

CHEMOMETRIC DATA ANALYSIS WITH AUTOENCODER NEURAL NETWORK

Muhammad Bilal¹ Mohib Ullah² Habib Ullah³

¹Department of Information Engineering and Computer Science, University of Trento, Italy.

²Department of Computer Science, Norwegian University of Science and Technology, Norway.

³College of Computer Science and Engineering, University of Ha'il, Saudi Arabia.

ABSTRACT

We propose novel deep learning based chemometric data analysis technique. We trained L2 regularized sparse autoencoder end-to-end for reducing the size of the feature vector to handle the classic problem of the curse of dimensionality in chemometric data analysis. We introduce a novel technique of automatic selection of nodes inside the hidden layer of an autoencoder through Pareto optimization. Moreover, Gaussian process regressor is applied on the reduced size feature vector for the regression. We evaluated our technique on orange juice and wine dataset and results are compared against 3 state-of-the-art methods. Quantitative results are shown on Normalized Mean Square Error (NMSE) and the results show considerable improvement in the state-of-the-art.

Index Terms— Chemometric data, sparse autoencoder, Gaussian process regressor, pareto optimization.

1. INTRODUCTION

Chemometrics or multivariate data analysis is the science which applies statistical and mathematical methods to process the data obtained through spectroscopic techniques, in order to derive information of interest. The need for chemometric analysis comes from the development of analytical instruments and techniques that are capable of producing large amount of complex data. Data collection through spectroscopic technique is based on interaction of light energy of variable wavelength with samples under test [1]. The ability of a sample to absorb or transmit light energy is recorded in terms of values throughout a selected bandwidth of electromagnetic spectrum. Whether it be food, pharmaceutical or textile industry, concentrations of chemical components of interest in samples are estimated through chemometric analysis. Spectroscopic data often comprises of more spectral features or variables than the samples or observations. This not only creates curse of dimensionality issue but also the consecutive variables in a spectrum are highly correlated in nature, that is, some spectral variables can be represented as linear combination of other independent variables. The work of Bertrand et al. [2] and Sutter et al. [3] have shown that existence of such high collinearity between the spectral vari-

ables can result in inaccurate predictions. It is problematic to apply directly statistical methods due to high collinearity, like multiple linear regression (MLR) in [4–7].

In order to deal with data dimensionality and data redundancy, different works have been proposed in literature [8]. Most of the previous work suggests reducing the number of variables or features to cope with curse of dimensionality problem, thus allowing to get more accurate results through regression techniques. The feature reduction can be generally achieved in two different methods. The first one contains selecting the most relevant features based on a chosen criterion from the original set of features [9–12], while in second case the original features are transformed from one space to another in such a way to keep the reconstruction error as minimum as possible. This transformation of features set from one space to another can either be linear or non-linear depending upon the scope of application. The examples of later method are Principle Component Regression [13] (PCR) and Partial Least Square Regression (PLSR) [6]. PCR is composed of simple linear regression model based on few principle components of the original spectral data. While PLSR focuses on calculating the linear projections that shows maximum correlation with the output or target variable, thus estimating a linear regression model determined by the projected coordinates. Benoudjit et al. [10] proposed linear and nonlinear regression methodologies which are based upon an incremental routine for feature selection and using a validation set. In [11, 12] different techniques have been introduced to improve the results of previous method by choosing the best feature set for initializing the routine and finding a feature selection strategy that depends entirely on the shared information between spectral data and target variable. An interesting approach to the chemometrics problems has been discussed in [14], where instead of traditional feature reduction approach, the whole information in spectral data space is exploited by using Multiple Regression System (MRS). Similarly, Douak et al. [15] come up with two stage regression approach that is based on residual-based correction (RBC) concept. Their basic idea is to correct any adopted regressor, called functional estimator, by analyzing and modeling its residual errors directly in feature space. The results obtained from MRS technique outperform the ones obtained through

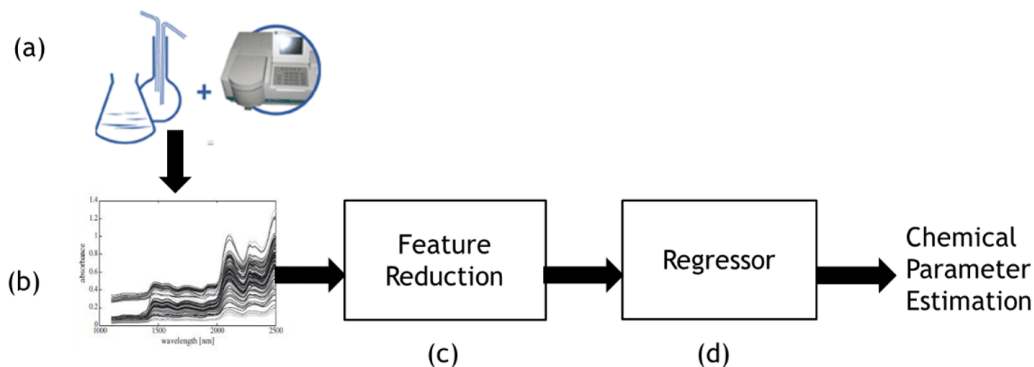


Fig. 1: (a) Data collection through spectrophotometry. (b) Spectral signature. (c) Feature reduction through autoencoder neural network. (d) Gaussian process regressor for estimating chemical parameter of interest.

traditional feature reduction techniques such as PCR and PLSR [6, 14].

Compared to these approaches, we proposed an automatic deep learning based feature extraction technique. In the last few years, deep learning has shown outstanding results on a variety of computer vision problems like tracking [16], video summarization [17], crowd analysis [18, 19], segmentation [20, 21], and anomaly detection [22], to name a few. In a nutshell, deep models learn hierarchical features [23] and have the capability to learn the structure of the underlying data. Inspired by this, we designed an autoencoder neural network [24] for removing redundant and irrelevant features from the spectral data. It is an automatic feature extraction technique that is capable of linear and nonlinear feature extraction based upon the selection of parameters of the architecture. Pareto optimization technique is applied in order to choose the best architecture (in terms of model complexity) of autoencoder for the feature extraction. After extracting meaningful features from the original feature set, we exploited Gaussian process regression to solve our regression problem.

The rest of the paper is organized in the following order. In the section 2, we explain our proposed method. In section 3, we present autoencoder training and parameter optimization. Section 4 illustrates our baseline regressor. In section 5, quantitative results are presented and section 6 concludes the paper.

2. PROPOSED APPROACH

We focus on estimating sugar and alcohol concentration in orange juice and wine datasets. The whole pipeline can be seen in 3 discrete steps. In the first step, data is acquired from the liquids through near infrared (orange juice data set) and mid infrared (wine dataset) reflectance spectroscopy technique Fig. 1(a,b). In the 2nd step, the acquired data is processed through auto-encoder neural network for feature reduction Fig. 1(c) and then in the last step, Gaussian process regressor estimate the concentrations of mentioned compo-

nents Fig. 1(d). The aim of autoencoder is to retrieve set of features which are the best representation of original data without redundancy. In the next section 3, the feature reduction step is explained.

3. AUTOENCODER

An autoencoder is a neural network that is trained to reconstruct the input data into the output with minimum amount of reconstruction error. They are designed in a way not to copy the input but to learn important and unique feature of the input data. Autoencoders are mainly used for pre-train deep networks dimensionality reduction, feature learning and generative modeling of data. It is composed of two main parts, input layer and output layer together with hidden layer connecting the two layers. The input layer has the same number of nodes as the output layer. To build an autoencoder we need encoding function at the input, decoding function at the output and loss function to calculate the amount of information loss between encoded representation and decoded representation of the input and output data respectively.

3.1. Encoder

It maps an input vector $x \in \mathbb{R}^n$, into encoded representation $h(x) \in \mathbb{R}^m$. The typical form is affine mapping followed by nonlinearity (eq. 1). The parameter is set $\Theta = (w, b)$ where w is weight matrix of size $m \times n$ and $b \in \mathbb{R}^m$ is bias vector, f is activation function.

$$h_{\Theta} = f(wx + b) \quad (1)$$

3.2. Decoder

It maps the resulting encoded representation $h(x)$ back into an estimate of reconstructed n -dimensional vector $r \in \mathbb{R}^n$, where

$$r_{\phi} = g(f(x)) \quad (2)$$

$$r_\phi = g(w'h + b') \quad (3)$$

The parameter $\phi = \{w', b'\}$, where w' is weight matrix of size is $n \times m$, $b' \in \mathcal{R}^n$ is bias vector and g is activation function of the decoder. The autoencoder tries to learn a function $r_\phi \simeq x$, thus each training data $x(i)$ is mapped to corresponding reconstructed data $r(i)$.

3.3. Loss function

Imposing sparsity constraint on the hidden nodes of an autoencoder enables the model to learn unique statistical features of the data even when the number of hidden units are larger compared to the feature space of the input data. A neuron is considered active if its output value is close to maximum value of the activation function used (close to 1 for sigmoid activation function) and inactive if its output is close to minimum value (0 in case of sigmoid). The average activation of i^{th} hidden unit $\hat{\rho}_i$ (eq. 4), where n is total number of inputs, x_j is the j^{th} training example and h_i activation of j^{th} hidden unit is given as:

$$\hat{\rho}_i = \frac{1}{n} \sum_{j=1}^n h_i(x_j) \quad (4)$$

Choosing sparsity parameter ρ small ($\rho=0.01$) and imposing $\hat{\rho}_i = \rho$ constraint, Kullback-Leibler divergence term is applied to penalize $\hat{\rho}_i$. Penalty value that diverges from ρ will give reasonable result.

$$KL(\rho || \hat{\rho}_i) = \rho \log\left(\frac{\rho}{\hat{\rho}_i}\right) + (1 - \rho) \log\left(\frac{1 - \rho}{1 - \hat{\rho}_i}\right) \quad (5)$$

where $KL(\rho || \hat{\rho}_i)$ is the Kullback-Leibler divergence between Bernoulli random variable with ρ mean and Bernoulli random variable with mean $\hat{\rho}_i$. If $\rho = \hat{\rho}_i$, $KL(\rho || \hat{\rho}_i) = 0$, else increases monotonically as $\hat{\rho}_i$ diverges from ρ . Minimizing the KL divergence penalty term leads to $\hat{\rho}_i$ to be close to ρ . The overall loss function will be as the following:

$$L(\Theta, \Omega) = \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K (x_{kn} - r_{kn})^2 + \lambda * \Omega_{weights} + \beta * \Omega_{sparsity}$$

where $\Omega_{sparsity}$ is sparsity regularizer and calculated as eq. 6:

$$\Omega_{sparsity} = \sum_{i=1}^m KL(\rho || \hat{\rho}_i) \quad (6)$$

Similarly, $\Omega_{weights}$ is $L2$ regularization term. It's task is to avoid overfitting by penalizing the rate in which the model reacts to changes in the training example distribution and forcing the model to learn most significant features. It is calculated as eq. 7

$$\Omega_{weights} = \frac{1}{2} \sum_l^L \sum_j^N \sum_i^K (w_{ij}^l)^2 \quad (7)$$

where L is number of hidden layers, N is the number of training examples and K is the number of features. In the lost function, λ is coefficient for the $L2$ regularization term $\Omega_{weights}$ and β is the coefficient for sparsity regularization $\Omega_{sparsity}$ term.

3.4. Training

Once the model is setup, our goal is to minimize the cost function $L(\Theta, \Omega)$ as a function of weights w and bias b . To train our autoencoder neural network, we initialized each parameter $w_{i,j}^{(l)}$, and $b_i^{(l)}$ each to a small random value near zero ($\mathcal{N}(0, \varepsilon^2)$ distribution for a small ε), and then apply stochastic conjugate gradient decent (SCG) algorithms to learn the network parameters of autoencoder. Random initialization is necessary, if all the parameters start off at identical values, then all the hidden layer units will end up learning the same function of the input.

3.4.1. Pareto Based Multi Objective Learning (PMOL)

The choice on model complexity and mean squared error (MSE) are two trade-offs to optimize. As we increase the number of hidden nodes MSE decreases. However, the increase in the number of hidden-neurons might lead to overfitting the data, thus decreasing the generalization performance of the model. The goal is to find an optimal solution for both model complexity and MSE that maximizes the model performance. We exploited PMOL for estimating the optimal number of parameters for our autoencoder. It uses vector of objective functions and therefore number of optimal solutions are more than one. Pareto front of optimal solution is a set of non-dominated solutions, being chosen as optimal, if no objective can be improved without sacrificing at least one other objective. On the other hand a solution X is referred to as dominated by another solution Y if, and only if, Y is equally good or better than X with respect to all objectives. Pareto based multi object optimization can be formulated as eq. 8 with Q objective functions;

$$f(p) = [f_i(p), i = 1, \dots, Q] \quad (8)$$

subjected to the equality constraints

$$g_j(p) = 0 \quad j = 1, 2, \dots, J \quad (9)$$

And the K inequality constraints

$$h_k(p) \leq 0 \quad k = 1, 2, \dots, K \quad (10)$$

The aim is to find vector p^* which minimizes $f(p)$, in our case since we have two objective function thus pareto based bi-objective learning problem can be formulated to minimize the two objectives, that is data fitting term and model complexity term given by eq. 11

$$f_1 = -\mathcal{L}(E|\Theta), \quad f_2 = \gamma k \log(L) \quad (11)$$

| Methods | Orange juice | Wine |
|-----------------------|--------------|--------|
| Douak et al. [15] | 01574 | 0.0070 |
| Benoudjit et al. [14] | 0.2435 | 0.0052 |
| Alhichri et al. [25] | 0.32076 | 0.0034 |
| Ours | 0.1711 | 0.0042 |

Table 1: Quantitative results of our method against 3 state-of-the-art methods. Numeric values shows Normalized Mean Square Error (NMSE), the lower is NMSE, the better is performance.

where $f_1 = -\mathcal{L}(E|\Theta)$ is data fitting objective function and $f_2 = \gamma k \log(L)$ is model complexity objective function. f_1 is log-likelihood function that is found with a maximum likelihood estimation algorithm. $E = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_L)$ is a set of multi-dimensional reconstruction error. Assuming the error is multivariate normal distribution with Mean vector $M = 0$ and covariance Σ , then the distribution function is given by:

$$p(\varepsilon) = \mathcal{N}(M, \Sigma) = \frac{1}{2\pi \sqrt{|\Sigma n_f^2|}} \exp\left(-\frac{1}{2} \varepsilon^T \Sigma^{-1} \varepsilon\right) \quad (12)$$

where the negative log-likelihood function will be represented by

$$-\mathcal{L}(E|\Theta) = \log \prod_{i=1}^L p(\varepsilon_i) \quad (13)$$

γ is a constant determined by a pareto optimizer, L is input training sample size, k is the number of parameters of the model to be estimated (weight and bias).

4. GAUSSIAN PROCESS REGRESSION

Gaussian process regressor (GPR) is a powerful non-parametric technique with precise uncertainty model used in regression problems. Instead of finding correct parameters to fit for basis function, GPR focus on deducing the correlation in all the measured data. A Gaussian process is a Gaussian random function that is characterized by mean and covariance functions, $m(x)$ and $k(x, x')$:

$$f(x) \sim GP(m(x), k(x, x')) \quad (14)$$

A detailed description GPR is beyond the scope of this paper. Interested readers may refer to [26]. We used GPR in this work because our data is non-linear and GPR fits very well for our problem.

5. EXPERIMENTS

In this work we have used two spectrophotometric datasets, coming from the food industry. The first dataset deals with determining sugar content in the orange juice sample by near infrared reflectance spectrometry [27]. The training and test samples in the orange juice data are 149 and 67 respectively; with 700 spectral variables that represents the absorbance ($\log 1/R$) between 1100nm and 2500nm. The value of R represents light reflected by the sample. The concentration of sugar ranges from 0% to 95.2% by weight in the sample. The second dataset deals with the determination of alcohol content by mid-infrared spectroscopy in wine samples [27]. The training and test data sets contain 91 and 30 spectra, respectively, with 256 spectral variables that are the absorbance ($\log 1/T$) at 256 wave numbers between 4000 and 400 cm^{-1} (where T is the light transmittance through the sample thickness). Alcohol content varies from 7.48% to 15.5% by volume. No preprocessing has been performed on the orange juice and wine datasets.

For the quantitative results, the accuracy of the approach is represented in normalized mean square error (NMSE) metric, which is define as:

$$NMSE = \frac{1}{V_{train+test}} \sum_{i=1}^M (y_{itest} - y'_{itest})^2 \quad (15)$$

where M represents the number of testing samples, y_{itest} and y'_{itest} are the real and estimated outputs for the i^{th} test sample x_{itest} and $V_{train+test}$ is the combined variance of the training and test output samples y_{itrain} and y_{itest} . The results are listed in Table ?? . It can be seen that our approach gives good results on both the datasets compare to other state-of-the-art methods.

6. CONCLUSION

We propose a novel deep learning based chemometric data analysis technique for estimating sugar and alcohol concentration in orange juice and wine samples. L2 regularized sparse autoencoder is trained end-to-end for reducing the size of the feature vector to handle the classic problem of curse of dimensionality in chemometric data analysis. The optimal set of parameters of the autoencoder are selected through pareto optimization. And Gaussian process regressor is applied on the reduced size feature vector for estimating the concentration of sugar and alcohol in the corresponding samples. The method is compared against 3 state-of-the-art methods. Quantitative results are shown on Normalized Mean Square Error (NMSE) and our approach shows better results on both datasets. In future, we are planning to test our approach on more datasets. Moreover, we are also planning to use stacked-autoencoder for feature reduction. Furthermore, we would apply different regressor like linear, support vector and ker-

nel support vector regressor for estimating the corresponding quantities.

7. REFERENCES

- [1] Svante Wold, "Chemometrics; what do we mean with it, and what do we want from it?," *Chemometrics and Intelligent Laboratory Systems*, vol. 30, no. 1, pp. 109–115, 1995.
- [2] Dominique Bertrand, "La spectroscopie proche infrarouge et ses applications dans les industries de l'alimentation animale," *Productions Animales 3 (15), 209-219.(2002)*, 2002.
- [3] Jon M Sutter and John H Kalivas, "Comparison of forward selection, backward elimination, and generalized simulated annealing for variable selection," *Microchemical journal*, vol. 47, no. 1-2, pp. 60–66, 1993.
- [4] David A Belsley, Edwin Kuh, and Roy E Welsch, *Regression diagnostics: Identifying influential data and sources of collinearity*, vol. 571, John Wiley & Sons, 2005.
- [5] Tomas Eklöv, Per Mårtensson, and Ingemar Lundström, "Selection of variables for interpreting multivariate gas sensor data," *Analytica Chimica Acta*, vol. 381, no. 2-3, pp. 221–232, 1999.
- [6] Paul Geladi, "Some recent trends in the calibration literature," *Chemometrics and Intelligent Laboratory Systems*, vol. 60, no. 1-2, pp. 211–224, 2002.
- [7] Tormod Næs and Bjørn-Helge Mevik, "Understanding the collinearity problem in regression and discriminant analysis," *Journal of Chemometrics*, vol. 15, no. 4, pp. 413–426, 2001.
- [8] Michel Verleysen et al., "Learning high-dimensional data," 2003.
- [9] John A Cornell, "Classical and modern regression with applications," 1987.
- [10] Nabil Benoudjit, E Cools, Marc Meurens, and Michel Verleysen, "Chemometric calibration of infrared spectrometers: selection and validation of variables by non-linear models," *Chemometrics and Intelligent Laboratory Systems*, vol. 70, no. 1, pp. 47–53, 2004.
- [11] Nabil Benoudjit, Damien François, Marc Meurens, and Michel Verleysen, "Spectrophotometric variable selection by mutual information," *Chemometrics and Intelligent Laboratory Systems*, vol. 74, no. 2, pp. 243–251, 2004.
- [12] Fabrice Rossi, Amaury Lendasse, Damien François, Vincent Wertz, and Michel Verleysen, "Mutual information for the selection of relevant variables in spectrometric nonlinear modelling," *Chemometrics and intelligent laboratory systems*, vol. 80, no. 2, pp. 215–226, 2006.
- [13] Svante Wold, Kim Esbensen, and Paul Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.
- [14] N Benoudjit, F Melgani, and H Bouzgou, "Multiple regression systems for spectrophotometric data analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 95, no. 2, pp. 144–149, 2009.
- [15] F Douak, N Benoudjit, and F Melgani, "A two-stage regression approach for spectroscopic quantitative analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 109, no. 1, pp. 34–41, 2011.
- [16] Mohib Ullah and Faouzi Alaya Cheikh, "Deep feature based end-to-end transportation network for multi-target tracking," in *IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 3738–3742.
- [17] Ahmed Mohammed, Sule Yildirim, Marius Pedersen, Øistein Hovde, and Faouzi Cheikh, "Sparse coded handcrafted and deep features for colon capsule video summarization," in *IEEE International Symposium on Computer-Based Medical Systems (CBMS)*, 2017, pp. 728–733.
- [18] Habib Ullah, Muhammad Uzair, Mohib Ullah, Asif Khan, Ayaz Ahmad, and Wilayat Khan, "Density independent hydrodynamics model for crowd coherency detection," *Neurocomputing*, vol. 242, pp. 28–39, 2017.
- [19] Habib Ullah, *Crowd Motion Analysis: Segmentation, Anomaly Detection, and Behavior Classification*, Ph.D. thesis, University of Trento, 2015.
- [20] Ahmed Mohammed, Yildirim Sule, Farup Ivar, Pedersen Marius, and Øistein Hovde, "Y-net: A deep convolutional neural network for polyp detection," in *British Machine Vision Conference (BMVC)*. BMVA, 2018.
- [21] Habib Ullah, Mohib Ullah, and Muhammad Uzair, "A hybrid social influence model for pedestrian motion segmentation," *Neural Computing and Applications*, pp. 1–17, 2018.
- [22] Habib Ullah, Ahmed B Altamimi, Muhammad Uzair, and Mohib Ullah, "Anomalous entities detection and localization in pedestrian flows," *Neurocomputing*, vol. 290, pp. 74–86, 2018.
- [23] Mohib Ullah, Ahmed Kedir Mohammed, Faouzi Alaya Cheikh, and Zhaohui Wang, "A hierarchical feature model for multi-target tracking," in *IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 2612–2616.
- [24] Pierre Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, 2012, pp. 37–49.
- [25] Haikel AlHichri, Yakoub Bazi, Naif Alajlan, Farid Melgani, Salim Malek, and Ronald R Yager, "A novel fusion approach based on induced ordered weighted averaging operators for chemometric data analysis," *Journal of Chemometrics*, vol. 27, no. 12, pp. 447–456, 2013.
- [26] Carl Edward Rasmussen, "Gaussian processes in machine learning," in *Advanced lectures on machine learning*, pp. 63–71. Springer, 2004.
- [27] Marc Meurens, "Spectrophotometric samples of orange juice and wine for chemometric analysis," .

JOIN US AT THE NEXT EI!

IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!



- **SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •**
- **INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •**

www.electronicimaging.org

