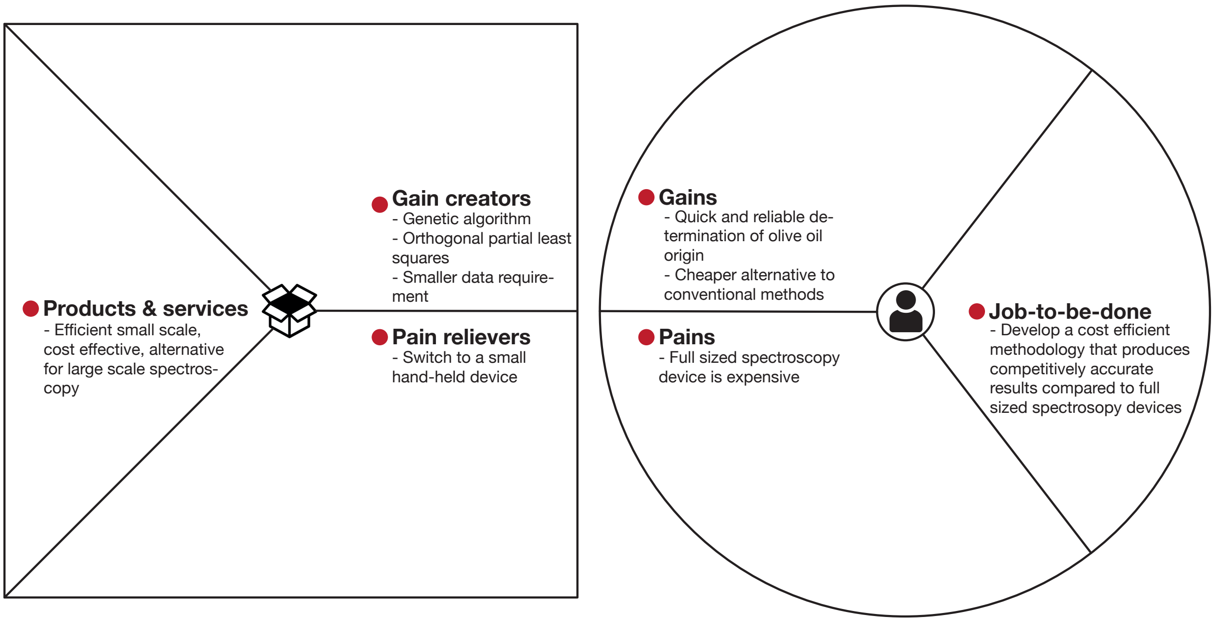
Pattern Recognition in Natural Science
Project Report
Olive Oils Discrimination

Lili Mészáros (s1015790), Evgeniia Martynova (s1038931)

January 2020

Introduction

In the food industry identifying the origin of a product has become a significant issue. Various factors determine the oil's chemical properties such as the soil it's grown on, the climate and agricultural practices may differ in countries. Several papers have been written on classifying olive oils, experiments include FTIR spectroscopy [1] and NIR and MIR spectroscopy fused [2], both combined with multivariate analysis. In this project we would like to see if olive oil origin could be determined by a smaller range of wavelengths, eliminating the use of a big spectrometer.



Products & services
- Efficient small scale, cost effective, alternative for large scale spectroscopy
Gain creators
- Genetic algorithm
- Orthogonal partial least squares
- Smaller data requirement
Pain relievers
- Switch to a small hand-held device
Gains
- Quick and reliable determination of olive oil origin
- Cheaper alternative to conventional methods
Pains
- Full sized spectroscopy device is expensive
Job-to-be-done
- Develop a cost efficient methodology that produces competitively accurate results compared to full sized spectrosopy devices

Figure 1: Value Proposition Canvas

Problem Definition and Research Goal

The papers mentioned earlier have been successful with predicting the origin of the oil. FTIR spectroscopy combined with PLS-DA had a cross-validation success-rate of 96% when using 8 PLS factors [1]. However, the partial least square's method has an issue of containing variation that is irrelevant for prediction, therefore we propose to use OPLS to improve the success rate. The genetic algorithm managed to achieve a 100% success-rate [1]. This result can hardly be enhanced, nevertheless, we are interested if it can be reproduced.

We propose to use smaller ranges for classification in order to reduce costs with a mini spectrometer. There are several ranges that such a device has, we are interested in 2 ranges, 790 to 1050cm^{-1} and 950 to 1700cm^{-1} because these are ranges that are currently available for example at Hamamatsu.

Moreover, double cross-validation will be applied instead of internal cross-validation used in the original paper, eliminating the bias it might cause.

Data Description

The data consists of sixty different samples of olive oils gathered from different producing regions from four European countries. The numbers of samples include 10 from Greece, 17 from Italy, 8 from Portugal and 25 from Spain.

The samples were collected in 2 periods within 1-24 days, thus resulting in 120 samples altogether. The spectral range was 800-4000 cm^{-1}. The spectra was then reduced to a smaller scale 799-1897 cm^{-1}, resulting in 570 data points [1]

Data Visualization and Biplots.

Data visualisation is the first step of data analysis. When multivariate data are being analysed, the dimensionality reduction should be performed first, so that the data representation in 2-D space was obtained. Principal Component Analysis (PCA) is a powerful and commonly used technique for multivariate data visualisation and analysis. It based on the idea that many high-dimensional data sets contain a limited amount of real information, and a few variables are enough to describe the data. The goal of PCA is to extract these latent variables from the data so that each new variable described the maximum remaining variance in the data and was orthogonal to the previous ones. The formula 1 shows how k_{th} principal component is represented as a linear combination of original variables:

$$PC_k = \sum_{d=1}^{D} l_{ki} x_i \tag{1}$$

D is a number of variables in the data set. The coefficients $l_k i$ are called loadings. The coordinates of patterns in the coordinate system defined by principal components are called scores. On biplot scores are shows as points and loadings as arrows from the origin. Loadings help to determine the correlated original variables (arrows pointing in the same direction) and which original variables are important for individual PCs. Also, it is clearly visible which samples have high values for particular variables by looking at which samples are far from the origin and lay on the direction of a loading for a variable. Finally, scores on biplot can give information about classes/clusters in data and how differentiable they are. [3]

For our research question PCA analysis may provide valuable information about the potential for discrimination of oils and also proper pre-processing choice. Loadings have less utility due to very large number of variables and high wavelengths correlation. The results of PCA analysis of olive oils data are given on Figure 8 in Appendix. Grouping of the samples by country is not perfect, however it is visible that oils can be separated by countries Also, the results demonstrate how autoscaling can mess up spectroscopy data.

Data Pre-processing.

Data pre-processing is an essential part of chemometrics data analysis. Measurements can be present in different units, often contain various artefacts and have other constraints. It helps to remove irrelevant information and highlight the information of interest in data. There are three types of pre-processing:

Variable-based. *Centring* - during data analysis we are not interests in the offset of data from origin, but we want to know the variation of data. Mean-centring, in which mean of the variable is subtracted from each measurement, is the most common method. Other options are median-centring, which is more robust if outliers are present, and control-centring (if one wants to emphasize the difference between the control group). centring should always be done before data analysis.

Scaling - helps to make variables given in different units of measurements have equal potential to contribute to the model by division of variable column by its standard deviation. This way variables measured on large scale do not dominate the model. However, there are situations when the difference in variance is essential for data analysis, e.g spectroscopy data. If such data are autoscaled the important information about peaks is removed.

Sample-based. *Normalization* - makes samples more cautiously by equalising their length. Should be applied cautiosly, because it introduces closures which lead to spurious correlations between samples. *Bucketing/alignment* - removes small shifts from individual peaks. *Baseline correction* - removal of slope in measured data (instrumental artifacts). It often is needed for analysis of spectroscopy data when information about the peaks in important, thus baseline presence affects the results of the analysis.

Value-based. *Transformation* - log-transformation is commonly used It makes the error of the data 'homoscedastic', less dependent on the absolute value. *Filtering* - removes instrumental spikes and missing values

There are many possible combinations of pre-processing techniques and it is very important to thoughtfully consider which combination is the best for a particular case. We work with FTIR spectroscopy data. From the raw data plot given on Figure 6 in the appendix it can be seen that there is no baseline and there is little noise. Except for mean-centring also scatter correction can be used, however, we do not apply it because of project time constraints. The mean-centered spectra of samples from different countries are depicted on Figure 7 in the Appendix. The similarities between groups are clearly noticeable now and it is a perfect demonstration of how the removal of variables offset can alleviate discrimination.

Alignment

It is a common problem in chemical measurements like spectra, chromatograms, LC-MS and GS-MS that identical features appear at different positions in different measurements. This makes it more challenging to analyse data, therefore it is crucial that these peaks are aligned during preprocessing. There are multiple reasons why problems like these arise, for example it could be changes in the temperature, pH or pressure.

There are various methods to solve misalignment, ranging from simple techniques as bucketing or slightly more complicated ones such as warping. Warping is a solution for spectroscopic, spectrometric and chromatographic signal alignment.

The general idea of warping is to shift, stretch and/or compress the function describing the raw data along their x-xis. Mathematically, the warping function $w(x)$ which in this case means shifting the original function by c points can be expressed as

$$f(w(x)) = f(x + c),$$

where c can be determined by an optimisation procedure. This is where more sophisticated algorithms come into the picture. *COW* and *dtw* use dynamic programming, *ptw* uses iterative regression, while *PAFFT*, *RAFF* and *icoshift* use cross-correlation via fast Fourier transformation. Moreover, in most cases peaks have several shifts and only parts of the query profile need to be transformed, requiring complex algorithms. It is worth noting that since warping methods are functions, peaks cannot be swapped.

Parametric time warping (ptw) is an extension of the simple warping function, where the function is a polynomial

$$w(x) = \sum_{k=0}^{K} a_k x^k.$$

The advantage of *ptw* is that it is easy to interpret, though it processes the whole query profile and it is inflexible. *stw* offers a solution to inflexibility by changing the polynomials to B-splines.

Both *dynamic time warping (dtw)* and *correlation optimised warping (COW)* are based on dynamic programming. While *dtw* relies on Euclidean distance and applies pointwise warping, *COW* uses correlation and piecewise linear stretching.

PAGA is based on a genetic algorithm that aims to find the optimal combination of shift and stretch segments. Optimisation is an issue for a genetic algorithm, nevertheless it can be useful. [4]

There are numerous more examples of warping methods, but since the focus of the project is not on this, warping is not investigated more thoroughly.

Our project uses PCA and OPLS, so it is important that the peaks are aligned. Luckily, the data was already aligned, therefore we did not have to do it ourselves.

SOMs.

Self-organising maps (SOMs) is a clustering method which maps high dimensional data to a 2D grid of units according to the similarity between neighbours [5]. Each unit corresponds to a cluster and the number of clusters is defined by grid's size and shape, which can be rectangular or hexagonal. This way the topology of the data in high-dimensional space is preserved in the two-dimensional map, which makes SOMs is a perfect visualisation technique.

SOM is initialised by assigning a random *codebook vector* to every unit, which can be treated as a typical pattern associated with this unit. This can be done by randomly assigning a subset of training objects to units. Then SOM is trained with the algorithm:

1. Pick a random object o_j.
2. Determine the map u which codebook vector is the most similar to o_j (winning map).
3. Update the winner and the units in its neighbourhood in the following way $u_{i+1} = (1-\alpha)u_i + \alpha o_j$
4. During training gradually decrease neighbourhood size and the learning rate α, so that the map converges.
5. The algorithm ends after a pre-defined number of operation or when unit's codebook vectors stop changing.

Many different parameters settings of SOMs can be tried and also the clustering results may be different even with the same setting dues to random initialisation of codebook vectors. Thus, it is recommended to experiment with SOM settings and repeat clustering a few times before drawing the conclusions. Data visualisation and clustering are classical SOMs applications, however, it can be used for classification as well if class information is modelled as a dependent variable.

For our data set SOMs can be used for visualisation and potentially discovery of cultivars of olive oils within a country. However, codebook vectors representation is not useful due to a very large number of variables.

MCR and related techniques.

Spectral data are often measured on the mixtures of the components. If we want to extract the information about the number of different components in spectra and their concentration, we need to use MCR techniques. PCA model is not useful for this purpose, but due to an infinite possible way to extract components scores and loadings (rotation ambiguity), a chemically meaningful decomposition can be obtained. MCR model formula:

$$\mathbf{D} = \mathbf{C}\mathbf{S}^{\mathbf{T}} + \mathbf{E} \tag{2}$$

Where **D** is spectra or chromotogram data. Columns of **C** correspond to a concentration profile of one chemical species and rows to samples. Rows of $\mathbf{S}^{\mathbf{T}}$ correspond to a pure spectra of one chemical species and columns to samples. **E** contains an error term. There are many different methods, but the most popular one is MCR-Alternating Least Squares (MCR-ALS).

SIMPLISMA - selects the purest data spectra rows or concentration profile. Variable purity is defined by $p_i = \frac{s_i}{m_i + f}$, where s_i is a standard deviation, m_i is mean and f is noise %.

EFA - evaluates the number of components and time when they emerge and decay by subsequently selecting subsets of data (by rows or columns) and applying PCA.

MCR-ALS - allows to put the constrains on **C** and $\mathbf{S}^{\mathbf{T}}$ based on the domain knowledge and iteratively evaluate **C** and $\mathbf{S}^{\mathbf{T}}$ based on their values using Moore-Perouse inverse. Some possible constrains: uni-modality, non-negativity, closure, etc. (possible due to rotation ambiguity and intensity ambiguity).

Some application areas of MCR: spectroscopy, chromatography, electrochemistry, temperature dependence, pH dependence, hyperspectral imaging.

MCR technique can be used to extract and analyse the components of olive oils FTIR spectra.

OPLS

OPLS is a preprocessing method for multivariate data that aims to improve PLS. PLS creates a linear regression model by projecting the predicted (**X**) and observed (**Y**) variables to a new space. **X** consists of n rows of observations and p columns of independent variables and **Y** is a $n \times k$ matrix where k is the number of dependent variables. In matrix form this can be written as

$$\mathbf{Y}_{nk} = \mathbf{X}_{np}\mathbf{B}_{pk} + \mathbf{E}_{nk} \tag{3}$$

Where **B** is a matrix of regression coefficients and **E** is a matrix of residuals.

However, **X** contains variation that might not be relevant for the prediction of **Y**. The OPLS method successfully solves this issue by removing the systematic variation from **X** that is orthogonal to **Y**. OPLS improves prediction and reduces model complexity, thus results can be evaluated more quickly.

Spectra often contains variation uncorrelated to the responses, resulting in unreliable predictions, therefore it is reasonable to apply it for this project. [6]

KPLS

As OPLS, KPLS is an extension of the PLS algorithm. The great advantage of KPLS is that it can create a non-linear regression model in high-dimensional feature space. The K in KPLS indicates that the method relies on kernels. The beauty of the method is that only linear algebra is required after the kernel matrix has been established. Moreover, kernels provide a very general approach to the problem of non-linear PLS models.

The idea of kernel partial least squares regression is to map input variables to a feature space with $\Phi : x_i \in R^n \rightarrow \Phi(x_i) \in \mathcal{F}$, constructing Φ, a matrix of regressors. From this, the kernel Gram matrix **K** is composed:

$$\mathbf{K} = \Phi\Phi^T$$

With the use of **K** consisting of cross dot products there is no need for an explicit mapping. In the final algorithm, the dot products can be replaced by kernel functions. [7]

KPLS could be applied to our project since kernel methods can be reduced to lower dimensions. However, it is not worth investing time in it if a simpler method can be used.

Validation and related procedures.

Validation is an essential step of building a model because it allows estimating model performance (prediction accuracy) on unseen data and prevents a model to be biased against training data. The main idea of validation is the splitting of the data set into training and tests sets. The test set should be representative of the training set, thus it is important to properly divide data. The most obvious way is a random selection, however, in this case, we should check if there are unlucky divisions and repeat the splitting. Also, if there are categories in the data, they should be evenly represented in both training and tests sets. An alternative technique, that has been found to demonstrate better results, is Duplex method, which alternatively puts the most distant samples into training and tests sets.

Cross-validation is a technique to make several splits of data into training and test sets and calculation of average performance metrics (e.g. RMSE). The percentage of data in the test set is defined and with this percentage we repeat the procedure so that on each step new samples are moved to test set until all the objects has been left our once. There are different kinds of cross-validations:

- Leave-One-Out (LOO) - the size of the test set equals one. It is shown to be unbiased but can have high variance.
- Leave-n-Out (LnO) - a fraction of data is left our, usually $10 - 20\%$. The largest errors cancel out to an extent and variance decreases, however, it is more biased than LOO.

In case when model parameters are needed to be optimised, simple cross-validation is not enough, because we need an independent measure of model performance. The solution is **double cross-validation**. After data division into training and independent test set, the training set is further separated into training subset and validation set. The model parameters are optimised using training set only; this procedure is called **inner cross-validation**. After best parameters values are determined the model is validated with independent test set (**outer cross-validation**).

Other possible validation methods are Jackknife, Bootstrap and permutation test. Basically, they verify that the classification results are statistically significant.

In our project double cross-validation is used since we need to optimize both numbers of latent variables in OPLS and wavelengths sets selected with GA. Since we have a small number of samples in the data set, LOO is used for internal cross-validation. The traits of our data, that should be taken into account for validation, are that there are four classes and also there are duplicated measurements for each sample. When split the data for double cross-validation we check that samples from different classes are evenly distributed between training and test sets and also that duplicated measurements were kept in one set only. This is needed to prevent using of information from test set for model training. Accordingly, in our case, LOO CV is actually "leave-two-out", because we take out both measurements of one sample.

SVM

Support vector machines are non-linear binary classifiers. SVM is a kernel-based classifier, therefore as in KPLS, the input space is mapped to a linear high-dimensional feature space where the goal is to find the optimal separating hyper plane.

Support vector machine is a basic machine learning algorithm, therefore the first step in constructing such a model is to create a training, test and validation set. Since we are trying to map the problem into a higher dimension and kernels are rather general, the next step is choose the right kernel function and its parameters. A regularisation constant needs to be chosen as well, that determines how much misclassification is allowed. The regularisation constant C needs to be optimised under certain constraints, a solution to that is Lagrange multipliers. Next, the kernel matrix can be calculated for the training set and the model can be created. The performance of the test set is evaluated with the model.

Since SVM is a binary classifier, in the olive oil case, 2 approaches can be used, use one against all (N models) or one against one ($\frac{N(N-1)}{2}$ models).

Variable Visualisation in case of kernels

Though KPLS is a useful technique, visualisation of the results is rather challenging since the data is rather complex and the contributions of different variables is lost when transformed to another space. A solution proposed for this issue was to use pseudo samples.

The PLS regression model is expressed as seen in equation (3). **B** seen in this equation combined with the Kronecker-delta can produce the pseudo samples. For example the first predicted value is given by equation (4).

$$[1, 0, \ldots, 0]_{(1\times m)}\mathbf{B}_{m\times 1} = b_1 \tag{4}$$

By applying a series of pseudo samples trajectories, all variables can be calculated and plotted. [8]

Kernel visualisation does not provide any additional information for normal PLS, neither does it provide any new observations for OPLS. As a result, this method is not applicable for our project.

ASCA.

ASCA is a method for multivariate data analysis in a particular experimental setup [9]. In this case, different sources of variation are present, e.g. measurements taken at different points in time, the difference between individuals involved into the experiment. Thus, in the analysis the experimental design and the relationship between different variables should be considered to gain insight into the system.

ANOVA is a method used to analyze data from experimental design to determine the effect of different experimental factors on the variation in a dataset. The ANOVA equation for an experiment in which two experimental factors α and β are varied and at each combination one sample is measured on a dependent variable is given below:

$$x_{cdj} = \mu + \alpha_{cj} + \beta_{dj} + (\alpha\beta)_{cdj} \tag{5}$$

In Equation 5 x_{cdj} is an $j_t h$ variable of a data set observed for a sample on level of c and d, μ is an offset term, α_{cj} and β_{dj} are the model parameters on levels c and d respectively, $(\alpha\beta)_{cdj}$ is an error term. To obtain the unique solution the constraints should be imposed on the parameters. The commonly used constraints are: $\sum_{c=1}^{C} \alpha_{cj} = 0$, $\sum_{d=1}^{D} \beta_{dj} = 0$, $\sum_{c=1}^{C} (\alpha\beta)_{cdj} = 0 \forall d$ and $\sum_{c=1}^{D} (\alpha\beta)_{cdj} = 0 \forall c$. A matrix form of 5 Equation is the following:

$$\mathbf{X} = \mathbf{m^T} + \mathbf{X}_\alpha + \mathbf{X}_\beta + \mathbf{X}_{\alpha\beta} \tag{6}$$

ANOVA is univariate, this it cannot be used to take the covariance between different variables into account. However, due to the constraints described above the variance of different components can be separated: $\|\mathbf{X}\| = \|\mathbf{m^T}\| + \|\mathbf{X}_\alpha\| + \|\mathbf{X}_\beta\| + \|\mathbf{X}_{\alpha\beta}\|$. Thus, the contribution of each factor to the total variance in data can be determined by using PCA (or SCA). So, ASCA model is obtain by combining ANOVA and PCA. The ASCA model which corresponds to the Equation 6 is the following:

$$\mathbf{X} = \mathbf{m^T} + \mathbf{T}_\alpha \mathbf{P}_\alpha^\mathbf{T} + \mathbf{T}_\beta \mathbf{P}_\beta^\mathbf{T} + \mathbf{T}_{\alpha\beta} \mathbf{P}_{\alpha\beta}^\mathbf{T} + \mathbf{E} \tag{7}$$

In the Equation 7 $\mathbf{T}_\alpha$, $\mathbf{T}_\beta$ and $\mathbf{T}_{\alpha\beta}$ are scores of components, $\mathbf{P}_\alpha^\mathbf{T}$, $\mathbf{P}_\beta^\mathbf{T}$ and $\mathbf{P}_{\alpha\beta}^\mathbf{T}$ are loadings and **E** are residuals. By analysing the loadings of different components, the relationship between the variables can be identified for every contribution to the variation.

For our data set ASCA usage does not make sense, because our data are just FTIR spectroscopy measurement without a specific experimental design.

Multi-way data and PARAFAC.

Parallel Factor Analysis (PARAFAC) is a method to analyse multi-way data [10]. Multi-way data are the data arranged in a three or higher-dimensional structure. An example of such data is fluorescence emission spectra measured at several excitation wavelengths for several samples. These data are arranged in a cubic structure instead of a matrix and appropriated decomposition should be done to analyse them.

PARAFAC can be viewed as a generalization of bilinear PCA. A decomposition of the data is made into trilinear components, but instead of one score vector and one loading vector as in bilinear PCA, each component consists of one score vector and two loading vectors. The formula of PARAFAC decomposition:

$$x_{ijk} = \sum_{f=1}^{F} a_{if} b_{jf} a_{kf} + e_{ijk} \tag{8}$$

Where A, B and C are loadings matrices and e_{ijk} are residuals. The model is obtained by minimising the sum of squares of the residuals. The extracted components are not orthogonal and it is impossible to measure variance explained by a component, but only with the overall model. However, PARAFAC has a few benefits over PCA. Its solution is unique since there is no rotational ambiguity in decomposition. Also, PARAFAC components can be interpreted as spectra and thus the results can be used for MCR. Our data do not have a multi-way structure, therefore, PARAFAC cannot be applied.

Optimization techniques.

Optimization methods are useful when a model contains a few parameters and a combination of these parameters should be optimised. In other words, we need to find an optimum of a function with many local optimums. In context of optimization such function is called **cost function/ objective function/ fitness function**. All the optimisation techniques require a methodology, evaluation and stop criteria. Generally, there are two groups of optimization methods:

Local optimization methods

Simplex optimization is the most efficient local optimization method. Explores the response surface with triangular structures (simplexes) by measuring the values on the vertexes, finding the worst one and mirroring this vertex through surface defined by remaining vertices. The algorithm stops when simplex starts to circulate. Nelder-Mead Simplex is an improved algorithm in which the size of simplex is modified. This way the steps of the algorithm are larger when a current simplex is far from optimum and smaller when it is close.

Other possible local optimization techniques are "single factor at a time" strategy and box-type optimization. Both are less effective and efficient than simplex. Since local optimization method explores an only small of a surface they may easily end up in a local optimum when applied to complex problems.

Global optimization methods

Allow to find a global optimum, but requires a large number of experiments. Several methods exist from which **genetic algorithm (GA)** is the most basic. GAs use biological evolution as a framework for search. The steps of GA:

1. Generate initial population. It can be done randomly or based on a priori domain knowledge.

2. Evaluate trial solutions. On this step, the quality of the trial solutions is accessed by evaluation of fitness function. A fitness function should be carefully designed since it may affect convergence rate and needed computational resources.

3. Construct new population. For a new population the solutions with higher fitness function values are selected. Then the population is modified by applying crossover and mutations. Crossover mating strings at random and recombine their part, whereas mutation changes the strings.

4. Repeat until the stop criterion is reached - maximum number of iterations exceeded or convergence.

The goal of our data analysis is to find a small subset of 570 wavelengths that gives a high classification accuracy. If we want to select m wavelengths the number of possible combinations is $N = \binom{570}{m}$, which is huge. Also, since the wavelengths values tend to be highly correlated, it is possible to have many local optimums. Thus, we decided to apply GA to our problem.

Data Fusion

Data fusion is the idea of combining several data sets originating from the same source into one. Data often contains noise and complementary information, this method makes it possible to increase accuracy or extend insight than what could be attained from individual sources. There are different levels of data fusion, depending on when the fusion happens.

Low level data fusion occurs at the beginning of the analysing process. The data sets are simply attached after each other, same size matrices $\mathbf{X}_1$ and $\mathbf{X}_2$ are concatenated into a new $\mathbf{X}$:

$$\mathbf{X} = [\mathbf{X}_1 \quad \mathbf{X}_2].$$

Either different matrices are concatenated to make use of the correlations in the data, or if there is noise, the same information is averaged by using the above equation with $\mathbf{X}_1 = \mathbf{X}_2$. This approach can be applied to our project, as the different wavelengths can be split up and evaluated apart and joined, similarly as Casale et al. did. They analysed olive oil samples measured with NIR and MIR spectroscopy and did classification separately and combined. [2] It is important to note that the wavelengths in our data set are only MIR and might not give new insight.

Low level data fusion can also be applied to non-linear data, by first creating separate kernels and then adding them together in some kind of linear combination.

Compared to low-level data fusion, mid-level fusion occurs later on in the process, first the dimension is reduced and then classification is carried out. It is valuable if the data has a lot of noise.

Results of data analysis - OPLS

To avoid over-fitting, the first step of OPLS is to determine the number of components. To choose the correct number, some kind of significance criterion needs to be used, for example cross-validation or large eigenvalues. In this project the number of latent variables are determined by cross-validation. The plot for this can be seen on figure 2 where it is visible that convergence starts at 4, therefore 4 latent variables are used for OPLS. Before applying the method, the data is split into a test set and a validation set, 25% of the original data is for testing, 75% for validation. When applying cross validation to the result of OPLS, $\approx 4\%$ of the samples are misclassified. The results qualify as statistically significant because the Q^2 results are around 0.55-0.7.

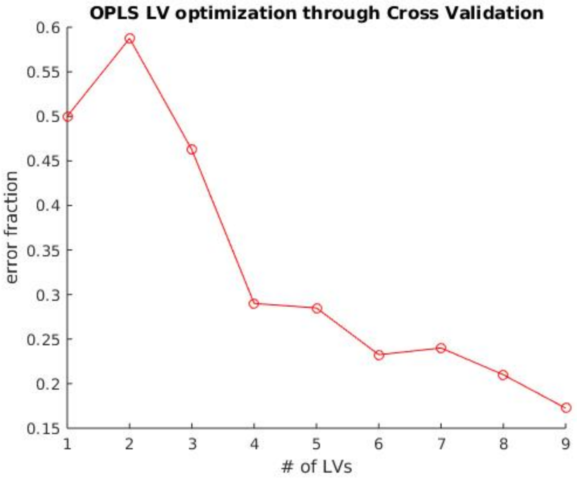


Figure 2: Latent variables

To decrease misclassification, the number of latent variables is increased to 8 and by that the success rate increases to 100%. This classification can be seen on figure 3 where a country is clearly separated from the rest and the test and validation sets can be spotted too.

After observing the overall data set, the two smaller ranges of wavelengths are investigated, 790 to 1050cm^{-1} and 950 to 1700cm^{-1}. The first range has 131 wavelengths, while the latter has 389. On

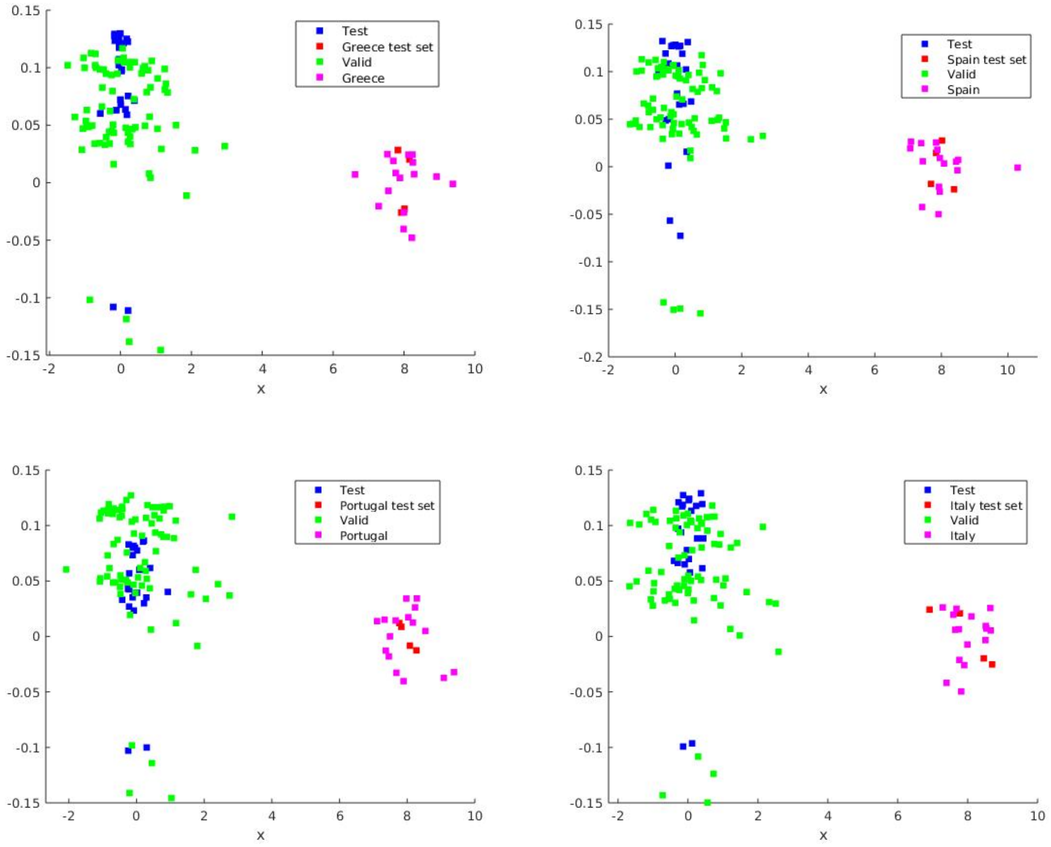


Figure 3: Countries are separated from the other locations by OPLS.

figure 4 the two different ranges' plots of the latent variables can be seen. Based on the images, in both cases 6 is picked as the best choice for prediction.

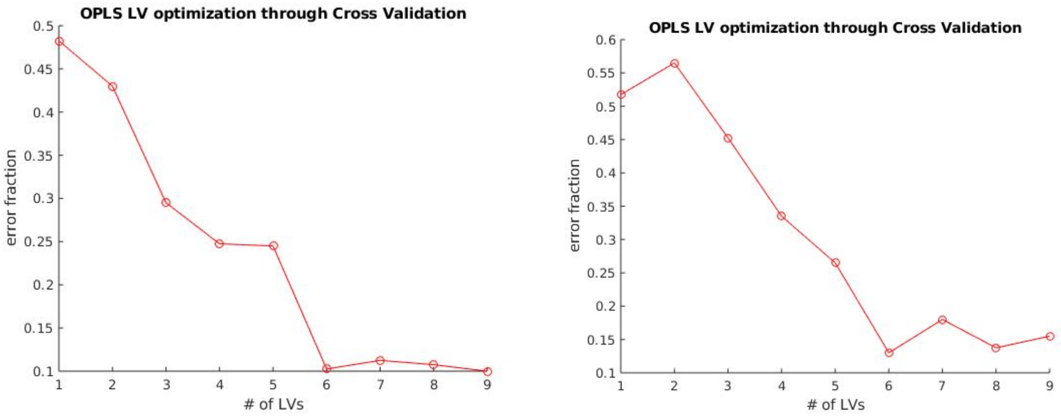
The two smaller subsets provide about the same results as the whole data set, the ≈ 96% success rate holds as in the entire set and Q^2 gives high values usually around 0.6, so the model is statistically reliable too. In conclusion, the experiment of using smaller ranges is satisfactory because while they do not improve prediction, they do reduce costs.

Results of data analysis with GA.

Based on the results of OPLS we aim to find a subset of 8 wavelengths to classify the oils with LDA. The initial population is generated randomly because we have little a priori knowledge to use dedicated initialisation. Since we are looking for 8 wavelengths, we chose real values encoding of trial solutions, which correspond to wavelengths numbers. For new solutions generation 2-points crossover has been implemented because it has lower positional bias than 1-points crossover and mutation we devised is shifting a wavelength number by an integer drawn for (-10, 10) uniform distribution. Generational reproduction with ranked-based selection strategy is used to select new generations. The fitness function is chosen to be an error rate of LDA classification of oils with a subset of 8 wavelengths calculated with LOO internal cross-validation. The GA is stopped either when zero error rate is reached (converged) or the maximum number of iterations is exceeded. Finally, the results of GA are validated with an independent test set.

Due to the time constraints, we managed to try 2 different GA settings. For each, the GA was run 100 times, each run took about 3 hours. For a simple performance baseline, the average error rate of LDA classification for 1000 randomly selected wavelengths subsets. Both GA and random baseline was accessed with the same independent test set. The results are represented in Table 1:

It can be seen that in spite of perfect convergence GA1 did not perform well. It could be caused



790-1050cm^{-1} 950-1700cm^{-1}

Figure 4: Estimating the number of latent variables.

Method	Average error rate	Convergence	Zero error rate runs
Random baseline 1	22.11	-	-
GA1	18	100 of 100 (min 1, max 12)	0
Random baseline 2	10.44	-	-
GA2	4.85	77 of 100 (min 4)	24

Table 1: The statistics for 2 GA parameters settings. GA1 - population size: 200, max number of iterations: 30; GA2 - population size: 50, max number of iterations: 50. In Convergence column ”min” is a minimum number of iterations after which convergence is reached, ”max” - a maximum.

by a unlucky division of data into training and test sets because as we can see the average error rate of random baseline 1 is significantly higher than for random baseline 1. Also, it might be due to convergence properties with 200 population size.

Further, we evaluate zero error rate the solutions obtained by GA2 with 4-fold cross-validation (since we have only 8 unique samples from Portugal we want at least 2 of them in the test set). To cope with possible unlucky data division, double-CV is performed 10 times with different training and test sets splits and the results are averaged for each solution. The statistics is shown in Table 2

Top ten solutions average error rates									
0.33	1.24	1.25	1.25	1.48	1.73	1.83	1.83	1.83	1.88

Table 2: The mean error rates for top-10 solutions found with GA2.

The results of the evaluation demonstrate that the best solutions found by GA2 are robust and have a low error rate. The wavelengths of the best solution (0.33 error rate) are: 964.8, 1011.1, 1126.9, 1279.3, 1373.9, 1599.7, 1622.8, 1667.2. They are shown as dashed lines on the mean values of spectra for different countries on Figure 5. It can be noticed that the wavelengths from solution correspond to peaks of averaged spectra well.

Conclusion

Even though the results of GA were worse than OPLS classification, they are promising, because with 200 GA runs only we could obtain a subset of 8 wavelengths, which demonstrated robust results for LDA classification under double-CV. We think that with further improvements of GA, such as the dedicated population initialisation, more advanced crossover and mutations and more runs even better results can be gained. Thus, the development of a spectrometer for rapid and reliable oils classification is a feasible task.

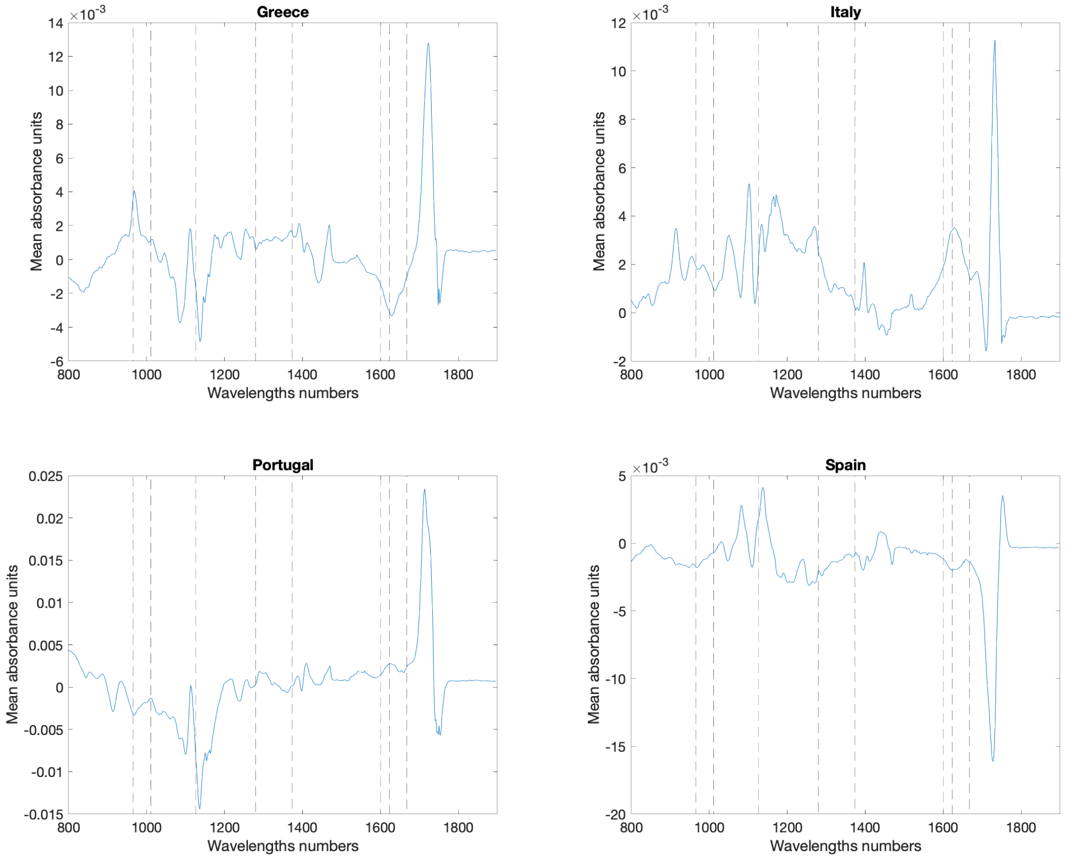


Figure 5: Wavelengths selected by GA2 shown as dashed lines.

References

[1] Henri S. Tapp, Marianne Defernez, and E. Katherine Kemsley. FTIR Spectroscopy and Multivariate Analysis Can Distinguish the Geographic Origin of Extra Virgin Olive Oils. *Journal of Agricultural and Food Chemistry*, 51(21):6110–6115, 2003. PMID: 14518931.

[2] Monica Casale, Nicoletta Sinelli, Paolo Oliveri, Valentina Di Egidio, and Silvia Lanteri. Chemometrical strategies for feature selection and data compression applied to NIR and MIR spectra of extra virgin olive oils for cultivar identification. *Talanta*, 80(5):1832 – 1837, 2010.

[3] Ron Wehrens, Jan Gerretzen, and Geert Postma. *Chemometrics I*. 06 2017.

[4] Tom G. Bloemberg, Jan Gerretzen, Anton Lunshof, Ron Wehrens, and Lutgarde M.C. Buydens. Warping methods for spectroscopic and chromatographic signal alignment: A tutorial. *Analytica Chimica Acta*, 781:14 – 32, 2013.

[5] Ron Wehrens, Lutgarde MC Buydens, et al. Self-and super-organizing maps in r: the kohonen package. *Journal of Statistical Software*, 21(5):1–19, 2007.

[6] Johan Trygg and Svante Wold. Orthogonal projections to latent structures (O-PLS). *Journal of Chemometrics*, 16(3):119–128, 2002.

[7] Roman Rosipal and Leonard Trejo. Kernel partial least squares regression in reproducing kernel hilbert space. *Journal of Machine Learning Research*, 2:97 – 123, 12 2001.

[8] G.J. Postma, P.W.T. Krooshof, and L.M.C. Buydens. Opening the kernel of kernel partial least squares and support vector machines. *Analytica Chimica Acta*, 705(1):123 – 134, 2011. A selection of papers presented at the 12th International Conference on Chemometrics in Analytical Chemistry.

[9] Jeroen J Jansen, Huub CJ Hoefsloot, Jan van der Greef, Marieke E Timmerman, Johan A Westerhuis, and Age K Smilde. Asca: analysis of multivariate data obtained from an experimental design. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 19(9):469–481, 2005.

[10] Rasmus Bro. Parafac. tutorial and applications. *Chemometrics and intelligent laboratory systems*, 38(2):149–171, 1997.

APPENDIX

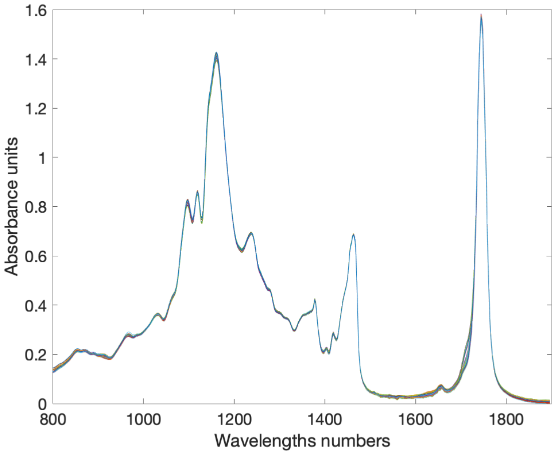


Figure 6: Raw spectra of all samples

Source code

The source code for of the implementation can be found in a repository PatternRecognition. https://github.com/meszlili96/PatternRecognition

Authorship of the sections

1. A problem definition and goal of your research, taking into account the Framework for Convergence and the Value Proposition Canvas - Lili Mészáros
2. A data description - Lili Mészáros
3. Visualization of the data and Biplots - Evgeniia Martynova
4. Pre-processing of the data - Evgeniia Martynova
5. Possible Alignment - Lili Mészáros
6. SOMs - Evgeniia Martynova
7. MCR and related techniques - Evgeniia Martynova
8. OPLS - Lili Mészáros
9. KPLS - Lili Mészáros
10. Validation and related procedures - Evgeniia Martynova
11. SVM - Lili Mészáros

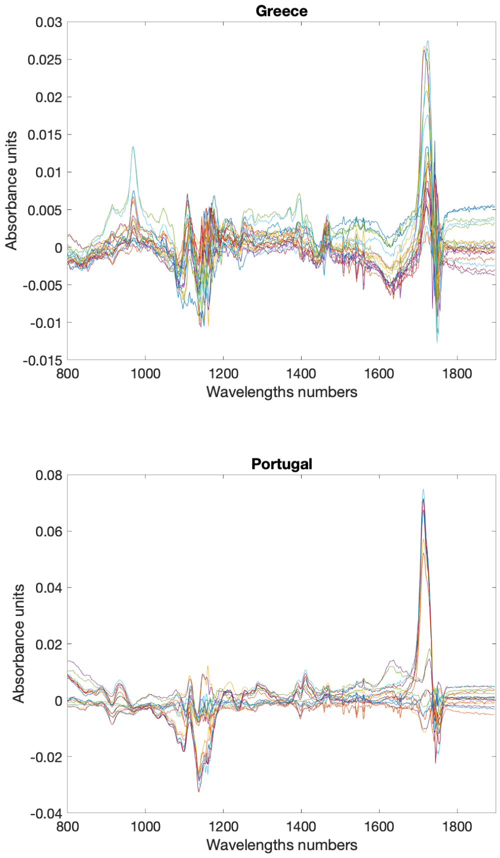


Figure 7: Mean-centered spectra by country.

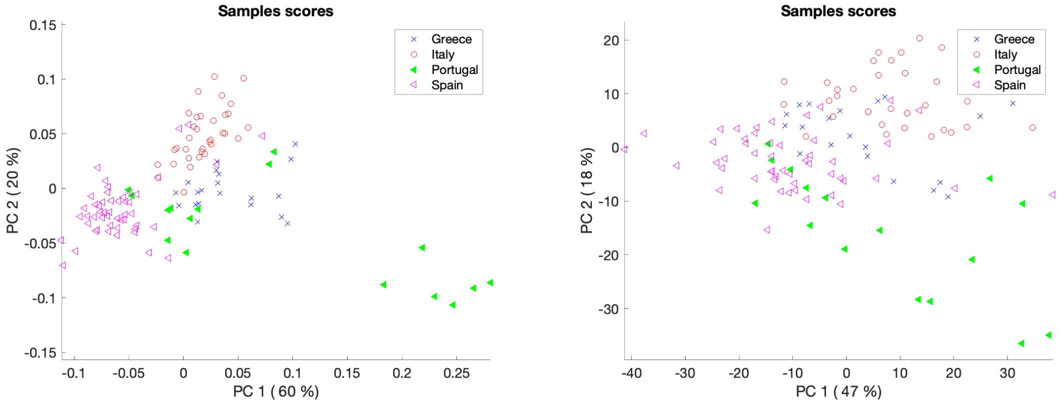


Figure 8: Samples scores from PCA analysis with 6 PCs. Mean-centered data are on the left and autoscaled are on the right.

12. Variable Visualization in case of kernels - Lili Mészáros
13. ASCA - Evgeniia Martynova
14. Multiway data and PARAFAC - Evgeniia Martynova
15. Optimization techniques - Evgeniia Martynova
16. Data Fusion - Lili Mészáros
17. Results of data analysis Method 1 - Lili Mészáros
18. Results of data analysis Method 2 - Evgeniia Martynova