American Journal of Civil Engineering and Architecture, 2015, Vol. 3, No. 3A, 1-8
Available online at http://pubs.sciepub.com/ajcea/3/3A/1

DOI:10.12691/ajcea-3-3A-1

Prediction of Daylighting and Energy Performance Using Artificial Neural Network and Support Vector Machine

Sixuan Zhou[1], Dong Liu[2,*]

[1]Florida State University, Tallahassee, United States
[2]Department of Civil and Architectural Engineering, Zhengzhou University, Zhengzhou, China
*Corresponding author: dongliuzzu@163.com

Received March 10, 2015; Revised May 5, 2015; Accepted August 18, 2015

Abstract The computerized building design has been developed to optimize building design. Machine learning techniques are explored to help predict building design performance. However, in the current building design tools, the optimization techniques have not been integrated closely with the computerized building design tool. Only a few tools add some optimization methods such as genetic algorithms. The aim of the paper is to use machine learning techniques to predict the daylighting metrics such as illuminance and thermal metrics for different combinations of window glazing transmittances, weather conditions and blind reflectance values. In this paper, three machine learning algorithms were evaluated, PCA (principal component analysis), ANN (artificial neural network), SVM (support vector machine). The PCA and forward feature selection algorithms were used to extract features or reduce the dimension of the features. Four comparisons were conducted: NN with PCA, ANN without PCA, SVM with PCA, and SVM without PCA. The results show that the NN with PCA has the best accuracy for the daylighting UDI classification problem. The ANN has an acceptable accuracy for the energy prediction problem.

Keywords: *artificial neural network, support vector machine, performance prediction, UDI, energy, daylighting, illuminance*

Cite This Article: Sixuan Zhou, and Dong Liu, "Prediction of Daylighting and Energy Performance Using Artificial Neural Network and Support Vector Machine." *American Journal of Civil Engineering and Architecture*, vol. 3, no. 3A (2015): 1-8. doi: 10.12691/ajcea-3-3A-1.

1. Introduction

Daylight harvesting has the potential to offer significant energy and economic benefits regarding space heating, cooling, and lighting energy that represent more than half of commercial site energy consumption. The buildings sector accounted for almost 40% of primary energy consumption in 2008 and accounted for approximately 8% of the global primary energy consumption.

The building sector consumed more primary energy than the transportation and industrial sectors. Commercial buildings accounted for one-fifth of U.S. energy consumption. The three main types of commercial buildings are office space, retail space, and educational facilities [1]. Space heating, lighting, and space cooling represent more than half of commercial site energy consumption. Daylighting has the potential to offer significant energy and economic benefits in these three aspects through daylight harvesting. Daylight harvesting can result in significant electric lighting reduction in commercial buildings by the application of electric lighting controls. Turning off electric lights when sufficient daylight is available can save lighting energy costs. Because daylight introduces less heat into a building than the equivalent amount of electric lighting, cooling costs can also be reduced with appropriate daylighting design [2]. Because these reductions are maximized during peak electricity demand periods, daylight harvesting can greatly contribute to peak electricity demand reduction [3,4].

The benefits of daylighting include improving visual performance as well as reducing energy consumption [5]. Research showed that student performance and health were related to daylighting conditions in the classroom [6]. The benefits from daylighting greatly depend on daylight availability, which varies significantly by latitude, sun path, sky conditions, and climate. The selection of appropriate daylighting systems is an important approach of making best use of daylighting [7].

The complexity of lighting reflection between surfaces makes it difficult to conduct accurate manual calculations. Computer simulation tools were developed to speed the calculation process [8]. Computer simulation methods offer flexibility that other methods sometimes cannot provide and provide a convenient way of parametrically evaluating designs in comparison to other design alternatives. The tools are based on different algorithms, which are generally based on ray tracing or radiosity methods. According to the survey conducted by Reinhart and Fitz [9], 50% of programs in the daylighting simulation field use the Radiance simulation engine. Different analytical models were developed to predict building performance [10]. A rule-based expert system was developed to integrate daylighting and thermal simulations [11,12].

The standard simulation software Radiance was validated by Mardaljevic [13,14]. Reinhart and Andersen [15] also validated Radiance for simulating translucent materials and the results shows that the mean bias errors (MBEs) were below 9% and rooted mean square errors (RMSEs) were below 19% for all desktop and ceiling sensors considered. These simulation accuracies were even better than those earlier reported for standard glass, plastic, and metal material types the errors of which were approximately 17% (MBE) and 30% (RMSE). One possible reason for generating these errors was that sharp indoor illuminance gradients (e.g. shadows generated by direct sunlight) were mitigated through translucent panels.

Researchers also validated the Radiance-based software. For example, Maamari and Andersen [16] assessed the accuracy of different lighting simulation methods to predict the performance of CFS. DElight had the advantage of the complete daylighting simulation without using other supplementary tools. The Radiance algorithm had the advantage of avoiding the need for BTDF data, but it was limited to a specific type of CFS. Reinhart and Herkel [17] compared six Radiance-based methods: ubiquitous daylight factor, ADELINE 2.0; the classified weather data and ESP-r version 9, new method daylighting coefficient (DC) total, and new method DC without direct sunlight. The classified weather data served as a reference case. They determind that the DC total approach showed a very low RMSE and MBE (for both global and diffuse daylight). Loutzenhiser and Maxwell [18] examined measured and simulated light levels in an actual building and determined the average differences in predictions of daylight illuminance at reference points. Andersen and Kleindienst [19] developed a visualization tool that not only could visualize the illuminance by using “spatio-temporal irradiation maps” developed by Mardaljevic [20] but also used Perez’s ASRC-CIE sky model [21]. Hu and Olbina developed a simplified analytical model to predict energy and daylight values. The model can significantly reduce the runtime while improving the accuracy [22]. Li reviewed the illuminance calculation methods that included daylighting measurement, sky illuminance and luminance, and daylighting coefficient (DC) and daylighting factor (DF) methods [23].

Radiance has the capability of effectively simulating the daylighting condition when three types of skylight [17,24]. The software DAYSIM developed by Reinhart and Walkenhorst [25] is based on the standard lighting simulation software Radiance. This software used the concept of DC to simulate the daylighting. Meanwhile, the energy consumption of electric lighting can also be calculated based on the daylighting illuminance. This software was validated using a test room [25].

Different machine learning methods have been used for lighting and energy prediction. For example, neural network also were used to predict lighting performance [26]. Hu and Olbina developed a model to predict the optimum slat angles of split blinds to achieve the designed indoor illuminance. The model was constructed based on a series of multi-layer feed-forward artificial neural networks (ANNs) [27]. To further improve the control strategy, a simplified control model was developed to improve the system performance [28].

This research is to use machine learning techniques to predict the daylighting metric and thermal metric for an arbitrary window glazing transmittance, weather condition and blind reflectance values. That is to say, given an exterior climate condition, window configuration, I need to predict the hourly daylighting levels at several sensor points (classification problem) and hourly energy consumption (regression problem). The daylighting metric uses hourly Useful Daylight Illuminance (UDI). The UDI is represented by the percentages of the occupied times of the year when the UDI is achieved (100-2000 lx), is not sufficient (less than 100 lx), or is exceeded (more than 2000 lx). The hourly energy consumption of a building will be used. E nergyPlus is developed by Department of Energy and was validated and recognized as one of the most accurate whole building energy simulation engines. This research uses EnergyPlus simulation engine to simulation annual energy. The generated data were used to train the algorithm to evaluate the performance.

2. Algorithm Development

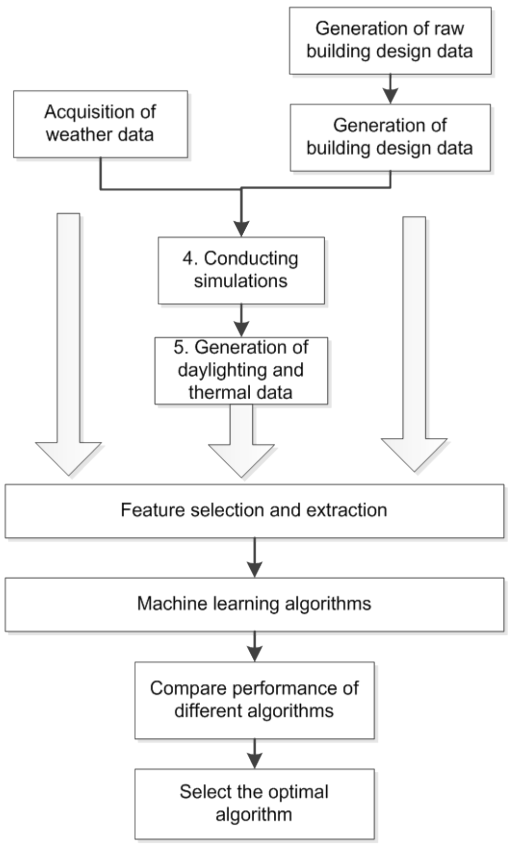

Figure 1. Overall structure

The overall structure of the algorithm is shown in Figure 1. The following part of this section would elaborate the workflow in more details. The data generation programs generate all data needed in the machine learning algorithms. These data include the output data and input data such as weather data, and building design data. These were generated by Python and

Matlab programs. The daylighting and thermal data are output data that will be used for supervised learning.

The algorithm used includes two tasks: (1) cclassification problem: This requires classifying the daylighting level values in interior room in to different UDI values; (2) regression problem: this requires predicting the energy consumption based on different weather condition and glazing and blind materials.

The following section would introduce each workflow of the daylighting UDI classification problem and energy prediction problem.

The algorithm mainly includes three parts: (a) algorithms used to generate input files for simulations; (b) algorithms used to process data for machine learning algorithms.

In Figure 2, a Python program was developed to generate the input files. There were a total of 81 input files. Each file has a different glazing transmittance and blind reflectance. Then Matlab was used to extract the values of all potential features from the simulated files. The data included the potential feature data as well as the output (i.e., class) data. The implementation of the algorithms is introduced in this section.

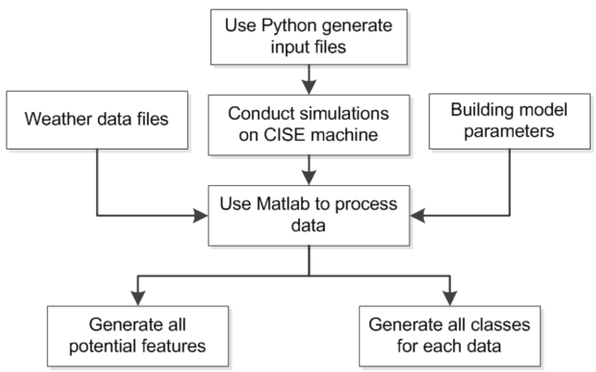

Figure 2. Process of generating and parsing data (input and output)

2.1. Input File Generation

The input files were generated by generating some of the key parameters such as window transmittance and glazing coefficients used by the simulation files. The program was coded by using Python and run in command line. The command line options are shown as follows (see Figure 3):

```
C:\U\j\Desktop\__genEPlus__> python genEPlus.py -h
usage: genEPlus.py [-h] [-g G] [-b B] [-inputFile INPUTFILE]
                   [-outputFolder OUTPUTFOLDER] [-xmin XMIN] [-zmin ZMIN]
                   [-ww WW] [-wh WH] [-option OPTION]

generate the EPlus file

optional arguments:
  -h, --help            show this help message and exit
  -g G                  glazing reflectance: 0.5
  -b B                  blinds reflectance: 0
  -inputFile INPUTFILE  EPlus file, no .idf
  -outputFolder OUTPUTFOLDER
                        output location
  -xmin XMIN            x coordinate
  -zmin ZMIN            z coordinate
  -ww WW                window width
  -wh WH                window height
  -option OPTION        select: 0-6
```

Figure 3. Command line options of generating simulation input files

As shown in Figure 3, the program can achieve the following functions:

- Generate the input file format specific for window specification (location, glazing parameters, dimensions and so on). The 3D model of the building is shown in Figure 4. The model was created in SketchUp. The model can be integrated with EnergyPlus and OpenStudio tool for energy simulations.
- Generate the input file format specific for blind setting specification (location, blind reflectance, slat angle, slat width, and so on)
- Other data related to construction materials, such as wall thermal properties. Note that most of the thermal and geometry parameters were loaded from a separate text files because these parameters are static and do not require parametric setting.
- The location of sensors such as occupancy sensors affects the building system performance [29]. From previous study One third of the space penetration is selected in this case study.

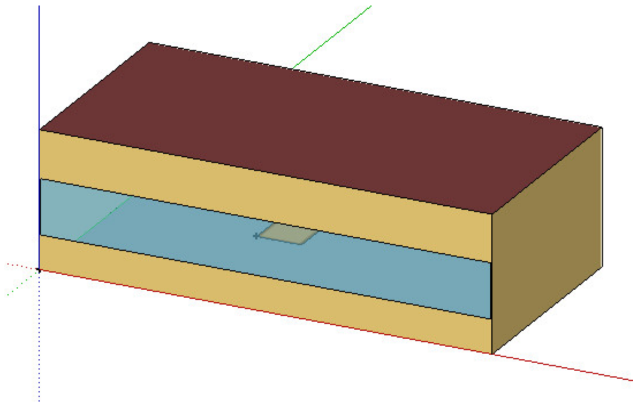

Figure 4. Illustration of building model geometry

2.2. Data Parsing

An algorithm was developed to process the data from the simulation results. The simulations were conducted by EnergyPlus in Linux machine. The simulation time was about 10 hours. The algorithms were coded in Matlab. The total data set include 8760*81 hourly data values include energy consumption as well as daylighting UDI range values.

To generate the data, the following procedure was used. Read the data from the generated the CSV files (8760*81 tuple of data), and then process the data to extract the occupied hour data which is from 8 am to 5 pm. The data were used as output because we do not need to consider the night time.

Permute the data to generate the data for input and output variables for the machine learning algorithms. From all the occupied data, I randomize the data and select 3285 hourly data for training and another set of 3285 hourly data for testing.

Use Matlab to extract the input variables from weather files and generate the input data sequence according to the sequence of the permutated output data.

The SVM, PCA and ANN were chosen. SVM is one of the best algorithms for classification, which might be suitable for this daylighting UDI classification. Some literature used neural networks to analyze the daylighting levels. Though their research is not classification problem. But a threshold method was used to convert it to classification. Some literature shows that using neural networks for predicting building energy usage. The problem in this paper considered more complicated issues by including glazing transmittance and blind reflectance [27]. A number of features might be considered. To reduce the dimensions, the principal component analysis was used to analyze how the reduction of feature dimension affects the accuracy of the algorithms.

PCA is used to generate the orthogonal input variables. The input variables will also considered by testing all possible features to select the best ones.

- In the SVM and NN algorithms, different sets of parameters were tested. For example, for SVM, Gaussian Radial Basis Function (RBF) and linear kernels were tested and their performances were compared.
- Three algorithms were considered: (a) principal component analysis; (b) neural network; (c) support vector machine. PCA was used to generate the orthogonal input variables that were used by NNs and SVM in the next phase. In addition, for the neural network models, different parameters were considered, for example, the hidden layers were evaluated to select the optimal hidden layer number. The appendix shows the basic structure of the code for neural network without/with PCA (see Figure 5).

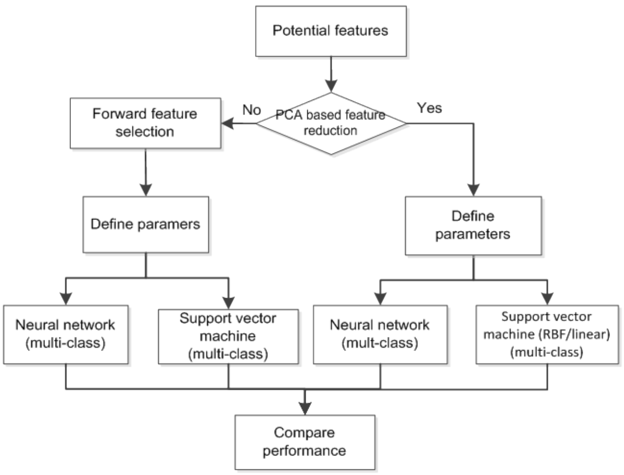

Figure 5. Process of solving the daylighting UDI classification problem

The overall process of solving the regression problem was shown in Figure 6. The appendix shows the basic structure of the code for SVM without/with PCA.

- It is similar with the classification problem. Python program will be used to generate the input files. There are a total of 81 input files. Each file has a different glazing transmittance and blind reflectance.
- Only neural network was considered. Different parameters were evaluated to select the optimal parameters for the neural network model.

The output of the classification problem (i.e., classification of daylighting levels) is that UDI value at the sensor point and how many types were explored, we can use different range for this problem. For example, the ranges: 0-100 lx, 100-500 lx, 500-200 lx, and >2000 lx, are standard ranges which are commonly used. But I would explore other ranges to provide an even precise division of ranges.

The output variables of the regression problem are: Hourly Sensible Heating Rate, and Hourly Sensible Cooling Rate.

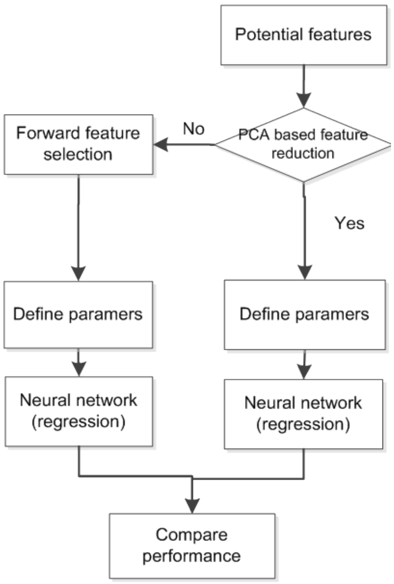

Figure 6. Process of solving the energy consumption prediction problem

The potential features come from three sources:

1) Weather related features: weather files provided by the U.S. Department of Energy and LBN, The weather data for US can be found in the following link: http://apps1.eere.energy.gov/buildings/energyplus/cfm/weather_data3.cfm/region=4_north_and_central_america_wmo_region_4/country=1_usa/cname=USA
2) Building related features: The building design data will be developed based on ASHRAE standard and other building design standards. The specific building design dataset will be from (1) building design book such as ASHRAE standards (link: https://www.ashrae.org/resources--publications/bookstore/standard-90-1#2007).
3) Window related features: input file generated by Python and Matlab. The following section would introduce the Python program and how to generate the building model. In addition, Matlab is used to extract the training and testing data from all the potential features and data.

Not all the above features were used for the classification and regression problems. Therefore, two methods were used to extract the features for the problem.

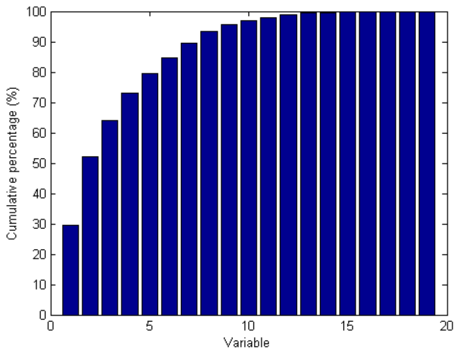

Figure 7. Cumulative sum of variances of features in PCA

First of all, pprincipal component analysis was used to extract the most important variables that would account for 90% variance. All the possible data were inputted and then PCA was used to analyze the input data. Figure 7 shows the cumulative of the variables. Therefore, the eight variables for neural network training was shown.

Select a minimum set of features that are very important to the problem. Then use forward selection approach: add one feature into the feature set at a time through comparing the output errors with this feature and output errors without this feature. Then add that feature that could give the least errors and high accuracy. Repeat this process until the error does not change much. The neural network algorithms were used to calculate the errors. The following input variables show a higher accuracy for neural network:

- Exterior Horizontal Illuminance From Sky
- Exterior Horizontal Beam Illuminance
- Glazing Transmittance
- Blind Reflectance
- Solar Azimuth Angle
- Solar Altitude Angle
- Solar Hour Angle

3. Experiment

The data generation algorithms was coded by Python,. The machine learning algorithms were coded by using Matlab. For the daylighting UDI classification problems, The following indicator was used to measure the accuracy of the model. Number of testing samples with correct prediction / total number of testing samples

For the energy usage prediction problem (regression problem), I measure the RMSE (root-mean-square error) by using the following equation

$$RMSD = \sqrt{\frac{\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}{n}}$$

3.1. Daylighting UDI Classification

There were 3285 data were used for training and an independent 3285 data samples were used for testing. The following four different algorithms were used to test the results.

Comparison of neural networks using PCA and without using PCA

The Table 1 shows the accuracy of the two models: the model using PCA shows that the input variables were extracted by using PCA; the neural network without PCA is used by directly using the features shown in Section 4: Feature selection and extraction.

Table 1. Accuracy of neural networks using PCA and without using PCA

# of hidden layers	Accuracy	
	Using PCA	Without PCA
5	92.51%	83.65%
10	94.61%	83.84%
15	96.35%	90.29%

Comparison of SVM using PCA and without PCA

The two accuracy rates are shown in Table 2. The SVM without PCA shows a higher accuracy than the SVM without PCA. This is contradictory with the methods using neural network. In addition, RBF shows a better performance than linear kernel function.

Table 2. Accuracy of support vector methane using PCA and without using PCA

Kernel function	Using PCA	Without PCA
RBF	70.14%	73.88%
Linear	62.15%	70.08%

3.2. Energy Usage Prediction

The regression problems are studied and analyzed by analyzing different input variables as well as the parameters (see Figure 8). The calculated RMSE = 127 (Watts). Though the error is large, from practical means, the error is still acceptable because 127 Watt error per hour is acceptable.

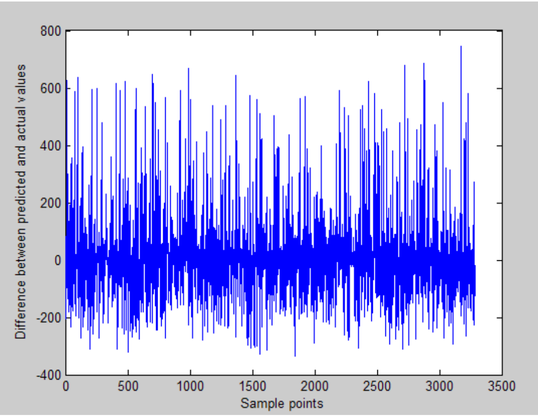

Figure 8. Residual error of the energy prediction

4. Discussions

The prediction accuracy of daylighting UDI is relative high compared to regression problem. Neural network using principal component analysis generated the highest accuracy (about 96%) compared to other algorithms. The high accuracy of neural networks is consistent with the current research. The neural network model shows a higher accuracy than SVM. The SVM models have only about 70% though my initial expectation was that SVM should be at least as good as neural network. There are several possible reasons:

- The feature selection used the neural network algorithm to select the features. Thus, this bias might cause SVM has a lower accuracy.
- The feature might not be enough for SVM. Therefore, if I may add more features without using feature selection, the accuracy of SVM might become higher.
- There might be some other kernel functions that will give a better result.

The SVM using PCA has the lowest accuracy. The possible reason is that the number of features is limited and SVM cannot accuracy create the optimal hyperplane. The results show that about 3 out of 365 days show a wrong prediction by using the neural network algorithm. The features that are important for the predictions include: Exterior Horizontal Illuminance From Sky; Exterior Horizontal Beam Illuminance; Glazing Transmittance; Blind Reflectance; Solar Azimuth Angle; Solar Altitude Angle, and Solar Hour Angle. Sky and sun illuminance levels are important because sky light is also an main source of daylighting. Unlike the energy consumption prediction, glazing and blind parameters are also necessary feature because change of glazing transmittance or blind reflectance changes the daylighting level significantly.

For energy usage prediction, the calculated RMSE is equal to 127W. The error is not high compared to the daylighting classification. From a practical pespective, the number is actually acceptable because 127W is significant deviation for energy consumption. The method still can be used for energy prediction. The analysis shows that the energy consumption is complicated and depends on more features than normally expected. In addition, the prediction of energy consumption depends on the exterior climate condition. The frequent change of external temperature significantly affects the energy consumption. Some features that are important because they are related to source of energy to a large extent. For example, temperature (Zone Outdoor Wet Bulb) is an feature for energy consumption (HVAC system). The solar is also an main feature for energy consumption. Those features that are significant include: Exterior Horizontal Beam Illuminance [lux](Hourly); Exterior Beam Normal Illuminance [lux](Hourly); Luminous Efficacy of Sky Diffuse Solar Radiation [lum/W](Hourly); Zone Outdoor Wet Bulb [C](Hourly); Diffuse Solar [W/m2](Hourly); Direct Solar [W/m2](Hourly); Solar Azimuth Angle [deg](Hourly); Solar Altitude Angle [deg](Hourly). Some other parameters such as sky efficacy are not important because sky light is not an main factor for building energy consumption based on normal analysis.

5. Conclusions

The literature review shows that this topic has not been studied thoroughly because of the dynamics of sky conditions. In addition, current literature show that the prediction is mainly based on very few building design parameters such as window glazing type. Three machine learning algorithms have been applied by analyzing daylighting UDI classification and energy usage prediction problems. Two feature extraction methods were used and compared: forward selection and PCA based feature dimension reduction. The results show that the neural network using principal component analysis generated the highest accuracy (about 96%) compared to other algorithms. The high accuracy of neural networks is consistent with the current research. For the energy usage prediction problem, the calculated RMSE is about 127W and the error is not high compared to the actual energy usage. This research has practical meanings. The literature review shows that currently only a very few tools integrate the machine learning techniques. The designers can use the methods to help them design window systems. The machine learning algorithms could not only reduce their design and testing times but also could provide sensibility analysis by allowing them to view the trend of different daylighting design strategies.

Appendix

```
Support vector machine with/without PCA
%% important parameters:
inIndex = [1,2,13,14,17,18]; % input variable index
outMode_train = 2;
outMode_scale01 = 2;
isPCA = 1 ; % using PCA == 1; not use PCA ==0;
%% load
alldata = importdata('alldata.dat');
nd = 3285;
pt = [1:nd];
%% input for training and testing
% method 1: without PCA
if(isPCA ==0)
%%% training: input
in = alldata(pt,inIndex)';
%%% test::input
ActIn = alldata(nd+1:nd*2,inIndex)'; %% transposed
End
%%% method 2: using PCA
if(isPCA==1)
%%% training: input
[COEFF,SCORE_in,latent_in]=
princomp(zscore(alldata));
in = SCORE_in(pt,:)';
in = in(1:10,:);
%%% test::input
[COEFF,SCORE_test,latent_test]=
princomp(zscore(alldata));
ActIn = SCORE_test(nd+1:nd*2,:)'; %% transposed
ActIn = ActIn(1:10,:);
end
%% output for training and testing
%%% training: output
out=zeros(1,nd);
m = alldata(pt,10)';
out = genDaylightOut(m,nd,outMode_train);
%%% testing: output:
m2 = alldata(nd+1:nd*2,10)';
ActOut_noscale = m2;
ActOut_scale01=
genDaylightOut(m2,nd,outMode_scale01);

%% SVM training
PreOut_scale01=multisvm_2(in',out',ActIn')';
% accuracy: case 2: classification
t1 = (PreOut_scale01 - ActOut_scale01 )==0;
mseClass01 = sum( t1) / nd; %% calcualte the accuracy
mseClass02 = sqrt( sum((PreOut_scale01 -
ActOut_scale01).^2)/nd);

%Models a given training set with a corresponding
group vector and
%classifies a given test set
function [result]=
multisvm_2(TrainingSet,GroupTrain,TestSet)
u=unique(GroupTrain);
numClasses=length(u);
result = zeros(length(TestSet(:,1)),1);

%build models
for k=1:numClasses
%Vectorized statement that binarizes Group
%where 1 is the current class and 0 is all other classes
G1vAll=(GroupTrain==u(k));
```

```
models(k)=
svmtrain(TrainingSet,G1vAll,'kernel_function','rbf');
end

%classify test cases
for j=1:size(TestSet,1)
for k=1:numClasses
if(svmclassify(models(k),TestSet(j,:)))
break;
end
end
result(j) = k;
end

Nneural networks with/without PCA
%% important parameters:
inIndex = [1,2,13,14,17,18];
outMode_train = 1;
outMode_scale01 = 2;
isPCA = 1 ; % using PCA == 1; not use PCA ==0;
hiddenSizes = 10;
%% load
alldata = importdata('alldata.dat');
nd = 3285;
pt = [1:nd];

%% input for training and testing
%%%%%%%%%%%%% method 1: without PCA
if(isPCA ==0)
%%% training: input
in = alldata(pt,inIndex)';
%%% test::input
ActIn = alldata(nd+1:nd*2,inIndex)'; %% transposed
end
%%%%%%%%%%%%%% method 2: using PCA
if(isPCA==1)
%%% training: input
[COEFF,SCORE_in,latent_in]=
princomp(zscore(alldata));
in = SCORE_in(pt,:)';
in = in(1:10,:);
%%% test::input
[COEFF,SCORE_test,latent_test]=
princomp(zscore(alldata));
ActIn = SCORE_test(nd+1:nd*2,:)';
ActIn = ActIn(1:10,:);
end
%% output for training and testing
out=zeros(1,nd);
m = alldata(pt,10)';
out = genDaylightOut(m,nd,outMode_train);
%%% testing: output:
m2 = alldata(nd+1:nd*2,10)';
ActOut_noscale = m2;
ActOut_scale01=
genDaylightOut(m2,nd,outMode_scale01);
%% train
aaa = trainNeuralnetworkFcn(net,in,out);

%% test::training
PreOut_noscale = aaa(ActIn); % actual number
PreOut_scale01=
genDaylightOut(PreOut_noscale,nd,outMode_scale01); %
scale the data
mseCon = sqrt( sum((PreOut_noscale -
ActOut_noscale).^2)/nd);
% accuracy: case 2: classification
t1 = (PreOut_scale01 - ActOut_scale01 )==0;
mseClass01 = sum( t1) / nd; %% calcualte the accuracy
mseClass02 = sqrt( sum((PreOut_scale01 -
ActOut_scale01).^2)/nd);
%% neural network function
function aaa = trainNeuralnetworkFcn(in, out)
learnRate = 0.05;
epoch =100;
error = zeros(1,epoch);
batchProcess = false;
hiddeNode = ones(10,1);
outputNode=ones(1,1);
deltaIW = cell(100);
for i=1:epoch
for j=1:nSample
hiddenNode = tansig(IW*input(:,j));% hiddenNode =
11 X 1, IW,OW is row vector
outputNode= tansig(OW*hiddenNode);
deltaOW(:,j)= learnRate*(output(j)-outputNode)*(1-
outputNode)*(1+outputNode)/2*hiddenNode;
deltaIW{j} =( learnRate*(output(j)-outputNode)*(1-
outputNode)*(1+outputNode)/2*OW'.*(1-
hiddenNode(1:10)).*hiddenNode(1:10) )* input(:,j)';
if(~batchProcess)
OW = deltaOW(:,j)'+OW;
IW = IW + deltaIW{j};
end
end
%batch processing
if(batchProcess)
OW = sum(deltaOW, 2)'+OW;
deltaIW2 = zeros(10,2);
for m=1:nSample
deltaIW2 = deltaIW2 + deltaIW{j};
end
IW = IW + deltaIW2;
end
end
end
```

References

[1] Pacific Northwest National Laboratory. 2011 Buildings energy data book., Washington, DC: U.S. DOE Energy Efficiency and Renewable Energy; 2011.

[2] Tzempelikos A, Athienitis AK. "The impact of shading design and control on building cooling and lighting demand," *SOL ENERGY*, 81 (3), 369-82, 2007.

[3] Shen E, Hu J, Patel M. "Energy and visual comfort analysis of lighting and daylight control strategies," *BUILD ENVIRON*, 78, 155-70, 2014.

[4] Shen E, Patel M, Hu J. "Comparison of Energy and Visual Comfort Performance of Independent and Integrated Lighting and Daylight Controls Strategies," in *ACEEE Summer Study on Energy Efficiency in Industry*, American Council for an Energy-Efficient Economy, New York, NY, USA.

[5] Hviid CA, Nielsen TR, Svendsen S. "Simple tool to evaluate the impact of daylight on building energy consumption," *SOL ENERGY*, 82 (9), 787-98, 2008.

[6] Heschong L. Daylighting in Schools: An Investigation into the Relationship between Daylighting and Human Performance. Detailed Report., Fair Oaks, CA: Heschong Mahone Group; 1999.

[7] Vine E, Lee E, Clear R, DiBartolomeo D, Selkowitz S. "Office worker response to an automated Venetian blind and electric

lighting system: a pilot study," *ENERG BUILDINGS*, 28 (2), 205-18, 1998.

[8] Reinhart CF, Wienold J. "The daylighting dashboard - A simulation-based design analysis for daylit spaces," *BUILD ENVIRON*, 46 (2), 386-96, 2011.

[9] Reinhart C, Fitz A. "Findings from a survey on the current use of daylight simulations in building design," *ENERG BUILDINGS*, 38 (7), 824-35, 2006.

[10] Liu M, Wittchen KB, Heiselberg PK. "Development of a simplified method for intelligent glazed fa?ade design under different control strategies and verified by building simulation tool BSim," *BUILD ENVIRON*, 74, 31-8, 2014.

[11] Hu J, Olbina S. "An Expert System Based on OpenStudio Platform for Evaluation of Daylighting System Design," in *ASCE International Workshop on Computing in Civil Engineering*, ASCE, Los Angeles, CA, 186-93.

[12] Gagne JML, Andersen M, Norford LK. "An interactive expert system for daylighting design exploration," *BUILD ENVIRON*, 46 (11), 2351-64, 2011.

[13] Mardaljevic J. "Validation of a lighting simulation program under real sky conditions," *Lighting Research and Technology*, 27 (4), 181-8, 1995.

[14] Mardaljevic J. "Simulation of annual daylighting profiles for internal illuminance," *Lighting Research and Technology*, 32 (3), 111-8, 2000.

[15] Reinhart CF, Andersen M. "Development and validation of a Radiance model for a translucent panel," *ENERG BUILDINGS*, 38 (7), 890-904, 2006.

[16] Maamari F, Andersen M, de Boer J, Carroll WL, Dumortier D, Greenup P. "Experimental validation of simulation methods for bi-directional transmission properties at the daylighting performance level," *ENERG BUILDINGS*, 38 (7), 849-68, 2006.

[17] Reinhart CF, Herkel S. "The simulation of annual daylight illuminance distributions - a state-of-the-art comparison of six RADIANCE-based methods," *ENERG BUILDINGS*, 32 (2), 167-87, 2000.

[18] Loutzenhiser PG, Maxwell GM, Manz H. "An empirical validation of the daylighting algorithms and associated interactions in building energy simulation programs using various shading devices and windows," *ENERGY*, 32 (10), 1855-70, 2007.

[19] Andersen M, Kleindienst S, Yi L, Lee J, Bodart M, Cutler B. "An intuitive daylighting performance analysis and optimization approach," *BUILD RES INF*, 36 (6), 593-607, 2008.

[20] Mardaljevic J. "Spatio-temporal dynamics of solar shading for a parametrically defined roof system," *Energy & Buildings*, 36 (8), 815-23, 2004.

[21] Perez R, Ineichen P, Seals R, Michalsky J, Stewart R. "Modeling daylight availability and irradiance components from direct and global irradiance," *SOL ENERGY*, 44 (5), 271-89, 1990.

[22] Hu J, Olbina S. "Simulation-Based Model for Integrated Daylighting System Design," *J COMPUT CIVIL ENG*, A4014003, 2013.

[23] Li DHW. "A review of daylight illuminance determinations and energy implications," *APPL ENERG*, 87 (7), 2109-18, 2010.

[24] Hu J, Olbina S. "Radiance-based Model for Optimal Selection of Window Systems," in *ASCE 2012 International Conference on Computing in Civil Engineering*, Clearwater Beach, FL, USA.

[25] Reinhart CF, Walkenhorst O. "Validation of dynamic RADIANCE-based daylight simulations for a test office with external blinds," *ENERG BUILDINGS*, 33 (7), 683-97, 2001.

[26] Kazanasmaz T, Gunaydin M, Binol S. "Artificial neural networks to predict daylight illuminance in office buildings," *BUILD ENVIRON*, 44 (8), 1751-7, 2009.

[27] Hu J, Olbina S. "Illuminance-based slat angle selection model for automated control of split blinds," *BUILD ENVIRON*, 46 (3), 786-96, 2011.

[28] Olbina S, Hu J. "Daylighting and thermal performance of automated split-controlled blinds," *BUILD ENVIRON*, 56 (0), 127, 2012.

[29] Hu J, Patel M. "Optimized Selection and Placement of Sensors using Building Information Models (BIM)," in *IES Annual Conference*, Pittsburgh, PA.