Correlation and prediction of saline solution properties for their use in mineral processing using artificial neural networks

José O. Valderrama, Richard A. Campusano and Alvaro S. Toro

ABSTRACT

Artificial neural networks (ANNs) are applied to correlate and predict physico-chemical, transport and thermodynamic properties of seawater. Values of these properties are needed in the design, simulation and optimization of processes in which seawater is used, mainly in the mining industry. Density, vapor pressure, boiling temperature elevation, specific heat, viscosity, thermal conductivity, surface tension, osmotic coefficient, enthalpy, entropy and latent heat of vaporization are analyzed. These properties depend on temperature and salt content in the saline solution, so these are the independent variables considered for the training and testing of the ANN. Several network architectures were considered and correlated, and predicted values of these properties were compared with values obtained from the literature. As a measure of the accuracy of the method, the average deviation and the average absolute deviation are evaluated. The ANN model obtained gave lower deviations than other more sophisticated models presented in the literature. The chosen ANN model gave absolute deviations lower than 0.5%, with a few exceptions, but maximum deviations were always below 1.0% for all properties.

Key words | ANN, density, seawater properties, thermodynamic properties, transport properties, viscosity

INTRODUCTION

Water scarcity is nowadays one of the main challenging problems in industry in general, and is of especial relevance in water-consuming processes in the mining industry (mineral concentration, hydrometallurgy, refining processes). Also, mining operations are usually located where water is extremely scarce. In most mining processes part of the water used is treated, purified and recirculated, but fresh water, which is rarely available, is needed in high amounts. Considering this adverse situation, most new mining projects include the use of seawater or partially desalinated water to supply part of their operations (Singh 2010).

Different types of salts present in seawater may affect processes in different forms, so in some cases one type of salt or another will have different effects on a given process. The overall concentration of salts in seawater varies between narrow ranges (34–37 g of salts per kilogram of solution), and partial or total desalination may be needed. During these desalination processes, temperature and salt concentration vary in a much larger range and properties also change. Therefore, it seems of especial importance to evaluate the properties of saline waters at different salt concentrations and temperatures (Valderrama et al. 2014a).

Highly accurate values of properties of seawater are needed in oceanographic studies, in which small differences in a property could lead to erroneous results and conclusions about certain phenomena occurring in the sea. Accurate density values are necessary in determining density fields, since density plays an important role in determining where water of the oceans will flow. For industrial applications such as those mining processes mentioned...
above, or other classical unit operations such as pumping, fluid flow, evaporation, flotation, leaching, reverse osmosis, solvent extraction and crystallization, very accurate estimates are not necessary, but still values must be within acceptable engineering deviations (Harg 1985).

The estimation of physical, chemical and thermodynamic properties of saline solutions is not an easy problem to solve. Water from different places has different types of salts at different concentrations, and during processes of purification of such waters or during processes in which these waters are used, salt types and salt concentrations may vary. Thus, the calculation of physical, transport and thermodynamic properties of saline solutions becomes complex. Fortunately, some simplifications must be introduced for engineering calculations. The definition of a global salinity, for instance, is one of the major contributions in this area and has proved to work well in process design calculations and in process simulation (Lewis & Perkin 1981; Valderrama & Campusano 2015).

Experimental values of properties of seawater and saline solutions are available in the literature at different ranges of temperature \( T \) and salinity \( S \) and several correlations, applicable in different ranges of \( T \) and \( S \), have been proposed (Sidney 1981; Safarov et al. 2009; Sharqawy et al. 2010). In a recent contribution, the authors explored the use of Padé approximants to model saline water properties (Valderrama & Campusano 2015).

Some years ago Taskinen & Ylír uusi (2003) presented a complete review of the use of artificial neural networks (ANNs) for the estimation of several fluid properties. Properties of seawater were not included in the list, but the study represents a good reference for what has been done and what problems and results were found. A good recent compilation of selected models to estimate the properties of saline solutions has been presented by Sharqawy et al. (2010). The authors analyzed several models for 11 properties and recommended one model for each property, based on its global accuracy. For those models recommended by the authors, deviations with respect to experimental data are very low. Thus, the values calculated by those models can be considered as pseudo-experimental data and are used in this work as the input data for ANN training. To the best of the authors’ knowledge, ANNs have not been used in the form presented here for correlating and predicting properties of seawater and saline solutions. The hypothesis behind this work is that if accurate data is used to train an ANN, the network will be capable of predicting a property at any value of salinity and temperature normally used in processes that use seawater and saline solutions derived from it.

**MODELS AND DATA FOR SEAWATER PROPERTIES**

As described above, there are several models proposed in the literature for different properties at different ranges of temperature and salinity. Although properties vary in a regular, smooth form with these variables, the combined effect (of \( T \) and \( S \)) is different for different properties. This has given origin to the proposal of sophisticated algebraic expressions with high numbers of adjustable parameters, including high degree polynomials, potential functions, and logarithmic functions, among others. Table 1 presents selected models for several properties of seawater, as presented by Sharqawy et al. (2010). The best correlations recommended by these authors were used to generate data for saline solution properties in the ranges given by the authors for each property. These values are considered as pseudo-experimental data and are used to train the ANN. Table 1 summarizes the correlations used for each property expressing the temperature \( T \) in Kelvin and the salinity \( S \) in grams of salt per gram of solution. The conversions between these units and those used by Sharqawy et al. (2010) are provided in Table 2. Table 3 gives the ranges of temperature and salinity for each property considered in the present study.

**ANNs**

An ANN is a computational tool that relates the values of a certain dependent function \( f(x_1, x_2, x_3 \ldots x_n) \) with the values of the independent variables \( x_1, x_2, x_3 \ldots x_n \). To find the relationship, the network is given a set of data \( F \) vs. \( x_1, x_2, x_3 \ldots x_n \) so that finding the relationship between the function and the variables is carried out by training. The form in which the network relates these functions and variables is inspired by the behavior of natural neurons (Bose & Liang 1996). Imaginary units resembling neurons are organized in a certain manner, known as network architecture, formed by several layers, each including a certain number
Table 1  Recommended models for several properties of seawater and saline solutions (adapted from Sharqawy et al. 2010)

Density
\[
\rho_{sw}(\text{kg/m}^3) = 10^{-1}(A_1T_1 + A_2T_2 + A_3T_3 + A_4T_4)B = (1/50) \times (1.998 - 150);
\]

Specific heat
\[
c_{sw}(J/kg K) = A + BT + CT^2 + DT^3
\]

Vapor pressure
\[
L_n(p_{v,sw}) = \frac{a_1}{T} + a_2 + a_3T + a_4T^2 + a_5T^3 + \frac{a_6}{T}L_n(T)
\]

Boiling temperature elevation
\[
BTE(K) = A(S/1.00472) + B(S/1.00472)^2
\]

Dynamic viscosity
\[
\mu_w = \frac{a_1}{T + 208,157} - a_2
\]

Thermal conductivity
\[
\log_{10} \frac{k_{sw}}{0.1} = \log_{10} \left[ a_1 + a_2 \left( \frac{S}{1.00472} \right) \right] \cdot \frac{1}{a_3 + a_4} \cdot \frac{S}{T}
\]

Osmotic coefficient
\[
\phi = a_1 + a_2(T - 273.15) + a_3(T - 273.15)^2 + a_4(T - 273.15)^3 + a_5T
\]

Surface tension
\[
\sigma_{sw} (\text{N/m}) = a_1 \left( \frac{T}{647,096} \right) \cdot \left[ 1 + a_2 \left( \frac{T}{647,096} \right) \right] \cdot \left[ 1 + (a_3(T - 273.15) + a_4(T - 273.15)) \cdot \log(1 + a_5(T - 273.15)) \right] \]

Enthalpy
\[
\frac{h_{sw}(J/kg)}{K} = (a_1 + a_2(T - 273.15) + a_3(T - 273.15)^2 + a_4(T - 273.15)^3)
\]

Entropy
\[
\frac{S_{sw}(J/kg K)}{K} = \frac{S_0(a_1 + a_2(T - 273.15) + a_3(T - 273.15)^2 + a_4(T - 273.15)^3 + a_5(T - 273.15)^4 + a_6(T - 273.15)^5 + a_7(T - 273.15)^6)}{1000}
\]

Heat of vaporization
\[
\frac{h_v}{K} = a_1 + a_2(T - 273.15) + a_3(T - 273.15)^2 + a_4(T - 273.15)^3 + a_5(T - 273.15)^4 + a_6(T - 273.15)^5
\]
of neurons. There is an input layer that receives the data $F$ vs. $x_1, x_2, x_3 \ldots x_n$ and makes a pre-processing of the data. The effect of each variable on the property of interest is considered by giving the data of that variable a certain weight and by shifting them by a bias factor specific to each unit or neuron. The network then calculates a value of the function and compares it with the initial input, and continues doing this for several iterations, until a minimum deviation between calculated and original input data is found. The network stores the values of weights and biases that give the lowest deviation between calculated and experimental data of the dependent variable, values that define the ANN model. Thus, the ANN model is not an explicit analytical model (such as the empirical correlations presented in Table 1), but is a structure of weights and biases presented as math matrices.

ANNs do not learn from programming, they do it through experience, with appropriate learning examples of the variables of interest. ANNs learn by detecting the patterns and relationships in the data provided. Finally, the neural network model capabilities are determined by the transfer functions of the neurons, by the learning rule (back-propagation being the most common one) and by the network architecture. The weights are the adjustable parameters of the model, so the ANNs can be considered as a parameterized system (Agatonovic-Kustrin & Beresford 2000).

A series of requirements must be fulfilled for an ANN model to be capable of correlating data: the number of data, the type and number of independent variables associated with the dependent variable (the property of interest), and the network architecture. In this work, the properties of saline solutions depend on the temperature of the solution and on the salt content, expressed as salinity. The properties considered in this study are listed in Table 1.

It has been demonstrated in several applications that ANNs possess some unique characteristics and advantages for the correlation and prediction of physico-chemical properties of substances, although properties of seawater and saline solutions have not been explored using neural networks. One of the main advantages is that ANNs do not need a pre-defined function to be correlated. The relationship is found by training, as human beings learn during their life.

The usual architecture normally used for this type of application considers a back propagation feed-forward neural network containing three or four layers: the input layer, one or two hidden layers and the output layer (Bose & Liang 1996). According to other studies, four layer architectures with 5–25 neurons in the inner layers are appropriate for correlating properties of fluids. Usually the most appropriate number of layers and of neurons per layer is a matter of trial and error (Valderrama 2014). It is not possible to tell beforehand what will be the best structure for a given application.

The disadvantages of ANNs that must be considered when this method is applied have been discussed in the literature (Livingstone et al. 1996; Valderrama 2014; Valderrama et al. 2014b), the following probably being the most important ones: (i) it requires a large amount of

---

### Table 2 | Relations between the different salinity and temperature scales. All salinities in grams of salt per kilogram of saline solution and all $T$ in keV/m and all $t$ in Celsius

| Property | Relation
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Salinity ($S$)</td>
<td>$S = 1.00472 \cdot S_p$</td>
</tr>
<tr>
<td>Salinity ($S_R$)</td>
<td>$S = S_R$</td>
</tr>
<tr>
<td>Temperature ($T$)</td>
<td>$T = 273.15 + 10^3 \cdot \left( t + 273.15 \right)$</td>
</tr>
<tr>
<td>Temperature ($T_0$)</td>
<td>$T = 0.99976 \cdot t_{48} + 273.15$</td>
</tr>
<tr>
<td>Temperature ($T_1$)</td>
<td>$T = 0.99976 \cdot t_{48} + 0.6554$</td>
</tr>
<tr>
<td>Temperature ($T_2$)</td>
<td>$T = 0.99976 \cdot t_{48} - 4.3989 \cdot 10^{-6}$</td>
</tr>
</tbody>
</table>

### Table 3 | Range of data used for training the ANN for the different properties considered in this study

<table>
<thead>
<tr>
<th>Property</th>
<th>$\Delta T$ (K) (lit)</th>
<th>$\Delta T$ (K) (used)</th>
<th>Range of $S$ (g/kg)</th>
<th>$N$ of data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density (kg/m$^3$)</td>
<td>293–453</td>
<td>293–393</td>
<td>10–160</td>
<td>234</td>
</tr>
<tr>
<td>Specific heat (kJ/kg K)</td>
<td>273–453</td>
<td>273–393</td>
<td>0–180</td>
<td>310</td>
</tr>
<tr>
<td>Vapor pressure (kPa)</td>
<td>283–453</td>
<td>283–393</td>
<td>35–170</td>
<td>261</td>
</tr>
<tr>
<td>Boiling temperature elevation (K)</td>
<td>293–453</td>
<td>273–393</td>
<td>35–100</td>
<td>130</td>
</tr>
<tr>
<td>Dynamic viscosity (kg/m s)</td>
<td>293–423</td>
<td>293–393</td>
<td>15–130</td>
<td>208</td>
</tr>
<tr>
<td>Thermal conductivity (mW/m K)</td>
<td>273–453</td>
<td>273–393</td>
<td>0–160</td>
<td>279</td>
</tr>
<tr>
<td>Osmotic coefficient (N/m)</td>
<td>273–473</td>
<td>273–393</td>
<td>10–120</td>
<td>217</td>
</tr>
<tr>
<td>Surface tension (N/m)</td>
<td>273–313</td>
<td>273–313</td>
<td>0–40</td>
<td>205</td>
</tr>
<tr>
<td>Enthalpy (kJ/kg)</td>
<td>283–393</td>
<td>283–393</td>
<td>0–120</td>
<td>203</td>
</tr>
<tr>
<td>Entropy (kJ/kg K)</td>
<td>283–393</td>
<td>283–393</td>
<td>0–120</td>
<td>203</td>
</tr>
<tr>
<td>Heat of vaporization (kJ/kg)</td>
<td>273–473</td>
<td>273–393</td>
<td>0–120</td>
<td>217</td>
</tr>
</tbody>
</table>
accurate data, depending on the complexity of the relationship between the variables; (ii) it needs to know the variables $x_1, x_2, x_3 \ldots x_n$ that most affect the value of the function $f(x_1, x_2, x_3 \ldots x_n)$, and (iii) the network can suffer from overfitting and overtraining (the network memorizes and does not learn).

Figure 1 presents a simple diagram of an ANN method that can be used for correlating and testing properties of seawater using Matlab™. In the figure, the density is used as the property of interest, but the diagram is similar for all properties. Connections between the different files created for training and testing can be observed in the figure. The file $w_{\text{density.mat}}$ is the weight matrix that defines the ANN model. The application of ANN also firstly requires the collection of accurate enough data for the property of interest, the first block in the diagram of Figure 1.

The ANN described above with the structure shown in Figure 1 was trained 50 times for each property. This was done in an automatic form so results for each property are obtained for the fifty runs, deviations are calculated for each point and the statistical parameters mentioned above (average deviation ($\%\Delta y$), the average absolute deviation ($|\%\Delta y|$) and the maximum absolute deviation ($\text{max}|\%\Delta y|$)). To develop an accurate model to correlate and test an artificial network model in the
% density.m

% This is the Matlab code for training an ANN with values of density of saline solutions
% using as independent variables the temperature and the salinity

% Training Section

%Reading independent variables for training (temperature and salinity)
p = xlsread('variables_density_for_training'); p = p';
% Reading the dependent variable for training (density) in the file density_training;
t = xlsread('density_for_training'); t = t';

% Normalization of all data (values between −1 y + 1)
[pn, minp, maxp, tn, mint, maxt] = premnnmx(p, t);

% Definition of ANN:(topology, activation functions, training algorithm)
net = newff(minmax(pn),[5, 10, 10, 1],{'tansig', 'tansig', 'tansig', 'purelin'}, 'trainlm');

% Definition of frequency of visualization of errors during training
net.trainParam.show = 10;

% Definition of number of maximum iterations (epochs) and global error between iterations (goal)
net.trainParam.epochs = 1,000; net.trainParam.goal = 1e-6;

% Network starts: reference random weights and gains
w1 = net.IW{1,1}; w2 = net.LW{2,1}; w3 = net.LW{3,2}; w4 = net.LW{4,3};
b1 = net.b{1}; b2 = net.b{2}; b3 = net.b{3}; b4 = net.b{4};

%First iteration with reference values and correlation coefficient
before_training = sim(net,pn);
corrbefore_training = corrcoef(before_training,tn);

%Training process and results
[net, tr] = train(net, pn, tn);
after_training = sim(net, pn);

% Back-Normalization of results, from values between −1 y + 1 to real values
after_training = postmnmx(after_training,mint,maxt); after_training = after_training';

Res = sim(net, pn);

% Saving results, correlated temperature in an excel file
dlmwrite('density_trained.xls', after_training, char(9));

% Saving the network (weights and other files)
save w_density.mat

% Testing Section

% This is the Matlab code for testing the density using the trained ANN determined
above and saved in the file property.tm

(continued)
form presented in this work the following files, using density as the property of interest, were written:

1. A Matlab code named density.m: this is the Matlab code for training the net with data on the density and to test the ANN model found by the program. The code is presented in Table 4. For each property a different name can be assigned to this Matlab program (for instance, density.m, viscosity.m).

2. An Excel file containing all data, and results for each run of the ANN (including the training section results and the testing results). The data needed is taken from this excel file as well as the results (the estimated properties) and are stored in the same file. In this spreadsheet all deviations and statistical parameters for each run are calculated.

When the program is run, the first action taken is reading the values of the independent variables, T and S (line 9 in Table 4) and the dependent variable, the property being studied (line 11 in Table 4). Also, the network architecture of the ANN is defined (line 15 in Table 4) as well as the maximum number of iterations and the error between values calculated during one iteration and the preceding one (line 19 in Table 4).

The accuracy of the model is checked by determining the relative, absolute and maximum deviations between calculated values of the properties after training and data from the literature. The trained values are automatically stored by the program in a file named density_trained.xls (line 33 in Table 4) and the results obtained during testing are stored in another file named density_tested.xls (line 52 in Table 4). The ANN model is stored in a file named density.m (line 35 in Table 4). This file is read in the testing section (line 42 in Table 4) to predict the property for new values of T and S. The new values of T and S used to test the network are read from the file variables_density_for_testing.xls (line 44 of Table 4).

Table 5 shows the code for the predicting ANN model that uses the ANN model stored in the file density.m. In fact the code for predicting the property is very similar to the testing section of the main Matlab code presented in Table 4; however, for clarity and simplicity, this is presented in a separate file in this work. With this code the user can determine a value or several values of the property for temperatures and salinities listed in an Excel file named variables_density_for_prediction.xls (line 8 in Table 5). These are new values of T and S not used during training or during testing. The results of these predictions are stored in a file named density_predicted.xls (line 16 in Table 5).

Our group has been working for several years on applications of ANN to property estimation and it seems that good data selection, good classification of data, a reasonable amount of data and the appropriate selection of the independent variables are the key factors for obtaining good correlating and predicting models. That is the reason why when modeling
using ANN, a set of data must always be kept away for testing (and not used in training). In this way the interpolating and predicting capabilities of the model can be evaluated.

**RESULTS AND DISCUSSION**

The results are organized in tables in which for each property three statistical parameters are shown: the average relative deviation (\%Δy), the average absolute deviation (\%Δy) and the maximum deviation between the calculated values (\(y_{m \text{cal}}\)) and data from the literature (\(y_{m \text{lit}}\)). These statistical parameters are the most representative ones of the accuracy of the method as other authors have discussed in the literature (Valderrama & Alvarez 2005). These deviations are defined as:

\[
\%\Delta y = \frac{100}{N} \sum_{i=1}^{N} \left[ \frac{y_{i \text{cal}} - y_{i \text{lit}}}{y_{i \text{lit}}} \right] 
\]

The relative average deviation (\%Δy) indicates how the data are dispersed around the experimental data. If deviations are well dispersed and distributed, the average deviation will be close to zero, independent of the magnitude of the deviations, because negative and positive deviations cancel each other. The average absolute deviation (\%Δy) gives an indication of the magnitude of the deviations. If these are low, the average would be low, and most probably we have an acceptable model. However, the maximum absolute deviation (max\%Δy) is important because it gives the maximum deviation to be expected when the model is used for predicting a value of a given property. The property 'y' is any of the properties of the saline solution of interest in this work, listed in Table 1: density, specific heat, osmotic coefficient, surface tension, viscosity, thermal conductivity, enthalpy, entropy, vapor pressure, latent heat of vaporization, and boiling temperature elevation.

To define the structure of the network, information from the literature was considered and a four layer architecture with five neurons in the inner layer, 10 in each of the two hidden layers and one in the output layer was considered. The accuracy of the chosen final network was checked by determining the relative and absolute deviations between the calculated values of the properties after training and data from the literature.

During training and testing, the absolute maximum deviations between correlated and literature values were below 3.1% for all properties. These maximum deviations are values considered to be acceptable for engineering calculations (Harg 1985) and indicate that the ANN learned in an appropriate way. This is expected, because the relationship between the properties and the independent variables, temperature and salinity, although it may be physically and chemically complex, is not from the mathematical point of view. In fact, the properties have all the characteristics of ‘mathematically favorable’ functions. Also, in all
runs and for all properties, network memorization was not observed. This memorization occurs when the network assumes as output exactly the same input values, so deviations are exactly zero, and the predicting capabilities of the network are lost.

Once the training was successfully done and the optimum network architecture was determined, 50 input data \((T, S)\) for each of the properties, not used in the training process, were used for testing the predictive capabilities of the network. Comparison of the accuracy of the ANN model during training and testing is presented in Table 6. Also, Figures 2–4 show the maximum absolute deviations and

![Figure 2](https://iwaponline.com/jwrd/article-pdf/5/4/454/377639/jwrd0050454.pdf)

**Figure 2** | Maximum absolute deviation between ANN values and data from the literature for density of saline solutions: 
--- during training and 
--- during testing.

<table>
<thead>
<tr>
<th>Property</th>
<th>Correlation</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N run</td>
<td>%Δ-y</td>
</tr>
<tr>
<td>Density</td>
<td>43</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Specific heat</td>
<td>42</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Vapor pressure</td>
<td>40</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Boiling temperature elevation</td>
<td>3</td>
<td>0.01</td>
</tr>
<tr>
<td>Dynamic viscosity</td>
<td>8</td>
<td>0.03</td>
</tr>
<tr>
<td>Thermal conductivity</td>
<td>28</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Osmotic coefficient</td>
<td>3</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Surface tension</td>
<td>9</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Enthalpy</td>
<td>22</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Entropy</td>
<td>1</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Heat of vaporization</td>
<td>30</td>
<td>&lt;0.01</td>
</tr>
</tbody>
</table>

![Figure 3](https://iwaponline.com/jwrd/article-pdf/5/4/454/377639/jwrd0050454.pdf)

**Figure 3** | Maximum absolute deviation between ANN values and data from the literature for entropy of saline solutions: 
--- during training and 
--- during testing.

![Figure 4](https://iwaponline.com/jwrd/article-pdf/5/4/454/377639/jwrd0050454.pdf)

**Figure 4** | Maximum absolute deviation between ANN values and data from the literature for viscosity of saline solutions: 
--- during training and 
--- during testing.
the maximum deviations for all 50 runs for the density, entropy and viscosity, respectively. Similar figures are obtained for all properties.

In the case of density, all 50 runs give absolute maximum deviations lower than 0.01%, run 43 being the one that gives the lowest values of the three statistical parameters. In the scale of the figure, the differences between maximum deviations during training and testing seem to be high but in fact all deviations are lower than 0.1%. In the case of entropy, deviations between training and testing vary between 0.35 and 3%. For runs 3–5 and 13–19, higher differences are observed but they are still within acceptable ranges of deviations. Run 7 presents the maximum absolute deviation for training (2.44%). For viscosity, Figure 4 shows behavior similar to that of the entropy of Figure 3. In this case run 8 gives the lowest deviations for both training and testing (0.17%).

Comparisons with the correlations recommended by Sharqawy et al. (2010) and with recent results using Padé approximants (Valderrama & Campusano 2015) were carried out and are presented in Table 7. The last column in the table presents the ANN model results found in this work. The results show the accuracy of the different models which, for engineering applications, are low and acceptable. The ANN results are not surprising if one considers the ‘good behavior’ of the properties of seawater with the variables T and S. This means that the variation in the values of the properties are continuous (increasing or decreasing) and follow smooth patterns. Thus the learning process, in which the ANN finds the relation between the property and the variables T and S, is relatively simple.

For vapor pressure, however, the variation of \( P_{\text{sat}} \) with temperature is dramatic, going from approximately 1 to 200 kPa. This is the only property of the 11 studied that changes by a factor of 200 in the ranges of \( T \) and \( S \) shown in Table 3 (283–393 for \( T \) and 35–170 for \( S \)). Because of this great variation, the ANN gives poor results when data of \( P_{\text{sat}} \) and the corresponding temperature and salinity are directly used in the training process. Absolute deviations can be as high as 16% in this case. Therefore, for vapor pressure the learning by training was simplified by providing the network with \( \ln P_{\text{sat}} \) vs. \( 1/T \) data. It is known that, to a very good approximation, the logarithm of \( P_{\text{sat}} \) presents a linear relationship with \( 1/T \) (Poling et al. 2001). After the ANN model is found, the vapor pressure \( P_{\text{sat}} \) is calculated from \( \ln P_{\text{sat}} \) (just by taking the antilog) and deviations between ANN results and literature values are determined using vapor pressure values, not the logarithm.

Finally, something must be said about the different models presented in the literature and used in this work for comparison with the ANN results (Table 6). The correlations presented in the literature, in particular those of Sharqawy used in this work, have the advantage of being based on experimental data and therefore if calculations are performed within the ranges established by each correlation the values are considered to be accurate within the accuracy of the model. In the case of Padé, the authors used the pseudo-experimental data of Sharqawy et al. (2010) and proposed simple Padé expressions for all properties. Simplicity and generality is then the main advantage of the Padé model. In that case, only few parameters are needed and one type of model is used for all properties. ANNs, on the other hand, do not provide analytical expressions (as Sharqawy correlations or Padé models) but provide a weight and bias matrices that relate the properties with temperature and salinity. The relationships were found by training the network, acquiring certain predicting

<table>
<thead>
<tr>
<th>Property</th>
<th>Various models ( \text{(Sharqawy et al. 2010)} )</th>
<th>Padé (Valderrama &amp; Campusano 2014)</th>
<th>ANN (this work) Training</th>
<th>ANN (this work) Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Density</td>
<td>0.01–2.5</td>
<td>0.13</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Specific heat</td>
<td>0.001–4.6</td>
<td>0.41</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>Vapor pressure</td>
<td>0.015–0.2</td>
<td>0.10</td>
<td>0.06</td>
<td>0.06</td>
</tr>
<tr>
<td>Boiling temp. elevation</td>
<td>0.018–0.7</td>
<td>0.10</td>
<td>0.04</td>
<td>0.06</td>
</tr>
<tr>
<td>Dynamic viscosity</td>
<td>0.4–1.5</td>
<td>0.82</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>Thermal conductivity</td>
<td>0.4–3</td>
<td>0.18</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Osmotic coefficient</td>
<td>0.1–1.4</td>
<td>1.14</td>
<td>0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>Surface tension</td>
<td>0.08–0.18</td>
<td>0.03</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>Enthalpy</td>
<td>0.5–2</td>
<td>0.26</td>
<td>0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>Entropy</td>
<td>0.5–35</td>
<td>0.42</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>Heat of vaporization</td>
<td>0.01</td>
<td>0.15</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
</tbody>
</table>
capabilities. In all models, however, extrapolations must be done with care, considering that in all cases the models were obtained using data in defined ranges of temperature and salinity.

CONCLUSIONS

An ANN model has been used to correlate and predict properties of seawater and saline solutions at a given temperature and salinity. The study and the results allow two main conclusions: (i) any of the properties studied can be obtained with good accuracy, giving absolute average deviations below 0.3%; and (ii) to facilitate the learning of the ANN, it is recommended to provide the network with data transformed in such a way that the relationship between the property and the independent variables is simpler (such as the more linear relationship of $\ln P^{sat}$ vs. $1/T$).

ACKNOWLEDGEMENTS

The authors thank the support of the National Council for Scientific and Technological Research (CONICYT) through the research grant Anillo ACT 1201, the project was financed by the Innovation for Competitiveness Fund of the Antofagasta Region in Chile. The authors also thank the Center for Technological Information (CIT, La Serena-Chile) for computer and library facilities and the Direction of Research of the University of La Serena, for permanent support.

REFERENCES


First received 9 November 2014; accepted in revised form 7 April 2015. Available online 11 May 2015.