Application of Artificial Neural Networks to Predict the Impact of Traffic Emissions on Human Health

Tânia Fontes[1], Luís M. Silva[2,3], Sérgio R. Pereira[1], and Margarida C. Coelho[1]

[1] Centre for Mechanical Technology and Automation / Department of Mechanical Engineering, University of Aveiro, Aveiro, Portugal
[2] Department of Mathematics, University of Aveiro, Aveiro, Portugal
[3] INEB-Instituto de Engenharia Biomédica, Campus FEUP (Faculdade de Engenharia da Universidade do Porto), Rua Dr. Roberto Frias, s/n, 4200-465 Porto, Portugal
{trfontes,lmas,sergiofpereira,margarida.coelho}@ua.pt

Abstract. Artificial Neural Networks (ANN) have been essentially used as regression models to predict the concentration of one or more pollutants usually requiring information collected from air quality stations. In this work we consider a Multilayer Perceptron (MLP) with one hidden layer as a classifier of the impact of air quality on human health, using only traffic and meteorological data as inputs. Our data was obtained from a specific urban area and constitutes a 2-class problem: above or below the legal limits of specific pollutant concentrations. The results show that an MLP with 40 to 50 hidden neurons and trained with the cross-entropy cost function, is able to achieve a mean error around 11%, meaning that air quality impacts can be predicted with good accuracy using only traffic and meteorological data. The use of an ANN without air quality inputs constitutes a significant achievement because governments may therefore minimize the use of such expensive stations.

Keywords: neural networks, air quality level, traffic volumes, meteorology, human health protection.

1 Introduction

Artificial Neural Networks (ANN) are powerful tools inspired in biological neural networks and with application in several areas of knowledge. They have been commonly used to estimate and/or forecast air pollution levels using pollutant concentrations, meteorological and traffic data as inputs. Viotti et al. (2002) proposed an ANN approach to estimate the air pollution levels in 24-48 hours for sulphur dioxide (SO_2), nitrogen oxides (NO, NO_2, NO_X), total suspended particulate (PM_{10}), benzene (C_6H_6), carbon monoxide (CO) and ozone (O_3). Since then, a lot of studies have been made based on the prediction of one or more pollutants. Nagendra and Khare (2005) demonstrate that ANN can explain with accuracy the effects of traffic on the CO dispersion. Zolghadri and Cazaurand (2006) predict the average daily concentrations

L. Correia, L.P. Reis, and J. Cascalho (Eds.): EPIA 2013, LNAI 8154, pp. 21–29, 2013.

of PM_{10} but to improve the results they suggest the use of traffic emissions data. On the other hand, Chan and Jian (2013) demonstrate the viability of ANN to estimate $PM_{2.5}$ and PM_{10} concentrations. They verified that the proposed model can accurately estimate not only the air pollution levels but also to identify factors that have impact in those air pollution levels. Voukantsis et al. (2011) also analyzed the $PM_{2.5}$ and PM_{10} concentrations. However, they propose a combination between linear regression and ANN models for estimating those concentrations. An improved performance in forecasting the air quality parameters was achieved by these authors when compared with previous studies. The same conclusions were obtained in a study conducted by Slini et al. (2006) to forecast the PM_{10} concentrations. Nonetheless, Slini et al. (2006) stress the need to improve the model by including more parameters like wind profile, opacity and traffic conditions. Cai et al. (2009) uses an ANN to predict the concentrations of CO, NO_x, PM and O_3. The ANN predicts with accuracy the hourly air pollution concentrations with more than 10 hours in advance. Ibarra-Berastegi et al. (2008) make a more embracing analysis and proposes a model to predict five pollutants (SO_2, CO, NO, NO_2, and O_3) with up to 8 hours ahead.

Literature review shows that ANN have been essentially used as regression models using pollutant concentrations in two ways: (i) first, as input variables to the model; (ii) and second, as variable(s) to be predicted (usually with up to *h* hours ahead). This means that information of such concentrations has to be collected, limiting the applicability of such models only to locals where air quality stations exist. In this work we propose two modifications. First, we rely just on meteorological and traffic data as input variables to the ANN model, eliminating the use of pollutant concentrations and consequently, the need for air quality stations. Second, we use an ANN as a classifier (and not as a regression model) of the air quality level (below or above to the human health protection limits, as explained in the following section). Such a tool will provide the ability to predict the air quality level in any city regardless of the availability of air quality measurements.

2 Material and Methods

2.1 The Data

Hourly data from 7 traffic stations, a meteorological station and an air quality station, located in a congested urban area of Oporto city (Portugal), were collected for the year 2004 (Figure 1). Traffic is monitored with sensors located under the streets and the meteorological station (at 10’23.9’’N and 8º 37’’21.6’’W) was installed according to the criteria of the World Meteorological Organization (WMO, 1996). Table 1 presents the traffic and meteorological variables used as inputs in the ANN model while Table 2 shows the air quality pollutants measured as well as the respective limits of human health protection according to the Directive 2008/50/EC. These pollutant

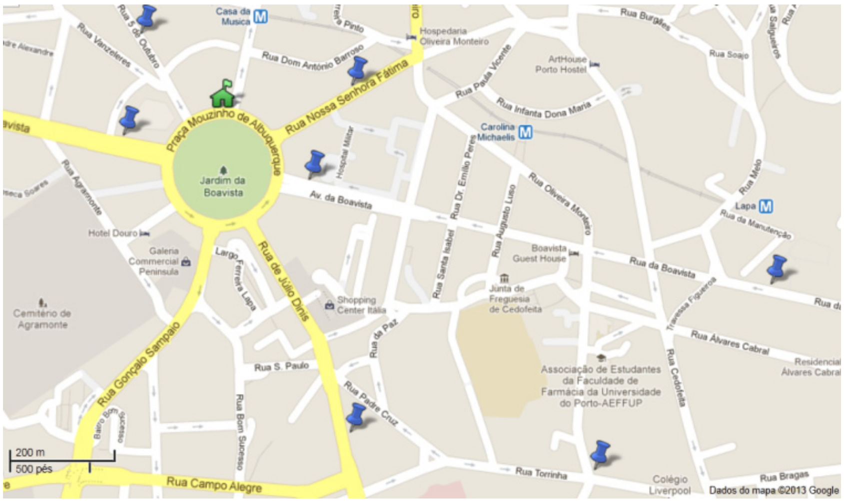

Fig. 1. Study domain (Oporto - Boavista zone): traffic station; air quality station. The meteorological station is located 2 km away from this area.

concentrations are just used to build our classification problem as follows: an instance belongs to class 1 if all the pollutant concentrations are below those limits and to class 2 otherwise.

For the year 2004, is expected that 8,784 hourly instances with data from all of the stations were available. However, due to the gaps and missing data existing in the historical records and also those produced after data pre-processing, the total number of available instances for this year is 3,469. Thus, we have a two-class problem organized in a data matrix with 3,469 lines corresponding to the number of instances and 8 columns corresponding to the 7 input variables and 1 target variable, coded 0 (class 1) and 1 (class 2). Table 3 shows the sampling details of each meteorological and air quality variables.

Table 1. Input variables for the ANN

Variables		Abbreviation	Units
Hour		H	-
Month		M	-
Traffic volumes		V	Vehicles
Meteorological variables	Wind speed	WS	m/s
	Wind direction	WD	º
	Temperature	T	ºC
	Solar radiation	SR	W/m^2

Table 2. Air quality limits of human health protection according to Directive 2008/50/EC

	Abbreviation	Units	Time reference	Human health limit protection
Nitrogen Dioxide	NO_2	µg/m^3	Hourly	200
Carbon Monoxide	CO	µg/m^3	Octo-hourly	10,000
Particles	PM_{10}	µg/m^3	Daily	50
Ozone	O_3	µg/m^3	Hourly	180
Sulphur Dioxide	SO_2	µg/m^3	Hourly	350

Table 3. Details of the monitoring instruments

Variables		Reaction time	Accuracy	Range	Technique
Meteo-rology	WS	n.a.	±0.1 m/s	0-20 m/s	Anemometer
	WD	n.a.	±4°	0-360°	Anemometer
	T	30-60 s	±0.2 °C	−50 to 70 °C	Temperature sensor
	SR	10 µs	±1%	0-3,000 W.m^{-2}	Pyranometer
Air quality	NO_2	< 5s	n.a.	0-500 µg.m^{-3}	Chemiluminescence analyzer
	CO	30 s	1%	0 ~ 0.05-200 ppm	Infrared photometry
	PM_{10}	10-30 s	n.a.	0 ~ 0.05-10,000 µg/m^3	Beta radiation
	O_3	30 s	1.0 ppb	0-0.1~10 ppm	UV photometric
	SO_2	10 s	0.4 ppb	0-0.1	UV fluorescence

2.2 The Model

We considered the most common architecture of an ANN, the Multilayer Perceptron (MLP). In general, an MLP is a nonlinear model that can be represented as a stacked arrangement of layers, each of which is composed of processing units, also known as neurons (except for the input layer, which has no processing units). Each neuron is connected to all the neurons of the following layer by means of parameters (weights) and computes a nonlinear signal of a linear combination of its inputs. Each layer serves as input to the following layer in a forward basis. The top layer is known as output layer (the response of the MLP) and any layer between the input and output layer is called hidden layer (its units are designated hidden neurons). In this work we restricted to the case of a single hidden layer. In fact, as Cybenko (1989) shows, one hidden layer is enough to approximate any function provided the number of hidden neurons is sufficient. The MLP architecture is 7: n_{hid}:1, that is, it has 7 inputs, corresponding to the 7 variables described in Table 1 (in fact, normalized versions of those variables, see Section 2.3), n_{hid} hidden neurons and one output neuron. Figure 2 depicts the architecture used.

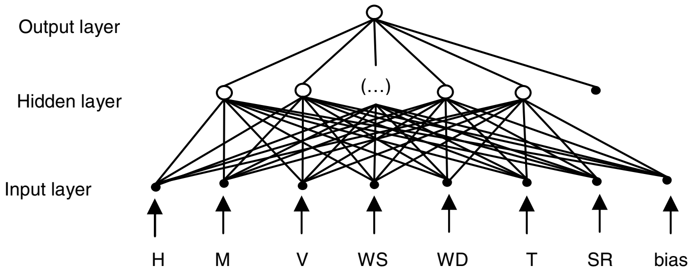

Fig. 2. MLP architecture used: 7- n_{hid}-1 with n_{hid} varying from 5 to 50 in steps of 5

In a formal way, the model can be expressed as:

$$\begin{aligned} \mathrm{y} &= \varphi_2\left(\sum_{j=1}^{n_{hid}} w_j^{(2)} h_j + b^{(2)}\right) \\ &= \varphi_2\left(\sum_{j=1}^{n_{hid}} w_j^{(2)} \varphi_1\left(\sum_{k=1}^{7} w_{kj}^{(1)} x_k + b_j^{(1)}\right) + b^{(2)}\right), \end{aligned} \quad (1)$$

where:

$w_j^{(2)}$ – Weight connecting hidden neuron *j* to the output neuron;
h_j – Output of hidden neuron *j*;
$b^{(2)}$ – Bias term connected to the output neuron;
$w_{kj}^{(1)}$ – Weight connecting input *k* to hidden neuron *j*;
x_k – *k*-th input variable;
$b_j^{(1)}$ – Bias term connected to hidden neuron *j*;
n_{hid} – Number of hidden neurons

and φ_1 and φ_2 are the hyperbolic tangent and sigmoid activation functions, respectively, responsible for the non-linearity of the model:

$$\varphi_1(\mathrm{a}) = \frac{e^{2a} - 1}{e^{2a} + 1} \quad (2)$$

$$\varphi_2(\mathrm{a}) = \frac{1}{1 + e^{-a}} \quad (3)$$

Parameter (weights) optimization (also known as learning or training) is performed by the batch backpropagation algorithm (applying the gradient descent optimization), through the minimization of two different cost functions: the commonly used Mean Square Error (MSE) and the Cross-Entropy (CE) (Bishop, 1995), expressed as

$$\mathrm{MSE} = \frac{1}{n}\sum_{\mathrm{i=1}}^{\mathrm{n}}(\mathrm{t_i} - \mathrm{y_i})^2, \tag{4}$$

$$\mathrm{CE} = -\sum_{\mathrm{i=1}}^{\mathrm{n}} y_i \log(y_i) + (1 - y_i)\log(1 - y_i). \tag{5}$$

Here, n is the number of instances, $t_i \in \{0,1\}$ is the target value (class code) for instance i and $y_i \in [0,1]$ is the network output for instance i. The MLP predicts class 1 (code 0) whenever $y_i \leq 0.5$ and class 2 (code 1) otherwise. We have also used an adaptive learning rate with an initial value of 0.001 (Marques de Sá et al., 2012).

For a comprehensive approach on ANN please refer to Bishop (1995) or Haykin (2009).

2.3 Experimental Procedure

The experimental procedure was as follows. For each number n_{hid} of hidden neurons tested, we performed 30 repetitions of:

1. Randomization of the whole dataset;
2. Split in training, validation and test sets (50%, 25% and 25% respectively of the whole dataset) maintaining class proportions;
3. Normalization of these sets, such as to have inputs with zero mean and unit standard deviation (validation and test sets are normalized using the parameters of the training set);
4. Training the MLP (initialized with small random weights) during 50,000 epochs.

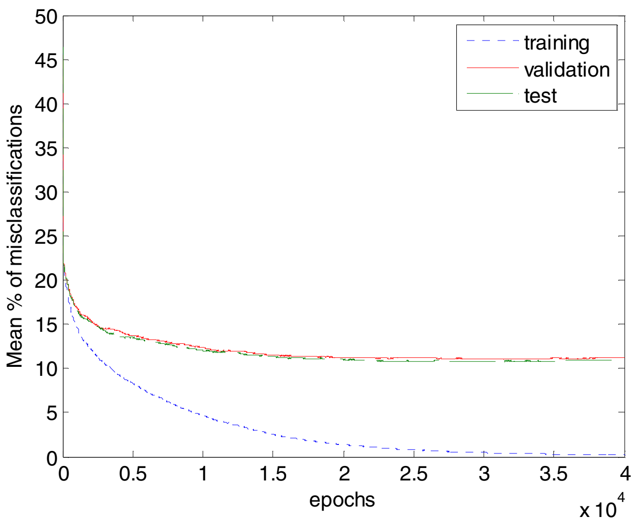

Fig. 3. Training, validation and test set mean misclassifications during the learning process of an MLP with 45 hidden neurons and using the cross-entropy cost

To assess the accuracy of the trained models, we keep track of the training, validation and test set misclassifications during the learning process. Graphs such as the one shown in Figure 3 where produced and used as follows: the validation error is used to perform "early stopping" by choosing the number of epochs m where its mean error is minimum. The mean misclassification error and standard deviation over the 30 repetitions at epoch m are then recorded for the training, validation and test sets (see Section 3). Computations were performed using MATLAB (MathWorks, 2012).

3 Results and Discussion

Table 4 presents the estimates of the mean (over 30 repetitions) misclassification errors and standard deviations both for the MSE and CE cost functions. As explained in the previous section, these records correspond to the epoch number (also given in Table 4) where the mean validation set error was smaller.

Table 4. Mean misclassification errors (in %) for different number of hidden neurons and different cost functions

n_{hid}	MSE				CE			
	Epochs (m)	Mean error (standard deviation)			Epochs	Mean error (standard deviation)		
		Train	Validation	Test		Train	Validation	Test
5	49,900	33,00(4.32)	33.97(3.82)	33.63(4.29)	31,800	18.39(0.81)	19.30(1.53)	19.46(1.47)
10	50,000	22.49(2.15)	25.29(2.01)	25.74(2.55)	25,700	15.05(0.87)	17.77(1.58)	16.96(1.23)
15	50,000	14.59(2.28)	19.46(1.53)	19.60(2.90)	10,300	12.70(0.72)	16.47(1.53)	16.07(1.14)
20	50,000	10.18(1.74)	17.10(1.55)	17.06(1.38)	35,600	9.85(0.89)	15.34(1.35)	15.40(1.29)
25	50,000	7.06(1.34)	15.54(1.15)	14.94(1.46)	19,000	7.93(0.77)	13.99(1.31)	14.40(1.33)
30	49,900	4.56(0.72)	13.78(1.21)	13.47(0.95)	21,400	5.28(0.80)	13.07(1.44)	13.03(1.03)
35	49,700	4.25(1.13)	13.70(1.57)	13.84(1.08)	40,000	1.83(0.68)	11.43(1.13)	11.88(0.08)
40	49,800	9.76(1.17)	13,67(1.63)	12.86(1.12)	23,700	1.64(1.06)	11.33(0.97)	11.28(0.08)
45	47,800	9.79(0.63)	13.27(1.30)	13.46(1.32)	30,000	4.60(0.38)	11.07(1.13)	10.77(1.05)
50	49,700	9.47(0.57)	13.08(1.07)	13.14(1.28)	30,500	3.70(0.36)	11.02(1.22)	10.68(1.08)

At a first glance we may observe that the use of different cost functions has important impacts in the results. In fact, CE revealed to be a more efficient cost function in the sense that for the same architectures, a better generalization error is achieved with the use of fewer training epochs when compared to MSE. This is in line to what is known about CE for which several authors reported marked reductions on convergence rates and density of local minima (Matsuoka and Yi, 1991; Solla et al., 1988).

Experimental results show that an MLP with 40 to 50 hidden neurons and trained with CE is able to achieve a mean error around 11%. The standard deviations are also small showing a stable behavior along the 30 repetitions. This suggests that the air quality level can be predicted with good accuracy using only traffic and meteorological data. This is significant as governments may therefore minimize the use of expensive air quality stations.

If the algorithm is applied in a densely urban network of traffic counters the air quality levels could be quickly obtained with a high spatial detail. This represents an important achievement because the implementation of this tool can contribute to assess the potential benefits of the introduction of an actuation plan to minimize traffic emissions in a real-time basis.

4 Conclusions

In this paper, a multilayer perceptron with one hidden layer was applied to automate the classification of the impact of traffic emissions on air quality considering the human health effects. We found that with a model with 40 to 50 hidden neurons and trained with the cross-entropy cost function, we may achieve a mean error around 11% (with a small standard deviation) which we can consider as a good generalization. This demonstrates that such a tool can be built and used to inform citizens in a real time basis. Moreover, governments can better assess the potential benefits of the introduction of an actuation plan to minimize traffic emissions as well as reducing costs by minimizing the use of air quality stations. Future work will be focused on the use of more data from urban areas, as well as data from other environmental types like suburban and rural areas. We will also seek for accuracy improvements by applying other learning algorithms to this data, such as support vector machines and deep neural networks.

Acknowledgements. This work was partially funded by FEDER Funds through the Operational Program "Factores de Competitividade – COMPETE" and by National Funds through FCT–Portuguese Science and Technology Foundation within the projects PTDC/SEN-TRA/115117/2009 and PTDC/EIA-EIA/119004/2010. The authors from TEMA also acknowledge the Strategic Project PEst-C/EME/UI0481/2011.

References

1. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press (1995)
2. Cai, M., Yin, Y., Xie, M.: Prediction of hourly air pollutant concentrations near urban arterials using artificial neural network approach. Transportation Research Part D: Transport and Environment 14, 32–41 (2009)
3. Chan, K.Y., Jian, L.: Identification of significant factors for air pollution levels using a neural network based knowledge discovery system. Neurocomputing 99, 564–569 (2013)
4. Cybenko, G.: Aproximation by superpositios of a sigmoidal function. Math. Control Signals System 2, 303–314 (1989)
5. Directive 2008/50/EC: European Parliament and of the Council of 21 May 2008 on ambient air quality and cleaner air for Europe entered into force on 11June 2008
6. Haykin, S.: Neural Networks and Learning Machines, 3rd edn. Prentice Hall (2009)

7. Ibarra-Berastegi, G., Elias, A., Barona, A., Saenz, J., Ezcurra, A., Argandona, J.D.: From diagnosis to prognosis for forecasting air pollution using neural networks: Air pollution monitoring in Bilbao. Environmental Modelling & Software 23, 622–637 (2008)
8. Marques de Sá, J., Silva, L.M., Santos, J.M., Alexandre, L.A.: Minimum Error Entropy Classification. SCI, vol. 420. Springer (2012)
9. MathWorks, MATLAB and Statistics Toolbox Release 2012, The MathWorks, Inc., Natick, Massachusetts, United States (2012)
10. Matsuoka, K., Yi, J.: Backpropagation based on the logarithmic error function and elimination of local minima. In: Proceedings of the 1990 IEEE International Joint Conference on Neural Networks (1991)
11. Nagendra, S.S.M., Khare, M.: Modelling urban air quality using artificial neural network. Clean Techn. Environ. Policy 7, 116–126 (2005)
12. Slini, T., Kaprara, A., Karatzas, K., Moussiopoulos, N.: PM10 forecasting for Thessaloniki, Greece. Environmental Modelling & Software 21, 559–565 (2006)
13. Solla, S., Levin, E., Fleisher, M.: Accelerated learning in layered neural networks. Complex Systems 2(6), 625–639 (1988)
14. Viotti, P., Liuti, G., Di, P.: Atmospheric urban pollution: applications of an artificial neural network (ANN) to the city of Perugia. Ecological Modelling 148, 27–46 (2002)
15. Voukantsis, D., Karatzas, K., Kukkonen, J., Rasanen, T., Karppinen, A., Kolehmainen, M.: Intercomparison of air quality data using principal component analysis, and forecasting of PM10 and PM2.5 concentrations using artificial neural networks. Thessaloniki and Helsinki, Science of The Total Environment 409, 1266–1276 (2011)
16. Zolghadri, A., Cazaurang, F.: Adaptive nonlinear state-space modeling for the prediction of daily mean PM10 concentrations. Environmental Modelling & Software 21, 885–894 (2006)
17. WMO, Guide to meteorological instruments and methods of observation, 6th edn., World Meteorological Organization, No. 8 (1996)