Local and GlobalLearning Methods for Predicting Power of a Combined Gas & Steam Turbine

HeysemKaya[1], Pınar Tüfekci[2], andFikret S.Gürgen[3]

***Abstract*—**In pursuit of finding accurate and efficient ways of predicting hourly electrical energy output, this study utilizes a dataset collected over 6 years (2006-2011) whose data points correspond to average hourly sensor measurements when the plant is set to work with full load. The input features are ambient temperature, relative humidity and ambient pressure which are known to be major factors in gas turbines as well as exhaust vacuum measured from steam turbine. We utilized conventional multivariate regression, additive regression, k-NN, feedforward ANN and K-Means clustering to form local and global predictive models. It is found that even with simple regression tools such as k-NN smoother it is possible to predict net yield with less than %1 relative error on the average. Using more sophisticated tools and proper preprocessing it is possible to significantly increase the performance.

***Keywords*—**Local Models, Gas Turbines, Power Prediction, Steam Turbines

I. Introduction& motivation

TO satisfy the increasing demand to electricity considering environmental aspects along with economic concerns, many power plants with gas turbine derivatives have been established all around the world [1]. The reliability and sustainability of a gas turbine depend highly on prediction of its power generation particularly when it is subject to constraints of high profitability and contractual liabilities.The effects of ambient conditions (i.e. ambient temperature (T), relative humidity (RH) and ambient pressure (AP)) are widely known and studied [1]-[10]. There are also relatively few studies [10]-[11] which utilize ambient conditions and machine learning tools such as Artificial Neural Networks (ANNs) for prediction of gas turbine variables the most important of which is electrical power (P_E). Therefore the aim of this study is not only analysis of individual factors but also finding the most appropriate ways of utilizing those factors to predictnet energy yield. The thermodynamic investigation of variables is beyond the scope of this paper. However, efficient intelligent learning methods in terms of ease-of-implementation, relatively low memory and computational needs are desired outputs.

One of the goals of this study for the power plant is to develop a reliablepredictor of following day's net energy yield per hour. The real life application requires the prediction of following day's hourly local ambient temperature values as a function of Meteorology Institute's max local temperature forecast and short-term temperature distribution over 24 hours. To meet this requirement it is feasible to use monthly average temperature distributions over the day where the data to calculate temperature statistics is available for the last 5 years.

Recently, Fast et al. [11] conducted a similar study with a small scale (22MW P_E) Combined Heat and Power (CHP) plant with three aforementioned ambient variables as inputs where the data is collected over one month. This study differs from [11] in two main aspects: 1) The dataset nature 2) Machine learning methods. First, the dataset is collected over dramatically longer period for a middle scale combined gas & steam turbine (CGST) whose P_E variance is larger than 22MW. Second, both simpler and more sophisticated machine learning methods are utilized and the results are compared using statistical tests.

The layout of the paper is as follows: section two describes the machine learning methods used, section three introduces the dataset and experimental results and section four concludes with recommendations for future work.

II. Tools and methods

A. k-Nearest Neighbor (k-NN) Smoother

k-NN is a widely used, simple and stable predictor which became popular after works of [12]. In machine learning, k-NN is an example of *nonparametric learning*while in Artificial Intelligence it is referred as *instance based learning*[13]. Nonparametric regression is called *smoothing* and the prediction is a *smooth*[14]. Therefore we refer to regression using k-NN as *k-NN smoothing.*The idea is "similar inputs result in similar outputs" [15]. In fact there is no training process (that's why it is classified as *lazy*) and when an unseen input is to be tested the closest *k*training samples are retrieved and the prediction is done based on their *k* outputs. In a classification task, majority voting is used i.e. the new

[1]Heysem.Kaya is pursuing his Phd degree in Bogazici University, Dept. of Computer Engineering ETA 26 (Medialab) TR-34342 Bebek / Istanbul, Turkey (corresponding author phone: +90 505 593 01 44; fax: +90 212 287 2461; e-mail: heysem@boun.edu.tr).

[2] Pınar Tüfekci is with Namık Kemal University,Dept. of Computer Engineering, SilahtarMahallesi, SinanDedeMevkiiCerkezkoyYolu 3. km, 59860 Corlu / Tekirdag, Turkey (e-mail: ptufekci@nku.edu.tr).

[3]Fikret S. Gürgen is with Bogazici University, Dept. of Computer Engineering, ETA 22 TR-34342 Bebek / Istanbul, Turkey (e-mail: gurgen@boun.edu.tr).

instance is labeled with the majority classout of k neighbors.In a regression task, the outputs of k neighbors are aggregated. In both tasks it is possible to weight the contributions manually by means of a *kernel.* A kernel can be seen as a smooth weight function which takes the distance between test instance and a neighbor as input. Later the aggregation can be normalized to the sum of kernel coefficients so that the weights sum up to 1. The most widely used kernel is Gaussian kernel:

$$K(u) = \left(\frac{1}{\sqrt{2\,\pi}}\right)^d exp\left[-\frac{\|u\|^2}{2}\right] \quad (1)$$

where exponent *d* is the dimensionality of the multivariate dataset. Then the smoothas weighted sum of k neighbors is attainedusing:

$$\hat{g}(x) = \frac{\sum_{t=1}^{k} K(x-x^t) r^t}{\sum_{t=1}^{k} K(x-x^t)} \quad (2)$$

B. MultivariateLinear Regression

Usually, polynomial regression with order higher than linearis used for univariate case. For multivariate datasets, however, it is not common to use polynomials of order higher than linear [13]. Using a linear model is more preferable due to two reasons: 1) It is simpler in terms of complexity. 2) It is easier to interpret the resulting model since the coefficients give direct information about the relative importance of variables. In fact using a *polynomial univariate regression* of an order *d*is a special case of*multivariate linear regression* where x_1=x, x_2=x^2,… ,x_d=x^d. Mathematically, if the linear model is [13]

$$r^t = g(x^t|w_0, w_1, \dots, w_d) + \epsilon$$
$$= w_0 + w_1 x_1^t + w_2 x_2^t + \cdots + w_d x_d^t + \epsilon \quad (3)$$

The error function can be stated as

$$E = \frac{1}{2}\sum_t (r^t - w_0 - w_1 x_1^t - w_2 x_2^t - \cdots - w_d x_d^t)^2 \quad (4)$$

Taking partial derivatives with respect to coefficients, w_j, j=0, …,d, *normalequations* are obtained. Defining a bias variable x_0 =1, let the bias-padded dataset be *X*, the weight vector be *w* and vector of outputs be *r*. Then d+1 normal equations can be written as

$$X^T X w = X^T r \quad (5)$$

Where w can be solved using

$$w = (X^T X)^{-1} X^T r \quad (6)$$

A linear regression model for a non-linear problem can be implementedusing mappings of input to a higher dimensional space by means of*basis functions* such as sin(x), ln(x) or exp(x). In statistics, this scheme is known as *models from linearization*[16]. This approach is commonly used in neural networks such as Support Vector Machines and Multi-Layer Perceptrons[13].

Another approach to a multivariate regression problem is fitting higher order univariate models and aggregating the individual estimates [13]. This approach is called *additive model* the details of which can be found in [17].

C. Feedforward Error Backpropagating ANN

An Artificial Neural Network (ANN) is simply a predictive model inspired from human brain. Similar to bionic NN, it has a layered architecture with *neurons* and *connections* among the layers. A basic ANN has three layers: *input*, *hidden* and *output* layers. The number of hidden layers and the number of hidden units (neurons) can be adjusted depending on problem complexity. It has been proven that using sufficient number of hidden units in one hidden layer has the same effect with using multiple hidden layers and such a one-hidden-layer ANN can learn any nonlinear function[13]. A neuron sums the input directed to it with the corresponding connection *weights* and transforms the sum using a non-linear activation function. This activation function is generally the sigmoid function.

$$\text{sigmoid(x)} = 1\,/[1 + \exp(-x)] \quad (7)$$

An important prerequisite for any activation function is *differentiability*. This property is used in *learning process*, i.e. updating weights based on the error. In a *feedforward* NN, units take the input only from lower layers, process and pass it to higher layers. The error, which is the difference between expected value and ANN output, is *backpropagated* to network using partial derivatives with respect to weights. Learning stops either when learning stabilizes, or the error is below a threshold or validation set accuracy does not increase for a specified number of *epochs.* An epoch is feeding the ANN with all training set instances in random order.

D.K-Means Clustering

K-Means clustering is a centroid based algorithm.In machine learning literature [13], K-Means clustering is used for several purposes: to label the unlabeled data, to map data into a lower dimensional space or to fit local models. This study utilizesK-Means for fitting local models to clusters. Since clustering is stochastic due to random initialization of *means*, it is necessary to aggregate the suitable models attained from various clusterings for better approximation. In this approach the training set is clustered and for each cluster a model (e.g. an ANN) is trained. Later when a test input is to be predicted, the model(s) associated with nearest mean to this input is/are used.

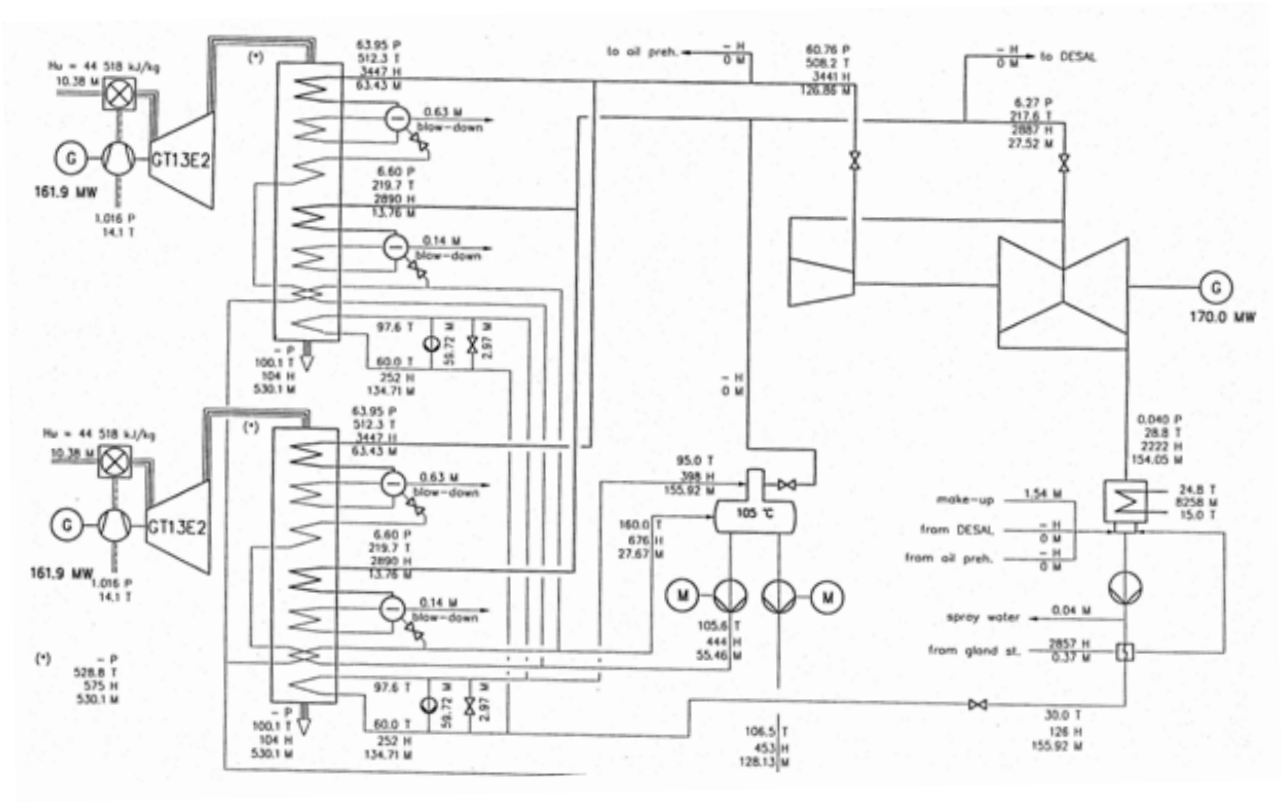

Fig. 1: Plant Layout

III. EXPERIMENTAL RESULTS

The ML experiments are carried out with k-NN, ANN, and K-Means implementations in MATLAB® [18], its NN Toolbox [19]. Reader may also utilize free ML tool which implements the algorithms in [15].

A. Data Collection &Dataset Description

The dataset is composed of 9568 data points collected when the combined cycle plant[4] (see Fig. 1 for layout)is set to work with full load over 674 different days. The dataset spans a variety of ambient conditions over 6 years of operation. Table I provides simple statistics of the dataset. The covariance and correlation matrices can be found in Table II and III respectively.

TABLE I: BASIC STATISTICS OF DATASET

	T	V	AP	RH	P_E
min	1.81	25.36	992.89	25.56	420.26
max	37.11	81.56	1033.30	100.16	495.76
mean	19.65	54.31	1013.26	73.31	454.37
variance	55.54	161.49	35.27	213.17	291.28

[4] The name of donor power plant is kept confidental.

TABLE II: COVARIANCE MATRIX

	T	V	AP	RH	P_E
T	55.54	79.94	-22.46	-59.03	-120.59
V		161.49	-31.21	-57.92	-188.64
AP			35.27	8.63	52.55
RH				213.17	97.13
P_E					291.28

TABLE III: CORRELATION MATRIX

	V	AP	RH	P_E
T	0.84	-0.51	-0.54	-0.95
V		-0.41	-0.31	-0.87
AP			0.10	0.52
RH				0.39

The temperature is not found to be below zero, as it is found to vary between 1 and 37. The plant generates 495MW net P_E (after deducting the internal use around 7MW)in the most convenient conditions. The covariance statistics indicate that the variables are not independent. The highest correlation among input features is between T and V (0.84). The highest correlations with target variable are also observed with T (-0.95) and V(-0.87).

1) Effect of Ambient Temperature (T)

The effect of T on performance is the most widely studied subject about gas turbines [1]–[7]. This can be appreciated since T is the most influential factor showing a correlation

around -0.95 with P_E. [6]utilize only this ambient variable for predicting gas turbine performance.

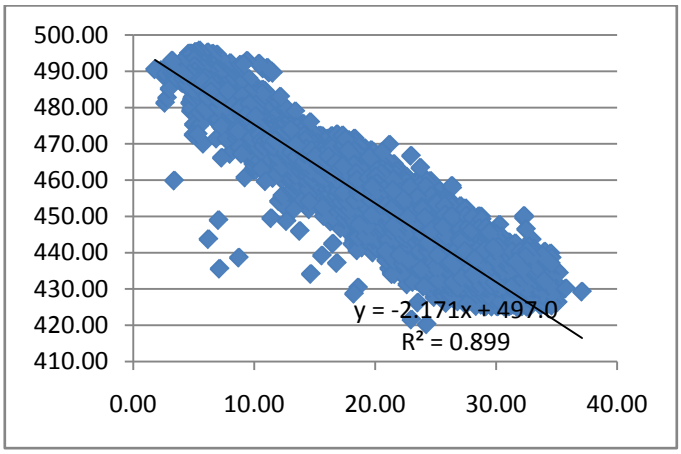

Fig. 2: Scatter diagram of T(˚C) vs. P_E(MW)

Fig. 2 shows the scatter diagram of T vs. P_E and the linear regression model fit to data. This illustration is done using the whole dataset for the purpose of preliminary investigation. The resulting predictive model

$$y = -2.1713x + 497.03 \quad (8)$$

can be interpreted as unit (˚C) increase in T accounts for a reduction of 2.17 MW in P_E. The performance reduction due to increase in temperature is known to stem from the decrease in the density of inlet air.

2) Effect of Ambient Pressure

Among the ambient variables, the second most influential one is AP [2]-[4]. However it doesn't have a strong correlation with the target variable sufficient for an individual prediction, as it can be observed from the scatter diagram shown in Fig. 3.

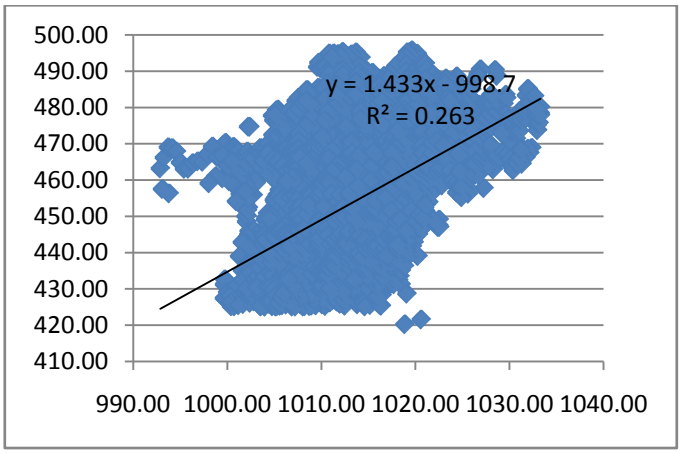

Fig. 3: Scatter diagram of AP (mb) vs. P_E (MW)

Similar to effect mechanism of T, AP is responsible for the density of inlet air. The slope of the linear regression function tells us that unit increase in AP corresponds to 1.4 MW increase in P_E.However this is a naïve estimation taking into account the other factors. When other factors remain constant it has been shown that P_E increases with increasing AP [4].

3) Effect of Relative Humidity

When other variables are kept constant, the performance is increased with RH [1]-[4],[8],[9]. However as it can be seen from Fig. 4, it is not self sufficient for prediction.

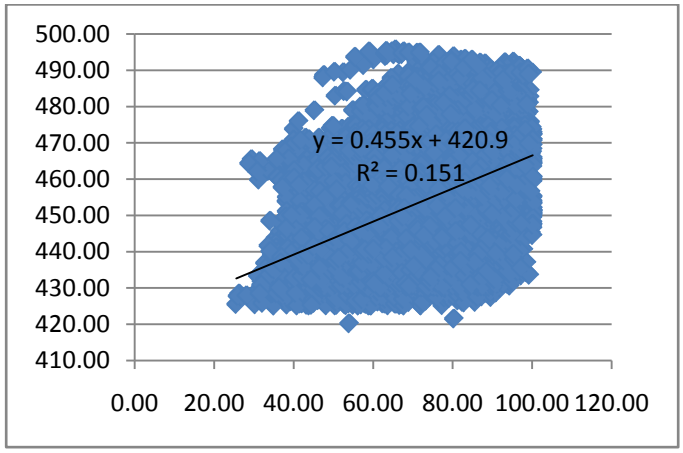

Fig. 4: Scatter diagram of RH vs. P_E (MW)

Higher relative humidity increases exhaust-gas temperature of gas turbine which leads to an in increase in the power generated by the steam turbine [4].

4) Effect of Vacuum

As seen in Fig. 1, the plant also employs a steam turbine which leads to considerable increase in total electrical efficiency.V variable (exhaust vacuum in cmHg) is collected from steam turbine and is found influential in its performance. When all other variables are kept constant exhaust-vacuum is known to have a negative influence on condensing-type turbine efficiency [20].As depicted in scatter diagram (Fig. 5), $P_E$is inversely proportional to V. It can also be observed that the slope of linear regression function is greater than those of RH and AP but less than T.

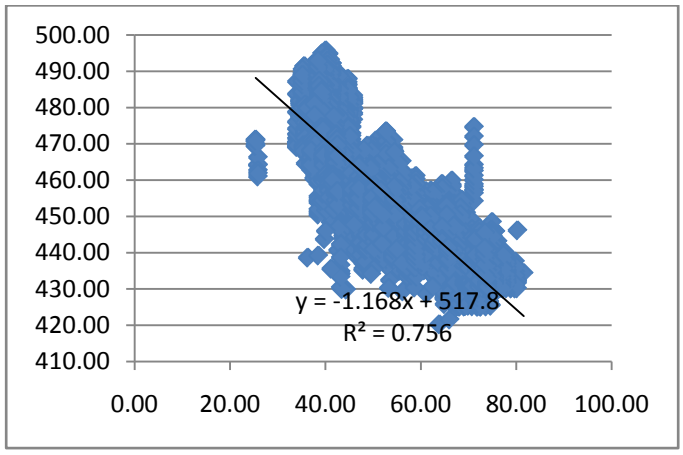

Fig 5. Scatter diagram of V(cmHg) vs. P_E (MW)

B. Experiments

After collecting preliminary statistical data, predictive models were tested. First of all simple k-NN smoother without

kernel weighting was applied with leave-one-day-out setting. The data points corresponding to a specific day was used for validation and the rest used as training; this was applied for all days. Since the real life application is predicting next day's hourly P_E, this setting reflects the most realistic case.

TABLE IV: PRELIMINARY K-NN TESTS WITH LEAVE-ONE-DAY-OUT

k	5	10	15	20	25
MRE(%)	0.82	0.79	0.78	0.77	0.77

The preliminary k-NN tests (Table IV), indicate less than %1 error for all k values tested. It also implies that the mean relative error decrease with increasing k up to 20.

Since the number of input features is four and the number of non-empty subsets is relatively small (2^4 -1=15), it was computationally feasible to test all feature subsets, to see the individual and cooperative validation set accuracies. Table V enlists the leave-one-day-out validation set performance both as Mean Relative Error (MRE) and Mean Absolute Error (MAE) for all feature subsets with k=20.

TABLE V: K-NN TESTS FOR ALL FEATURE SUBSETS (K=20)

Subsets	T	V	AP	RH	MRE(%)	MAE(MW)
1	1	0	0	0	0.92	4.19
2	0	1	0	0	1.61	7.38
3	1	1	0	0	0.81	3.67
4	0	0	1	0	2.58	11.79
5	1	0	1	0	0.91	4.11
6	0	1	1	0	1.30	5.97
7	1	1	1	0	0.78	3.55
8	0	0	0	1	2.82	12.89
9	1	0	0	1	0.84	3.80
10	0	1	0	1	1.33	6.09
11	1	1	0	1	0.78	3.53
12	0	0	1	1	2.30	10.54
13	1	0	1	1	0.85	3.87
14	0	1	1	1	1.24	5.67
15	1	1	1	1	0.77	3.51

Tests with subsets imply that the best accuracy is obtained when all features are used. As expected, T gives the highest individual predictive accuracy. Also collective performance of T and V is found significantly better than individual performances. If dimensionality reduction by feature selection is intended, any of the remaining two ambient variables can be used besides T and V.

For subsequent tests, in order to compare performance of different learning methods, 5x2 cross-validation [21] was applied. In this scheme, the dataset is randomly shuffled 5 times and each of them used in 2-fold CV. The resulting validation set performances of size 10 are used for statistical significance test. The results indicated that both preprocessing and using kernel smoothing significantly increase performance. MATLAB's ANN Toolbox uses *mapminmax* function to normalize data into [-1,+1] range. This preprocessing does not change performance of k-NN, however scaling with mean and standard deviation gives the desired outcome. Figure 6 depicts 1-Way ANOVA results (compare means) of k-NN Mean Square Error (MSE) for datasets using 1) no preprocessing, no kernel smoothing 2) only kernel smoothing 3) only preprocessing 4) both.

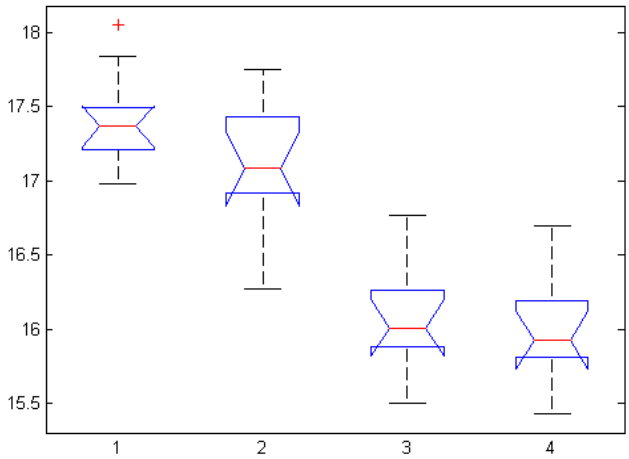

Fig. 6: ANOVA for k-NN MSE Performance of 4 Settings

It was observed that using preprocessing and kernel smoothing requires no more than 5 neighbors to aggregate.

Later tests with MATLAB's NN Toolbox were carried out with 'trainlm' training algorithm, one hidden layer with 10 hidden units. Learning stops when either of the following is reached (the default values are given in brackets):

- Max epochs (1000)
- Min error (1×10^{-5})
- Validation set accuracy does not increase (6 epochs)
- Max η
- Min gradient

Contrary to our expectation,in 5x2 CV setting the MSE performanceof 5-NN was found significantly better than fine-tuned ANN showing mean statistics 16.07 and 16.64, respectively.

Fitting linear regression functions for T in 5x2 CV yields an average MSE performance of 29.45 which is dramatically weaker than both k-NN and ANN. In the same CV setting, while additive regression was found to worsen predictive performance,a multivariate linear regression modelover all features was found to reduce MSE to 21.84.

Lastly, local models were applied as proposed in [22]. In the same 5x2 CV setting, for each validation set instance a specific local ANN was trained using the 100 nearest neighbors in training set. This means 5x2xN/2=95.680 ANNs were constructed using 100 training instances for each. After this computationally complex process, the MSE performance was found to be 19.99 which fell significantly behind global model performance. Later, a more efficient local approach with K-Means clustering is used. In this model first the training set is clustered and K models are trained for

corresponding clusters. When an instance from validation set is given, the system first finds the closest cluster (comparing withstored K mean vectors)and gives the prediction of associated ANN model. This setting yielded MSE performance of 15.48 with K=10. Since both ANN and K-Means are stochastic, ensemble of local models using different number of ensembles are tested. As anticipated, the ensemble smoothing yielded significantly better performance.

Table 5 enlists MSE performances of local models (LM). Here, LM1 is k-NN + ANN model with k=100, LM2 is K-Means + ANN with K=20, LM3 is a K-Means + ANN ensemble of population 3 with K=20, LM4 is a K-Means + ANN ensemble of population 5 with K=20 and LM5 is a K-Means + ANN ensemble of population 3 with K=10. As it can be seen local models with clustering yield better results with ensembling.

Table VI: 5x2 CV Performances for Local Models

	LM1	LM2	LM3	LM5	LM6
F11	20.95	20.08	17.86	15.33	15.04
F21	19.77	20.51	15.71	14.98	14.98
F31	20.03	18.43	15.94	15.35	14.94
F41	20.08	19.09	16.59	14.97	14.46
F51	19.36	19.04	16.10	15.38	15.11
F12	19.46	27.50	15.65	15.23	15.21
F22	19.72	18.09	16.27	15.59	16.04
F32	19.77	19.34	16.04	15.50	15.27
F42	20.57	18.68	16.88	16.09	18.32
F52	20.21	20.94	16.11	15.90	15.50
Mean	**19.99**	**20.17**	**16.32**	**15.43**	**15.49**

IV. CONCLUSION

The aim of this study was to investigate efficient and accurate methods for predicting electrical power to be generated. The additive models and linear regression were found to be with mediocre performance compared to ANN. Fine-tuned k-NN yielded better results than fine-tuned ANN though the difference was not dramatic. When test speed and memory need is important ANN is more suitable than k-NN. Local models without regard to temporal data were applied and found significantly better. However, there is a trade-off between complexity and performance. When dataset is large and variables are highly diverse in specific periods (e.g. summer and winter) local models could be more beneficial.

For ANN applications, tools such as MATLAB could be utilized. However k-NN implementation is simple and does not require a sophisticated tool. Fine-tuning of k and appropriate preprocessing help attaining desired performance.

As a future work on this problem, other learning models such as K-Star, Regression Trees, Support Vector Machines and their combinations could be used to provide a plethora for comparing predictive accuracy and learner efficiency.

REFERENCES

[1] J. J. Lee, D. W. Kang, T. S. Kim, "Development of a gas turbine performance analysis program and its application" ,*Energy*, vol. 36, is. 8, pp. 5274-5285, Aug. 2011

[2] F. J. Brooks, "GE Gas Turbine Performance Characteristics", Available online: http://143.107.98.150/Silvio/PME2517/GEgasTurbine.pdf, Accessed : Dec. 2011

[3] CAST, "Fundamentals Of Gas Turbine Engines", Available Online: http://www.cast-safety.org/pdf/3_engine_fundamentals.pdf, Accessed : Dec. 2011

[4] F. R.P. Arrieta, E. E. S. Lora. "Influence of ambient temperature on combined-cycle power-plant performance", *Applied Energy*,vol. 80, pp. 261–72, Mar. 2005

[5] F. Basrawi, T. Yamada, K. Nakanishi, S. Naing , "Effect of ambient temperature on the performance of micro gas turbine withcogeneration system in cold region", *Applied Thermal Engineering,vol. 31, is. 6-7*, pp. 1058-1067, May 2011

[6] A. De Sa, S. Al Zubaidy, "Gas turbine performance at varying ambient temperature",*Applied Thermal Engineering*, vol. 31, is. 14-15, pp. 2735-2739, Oct. 2011

[7] H. H. Erdem, S. H. Sevilgen, "Case study: Effect of ambient temperature on the electricity production and fuel consumption of a simple cycle gas turbine in Turkey", *Applied Thermal Engineering*, vol. 26, is. 2-3, pp. 320-326, Feb. 2006

[8] W. Yang, M. Su, Influence of moist combustion gas on performance of a sub-critical turbine, *Energy Conversion and Management*, vol. 46, is. 5, pp. 821-832, Mar. 2005.

[9] R. Lugo-Leyte, J.M. Zamora-Mata, M. Toledo-Velázquez, M. Salazar-Pereyra, A. Torres-Aldaco, "Methodology to determine the appropriate amount of excess air for the operation of a gas turbine in a wet environment Original Research Article",*Energy*, vol. 35, is. 2, pp. 550-555, Feb. 2010

[10] F. J. E. Aguilar, M. T. García, E. C. Trujillo, J. A. B. Villanueva, F. J. F. Ojeda, "Prediction of performance, energy savings and increase in profitability of two gas turbine steam generator cogeneration plant, based on experimental data", *Energy*, vol. 36, is. 2, pp. 742-754, Feb. 2011.

[11] M. Fast, M. Assadi , S. Deb , "Development and multi-utility of an ANN model for an industrial gas turbine", *Applied Energy*,volume 86, is. 1, pp. 9-17, Jan. 2009.

[12] D. W. Aha, D. Kibler and M. K. Albert, "Instance-Based Learning Algorithms", *Machine Learning*, vol.6, 1991, pp. 37-66.

[13] E. Alpaydın, *Introduction to Machine Learning*, Cambridge, MA: MIT Press, 2010.

[14] W. Härdle, *Applied Nonparametric Regression*, Cambridge, UK: Cambridge University Press, 1990.

[15] S. Russell and p. Norvig, *Artificial Intelligence: A Modern Approach* ,3rd ed., Prentice-Hall, 2010.

[16] R. Shiavi, *Introduction to Applied Statistical Signal Analysis*, 3rd ed, Acedemic Press, 2007.

[17] T. J. Hastie and R. J. Tibshirani,*Generalized Additive Models*, London: Chapman and Hall, 1990.

[18] MATLAB, ver. 7.11.0, 2010.

[19] MATLAB NNToolboxUser Guide, Available Online: http://www.mathworks.com/help/pdf_doc/nnet/nnet_ug.pdf, Accessed: Dec. 2011.

[20] Improve Steam Turbine Efficiency, Available Online: http://www.iffco.nic.in/applications/brihaspat.nsf/0/fddd5567e90ccfbde52569160021d1c8/$FILE/turbine.pdf, Accessed: Jan. 2012.

[21] T. G. Dietterich, "Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms",*Neural Computation*, vol. 10, pp. 1895-1923, 1998.

[22] L. Bottou, V. Vapnik, "Local Learning Algorithms", *Neural Computation*, vol. 4, pp. 888-900, 1992.