmen entscheiden wir und für eine geeignete Auswahl und legen weitere gemeinsame Kriterien für die Beurteilung und den Vergleich der Scores fest.

Nach einer kurzen Prüfung mit z.B. dem Decision Tree wurde festgestellt, dass sieben Qualitäts-Kategorien zu viel sind. Der CV-scores des Decision Tree ist nach einem Grid search bei 0.52 und der Mean Squared Error ist bei 0.72 . Ein MSE von unter 1 ist nicht schlecht, weil es heißt, dass man in einer benachbarten Ausprägung landet. Aber die Präzision und der CV-score liegen bei einem Maximalwert von 52%. Dieser Wert ist weit unter dem geforderten Wert von 80%. Auch ist aus der Confusion Matrix zu erkennen, dass die Vorhersagen in den niedrigsten und höchsten Ausprägungen nicht gut genug ist. Das folgt aus den niedrigen Anzahl von Weinen in diesen Klassen.

Aus diesem Grund haben wir beschlossen die Qualität in drei Kategorien einzuteilen und für die weitere Untersuchungen zu verwenden.

Gemeinsame Festlegungen:

* Seed=42
* Drei Qualitäts-Kategorien: „schlecht“ für [3,4], „mittel“ für [5,6] und „gut“ für [7,8,9].  
  (nur für Algo.-Ergebnisvergleich.)
* Vergleichen den Score: „precision“, und davon „weighted average“ (aus z.B. dem Classification Report)

Es werden verschieden Algorithmen ausprobiert. Der mit der besten Precision wird dem Kunden als Ergebnis vorgelegt.

# Scores ermitteln

Tabelle mit den Scores zum Vergleich.

Weil wir nur Klassifikationen durchgeführt haben, wurde beschlossen den Score: „precision“ als „weighted average“ zu vergleichen.

Zusätzlich wurde vermerkt ob PCA verwendet wurde und wie viele Qualitäts-Kategorien gewählt wurden. Jeder hat mindestens einen Score mit drei Qualitäts-Kategorien.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Algorithmus** | **Bester score: precision,  weighted average** | **PCA  (ja/nein)** | **Qualitäts- Kategorien** | **Anmerkungen** |
| DecisionTree | 0,47 | nein | 7 |  |
| DecisionTree | 0,73 | nein | 3 |  |
| LogisticRegression | 0,71 | ja | 3 |  |
|  |  |  |  |  |
| MLPClassifier | 0,51 | nein | 7 |  |
| MLPClassifier | 0,49 | nein | 7 | Nur vier Spalten verwendet. Diejenigen die mit "alcohol" am stärksten korrelieren |
| MLPClassifier | **0,79** | ja | 3 | Bestes Ergebnis für MLPClassifier mit drei Q.-Kat. |
| MLPClassifier | 0,72 | nein | 3 |  |
| MLPClassifier | 0,75 | nein | 2 |  |
| MLPClassifier | 0,77 | ja | 2 |  |
| SVM | 0,73 | ja | 3 |  |

Der beste Score ist bisher 0,79 mit MLPClassifier. Die anderen folgen aber dicht darauf.

# Gridsearch und Ensemble Methoden

## Gridsearch

Im Projekt macht jeder einen Gridsearch und versucht Hyperparameter zu optimieren für einen optimalen Score.

## Ensemble Methode

(noch) nicht gemacht.

# Score auf der Testmenge

Passt er zu den Scores auf der Evaluationsmenge?(z.B. liegt innerhalb der Standardabweichung der cv-scores)

## Konnten wir unser gestecktes Ziel erreichen?

Verwendung der PCA Hauptkomponenten vor der Anwendung des MLPClassifier hat ein verbessertes Ergebnis geliefert.

Wenn mehr Zeit vorhanden wäre, wurden wir den Datensatz nach Alkoholgehalt aufteilen in ungefähr gleiche Teile und darauf den besten Algorithmus erneut laufen lassen. Grund ist die sehr starke Korrelation von „alcohol“ zur Zielspalte.

Ein weiterer Versuch wäre betreffend die Duplikate interessant. Warum waren diese vorhanden? Wie sieht der Score ohne Entfernung der Duplikate aus?

Wenn ja, was war die entscheidende Verbesserung bei der Methodenwahl?

Wenn nein, was hat gefehlt? Wie könnten wir das Resultat verbessern, wenn wir mehr Zeit hätten?

## Können wir unseren Resultaten trauen?

Etwas problematisch ist die geringe Anzahl der Daten für einige Qualitäts-Kategorien.

Es waren nur sehr wenige Daten für z.B. Qualität = 9, mit 5 Werten. Ebenso Kategorie 3, 4 und 8. Dadurch ist die bestimmung von sehr guten oder sehr schlechten Weinen schwierig.

Ist es wirklich sicher,/adäquat den finalen Algorithmus für die Vorhersage von neuen Daten zu nutzen?

Ist der Score der Testmenge innerhalb der Bandbreite der CV-Scores?

# Wir haben folgende Codebestandteile programmiert

a) eine Klasse, die die Ergebnisse der linearen Regression in Kategorien umwandelt

oder b) eine eigene Scoring-Funktion

oder c) eine eigenen Metrik-Funktion

oder d) eine Klasse, die im Preprocessing eingesetzt wird z.B. einen eigenen FunctionTransformer

oder e) eine Funktion, die Training, Cross-Evaluation ...etc für einen beliebigen Algorithmus durchführen kann

oder f) eine Funktion, die dazu da ist, in Monatsabständen/ oder Jahresabständen das Ergebnis zu überprüfen-