Prediction of land surface temperature using machine learning

[a] *Civil Engineering Department, McGill University, Montreal, Quebec, Canada*

ABSTRACT

To interpolate the missing land surface temperature, 17 cases of machine learning models were developed. Three types of machine learning methods, sparse Gaussian Process (GP), random forest, and decision tree, were utilized. The influence of kernel function, adding of elevation data, and type of machine learning on model performance were discussed. The results demonstrate that random forest model is the best model; Elevation data makes no contribution to the model performance; sparse GP models' performance is not satisfactory for its high time complexity and limitation of computer memory.

1. Introduction

The aim of this research is to build the probabilistic interpolation models for the land surface temperature data with shape 100 × 200 × 31. The data was accessed from MODIS platform. The Training data contains 494762 data points and the test data contains 85942 points. The figures for training data and test data in days are shown in Fig. 1 and Fig. 2. The latitude is divided into 100 points from 35 to 40 degrees and longitude is divided into 200 points from -115 to -105 degrees. And there are 31 days in the dataset. Moreover, there is no overlap between the training data and the test data so only the training data is used in training models and testing data is used to evaluate the performance of each model. Moreover, it is found that the temperature distribution each day is similar. There is an M shape in the right top corner of the area. Therefore, in this research, the elevation data are induced in prediction models and its impact on the model performance is studied.

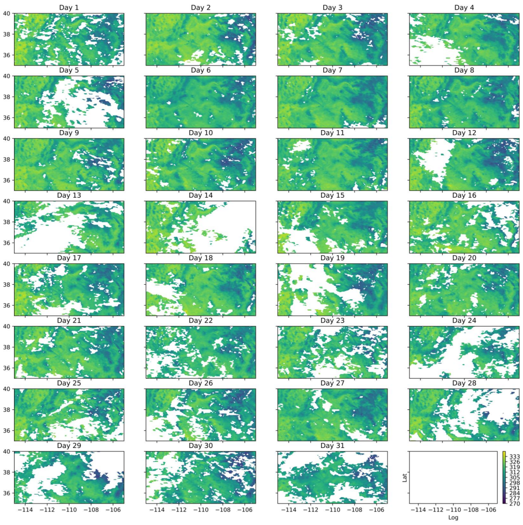


Fig. 1. Distribution of training data

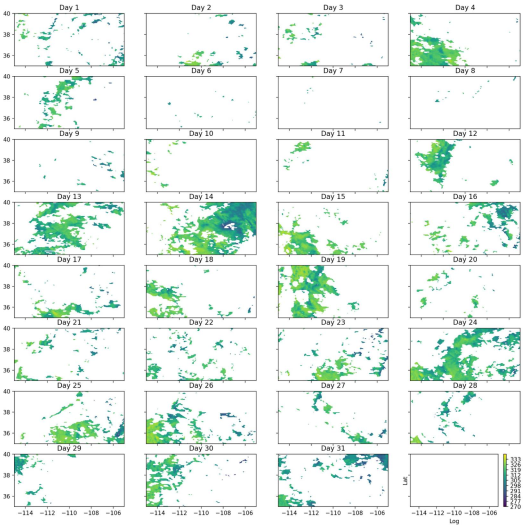


Fig. 2. Distribution of test data

As nonparametric models, GP aims to infer the latent function for the distribution of given available training data[1]. The functional space is defined by mean $m(x)$ and the kernel $k(x, x')$. Given n training points, the time complexity of standard GP $O(n^3)$ is high, and therefore it is the most weakness of the standard GP[2]. To address this problem, the sparse GP using m inducing points is proposed. Its training complexity is $O(nm^2)$, which speeds up the computing[3]. To achieve this goal, there are several steps:

1) selecting a subset of training data call inducing points, resulting in a smaller kernel matrix.

2) using sparse cornels by removing low correlation entries of full kernel function.

3) using sparse approximation which is also called low-rank representation.

A decision tree model contains three kinds of components: decision modes, chance nodes, and end notes[4]. It is a tree-shape decision rules combination and the group of the data is decided by of series of decision rules and the models are optimized by the accuracy of the models. As for the random forest, it is an ensemble of several machine learning models[5-6]. As a simple example, a random forest model can be a combination of three decision tree models. When predicting the output, the output values of each decision tree are weighted by [0.2, 0.4.0.4], and the sum of the weight multiplied by decision tree output is considered as the output of the random forest model.

In this research, the first section introduces machine learning including sparse GP, decision tree, and random forest. The second section demonstrates the acquisition of elevation data, prediction methods, and case details. In the third section, by comparing the R^2 and RMSE of each case, the influence of the sparse GP kernel, introduction of elevation data, and types of machine learning methods are discussed. Finally, the conclusion is concluded.

2. Methodologies

Three main kinds of methodologies were applied: sparse gaussian process, random forest, decision tree models. The model performance was compared by coefficient of determination (R^2), Root Mean Square Error (RMSE). The influence of kernel function on the sparse GP model performance are discussed. Meantime, the impact of inducing elevation on random are also discovered.

2.1. Elevation data acquisition

As shown in Fig. 1, it is obvious that there are similar distribution patterns between the temperature figures each day. Taking the temperature of Day 6 as an example, the temperature and terrain figures for the research area are shown in Fig. 3 and Fig.4 respectively. The M-shape in the right top corner can be found in both two figures. Moreover, the low temperature area in Fig. 3 overlaps the mountain area in Fig. 4. Therefore, the mountain area which have higher elevation tends to have lower temperature, which is understandable as the air in these areas is thin due to the low barometric pressure. Thin air has poor thermal insulation, leading to a large amount of heat loss[7].

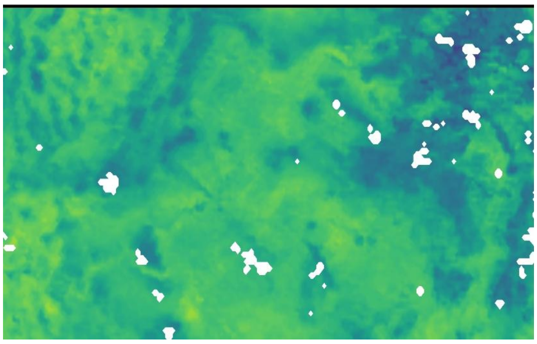


Fig. 3. Temperature distribution of Day 6

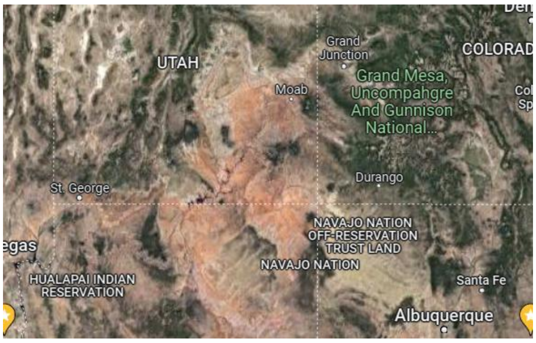

Den
Grand Junction
UTAH
COLORAD
Moab
Grand Mesa, Uncompahgre And Gunnison National...
Col Sp
Durango
St George
NAVAJO NATION OFF-RESERVATION TRUST LAND
egas
NAVAJO NATION
HUALAPAI INDIAN RESERVATION
Santa Fe
Albuquerque

Fig. 4. Terrain map

The elevation data are acquired form the Google Earth pro. The sampling space is 3.88 km. There are 240×128 data points in the whole area. As the point gird in given data is 100 × 200. Hence the elevation for each point is calculated in by the griddata function in Python. The elevation map for the area is shown below. The similar pattern still can be seen in this Fig. 5, which demonstrates that elevation have high relation with the ground temperature. Taking Day 1 as an example, the temperature-elevation relation is shown in Fig. 6.

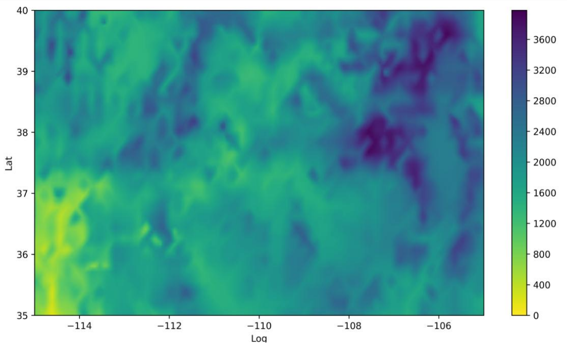


Fig. 5. Elevation distribution

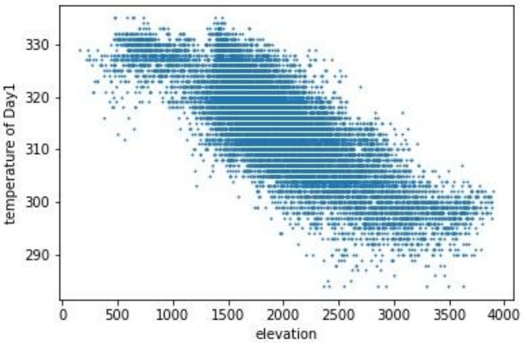
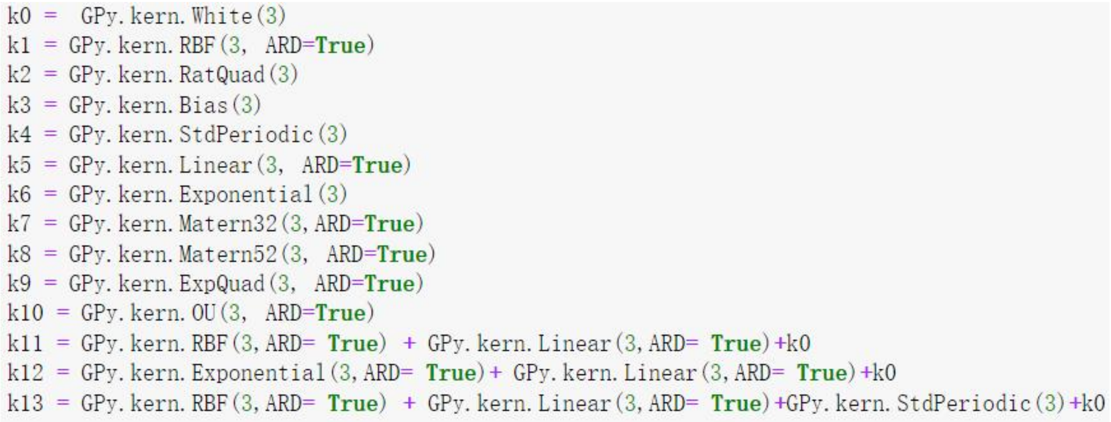


Fig. 6. Elevation-temperature relationship in Day1

2.2. Sparse gaussian process

In this research, GPy modules is applied to build sparse gaussian process analysis. This module function selects the inducing points, reconstructs the data, reselect the induing points in each iteration by calculating the model performance. The maximum iteration is set to be 100, and the inducing points are set to be 300 points for each case. The Before conducting the sparse GP, the values of temperature are subtracted by the mean of the temperature as the mean of the GP is set to be zero automatically[8].

The influence of kernel function and dimension of input feature are studies. There are 13 kinds of kernel are applied and these models' performances are compared. The kernels include Radial basis function (RBF)[9], RatQuad[10], Bias[11], StdPeriodic[12], Linear[13], Exponential[14], Matern32[15], Matern52[8], ExQuad, OU kernel[16] and the combination of these single kernels. In addition, the details for each kernel are shown in Fig. 7.



```
k0 =  GPy.kern.White(3)
k1 = GPy.kern.RBF(3, ARD=True)
k2 = GPy.kern.RatQuad(3)
k3 = GPy.kern.Bias(3)
k4 = GPy.kern.StdPeriodic(3)
k5 = GPy.kern.Linear(3, ARD=True)
k6 = GPy.kern.Exponential(3)
k7 = GPy.kern.Matern32(3,ARD=True)
k8 = GPy.kern.Matern52(3, ARD=True)
k9 = GPy.kern.ExpQuad(3, ARD=True)
k10 = GPy.kern.OU(3, ARD=True)
k11 = GPy.kern.RBF(3,ARD= True) + GPy.kern.Linear(3,ARD= True)+k0
k12 = GPy.kern.Exponential(3,ARD= True)+ GPy.kern.Linear(3,ARD= True)+k0
k13 = GPy.kern.RBF(3,ARD= True) + GPy.kern.Linear(3,ARD= True)+GPy.kern.StdPeriodic(3)+k0
```

Fig. 7. Definition of kernel functions

2.3. Other machine learning methods

In this research, the Pycaret.regression modules in Python was applied and trained 14 kinds of ML model automatically and compared the models by R2 as well as RMSE indicators. When applying Pycaret, the input data is [latitude, longitude, day], [latitude, longitude, day, elevation] respectively. The day parameter is defined as category data type and the other parameters are defined as Numeric data. The reason for setting the data type as categorical feature is that training data contains data from each day and there is no need to interpolate the missing data between each day.

The most two efficient machine learning models in this research is turned out to be random forest and decision tree from the Pycaret compare_models() function. Then the prediction of the test data is acquired by predict_model modules.

3. Results and discussion

3.1. Influence of Sparse GP kernel function

The R2 and the RMSE for Sparse GP models with different kernel is shown in Table 1. From the table, it is found that the performance of models is highly relative to the type of kernel. The R2 of the models range from -0.01 to 0.66. The RMSE is range from 4.98 to 8.57. The kernels contain RBF, Matern32, Martern 52 kernel show better performance. But the bias and StdPeriodic kernel makes no contribution to the models, indicating that there is no bias and periodic gaussian performance in the dataset. The predictive mean and standard deviation figure for case 7 is shown in Fig. 8 and Fig. 9. The error of temperature is all lower than 25 degrees. The distribution of error of temperature is shown in Fig. 10.

Table 1
Sparse GP scenarios 1-13 and their
Corresponding cross-validation RMSE and R^2

Scenarios	kernel	R^2	RMSE
1	RBF	0.65	5.05
2	RatQuad	0.43	6.44
3	Bias	-0.01	8.57
4	StdPeriodic	-0.01	8.57
5	Linear	0.21	7.59
6	Exponential	0.41	6.56
7	Matern32	0.66	4.96
8	Matern52	0.66	5.05
9	ExQuad	0.65	5.05
10	OU	0.65	5.08
11	REF+Linear	0.66	4.98
12	Exponential+Linear	0.65	5.08
13	REF+StdPeriodic+Linear	0.66	4.99

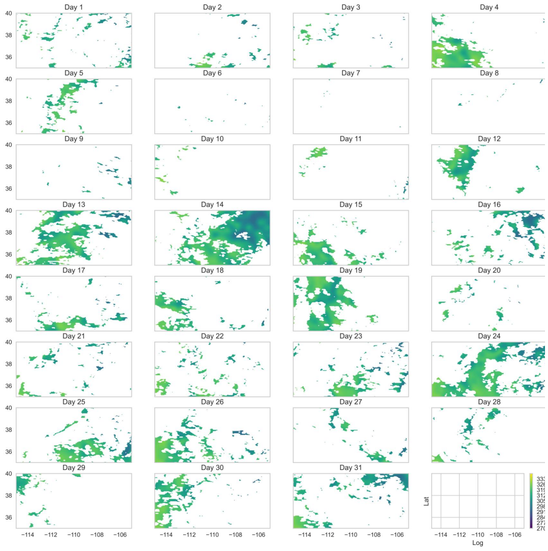


Fig. 8. Prediction mean of case 7

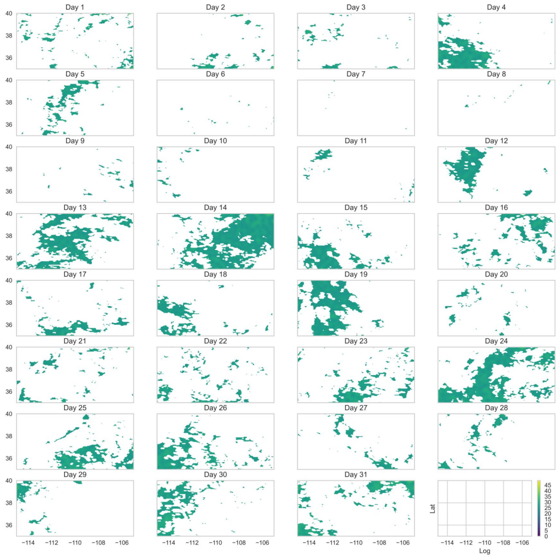


Fig. 9. Prediction error of case 7

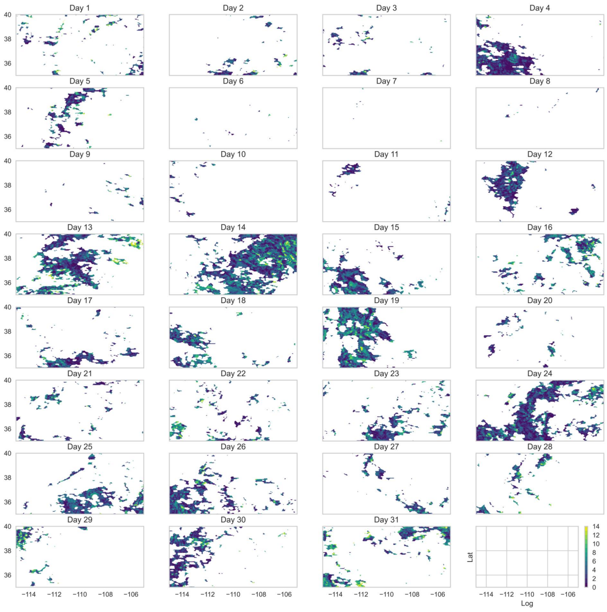


Fig. 10. Prediction mean error of case 7

3.2. Machine learning method comparation

The machine learning models built by training data without or with elevation data are shown in Table. 2 and Table. 3 respectively. From these two tables, the random forest and decision tree are the best two models for each case. The R^2 of these two models are all greater than 0.85. Besides, the R^2 of random forest training models are slightly greater than that of decision tree. Decision tree can classify the categorical features and this capability making it performs really well in this model as the temperature distribution is highly relative to the day, which is a categorical feature. The reason for that is random forest is ensambling machine learning model and therefore it combines several models and could get better models. The decision tree model can be regarded as a special random forest model which contains only one decision tree model and therefore its weight is 1.

Moreover, the R^2 for models with or without elevation is close to each other. It demonstrates that the elevation data does not contribute to the model. As it strongly connected to the temperature, the most likely explanation is that the training data have adequate information of the temperature distribution. The information provided by the elevation overlaps with the provided information and hence adds no additional information. Besides, though the elevation has strongly connected to temperature, different temperatures were recorded in same elevation, illustrating that it does not have all information of the temperature distribution.

Random forest and decision tree are utilized to interpolate the missing data, the RMSE and the R2 for each case are shown in Table 4. The interpolate models perform slightly worse than the training models as these two machine learning methods are overfitting. The reason for that is they have studied the noise or the irrelevant information of the training data and unable to generalize well in the missing data. Similar to the training models, the performance of random forest is slightly better than decision tree and introducing of the elevation data makes no difference in interpolation models. All in all, the random forest models not considering elevation turns out to be the best models. The predictive temperature distribution and prediction error for case 15 is shown in Fig. 11 and Fig. 12 respectively. The error of temperature is all lower than 18 degrees, indicating that random forest is suitable interpolate models for this research.

Table 2
Training model comparation without elevation data

Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
Random Forest Regressor	1.70	5.60	2.37	0.93	0.01	0.01	2.44
Decision Tree Regressor	1.73	7.36	2.71	0.90	0.01	0.01	0.82
Light Gradient Boosting Machine	3.42	19.34	4.40	0.75	0.01	0.01	0.48
Gradient Boosting Regressor	4.57	33.25	5.77	0.56	0.02	0.01	10.80
Linear Regression	5.59	47.88	6.92	0.37	0.02	0.02	0.93
Ridge Regression	5.59	47.88	6.92	0.37	0.02	0.02	0.09
Least Angle Regression	5.59	47.88	6.92	0.37	0.02	0.02	0.10
Bayesian Ridge	5.59	47.88	6.92	0.37	0.02	0.02	0.61
Orthogonal Matching Pursuit	5.79	51.36	7.17	0.33	0.02	0.02	0.11
Lasso Regression	5.89	53.06	7.28	0.30	0.02	0.02	0.54
Elastic Net	5.89	53.08	7.29	0.30	0.02	0.02	0.11
Huber Regressor	5.89	54.26	7.36	0.29	0.02	0.02	6.21
AdaBoost Regressor	6.19	54.52	7.38	0.28	0.02	0.02	11.52
Passive Aggressive Regressor	6.49	68.31	8.18	0.10	0.03	0.02	2.23
Lasso Least Angle Regression	7.00	76.12	8.72	0.00	0.03	0.02	0.10
Dummy Regressor	7.00	76.12	8.72	0.00	0.03	0.02	0.07

Table 3
Training model comparation with elevation data

Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
Random Forest Regressor	1.59	5.10	2.26	0.93	0.01	0.01	43.22
Decision Tree Regressor	2.08	9.78	3.13	0.87	0.01	0.01	0.82
Light Gradient Boosting Machine	3.42	19.36	4.40	0.75	0.01	0.01	0.49
Gradient Boosting Regressor	4.58	33.27	5.77	0.56	0.02	0.01	11.86
K Neighbors Regressor	5.07	41.88	6.47	0.45	0.02	0.02	1.29
Least Angle Regression	5.59	47.84	6.92	0.37	0.02	0.02	0.10
Bayesian Ridge	5.59	47.84	6.92	0.37	0.02	0.02	0.57
Ridge Regression	5.59	47.84	6.92	0.37	0.02	0.02	0.08
Linear Regression	5.59	47.85	6.92	0.37	0.02	0.02	0.73
Orthogonal Matching Pursuit	5.79	51.35	7.17	0.32	0.02	0.02	0.09
Lasso Regression	5.89	53.04	7.28	0.30	0.02	0.02	0.53
Elastic Net	5.89	53.06	7.28	0.30	0.02	0.02	0.09
AdaBoost Regressor	6.18	54.42	7.38	0.28	0.02	0.02	12.22
Huber Regressor	6.59	66.22	8.14	0.13	0.03	0.02	5.05
Lasso Least Angle Regression	7.00	76.04	8.72	0.00	0.03	0.02	0.10
Dummy Regressor	7.00	76.04	8.72	0.00	0.03	0.02	0.09
Passive Aggressive Regressor	10.94	176.24	13.10	-1.32	0.04	0.03	0.59

Table 4
Scenarios 14-17 and their corresponding cross-validation RMSE and R^2

Scenarios	Consider Elevation?	Model	R^2	RMSE
14	N	Decision Tree Regressor	0.79	3.87
15	N	Random Forest Regressor	0.86	3.24
16	Y	Random Forest Regressor	0.85	3.30
17	Y	Decision Tree Regressor	0.77	4.06

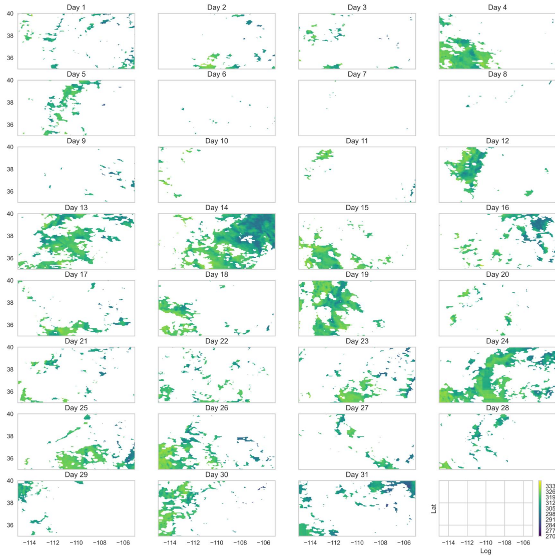


Fig. 11. Prediction mean of case 15

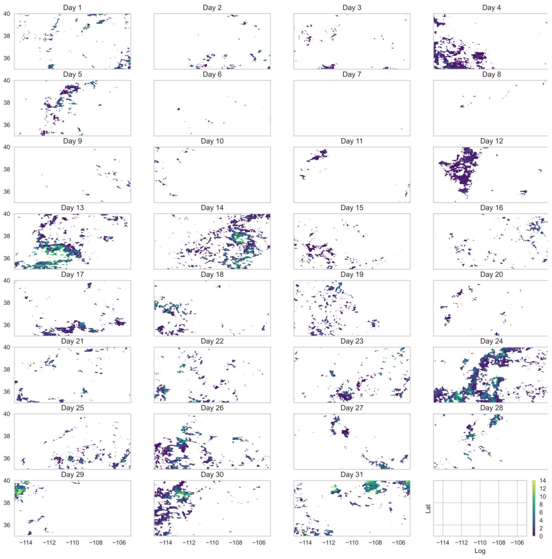


Fig. 12. Prediction error of case 15

3.3. Comparation of model type

From the previous result, the best model is the case 15 models, which is the random forest models. The main reason for random forest outperforms the gaussian process is that only 300 inducing points are selected in Sparse GP but the random forest considers all data points. The limitation of the inducing points resulting the losing of information and therefore can not find the precise latten function of gaussian distribution. The performance of the GP can be enhanced by increasing the inducing points or selected more efficient GP method.

Moreover, the training time of Sparse GP is much greater than that of the random forest and decision tree. The former takes around 30 minutes while the latter takes around 10 minutes and has better performances. This is due to the high time complexity of GP and also the limitation of the computer memory. The computation time cost of the GP is linearly increase when the inducing points increase and the laptop memory only allows 1000 inducing points computation. It is found that Matlab can perform greater inducing point calculation and thus becomes a better platform to conduct GP model training.

4. Conclusion

In this research, three kinds of machine learning models, sparse GP, decision tree, random forest were utilized to interpolate the missing data of ground temperature data. Sparse GP with different kernel function, random forest and decision trees models with different input were compared.

The best model is random forest model. This model has less computation time and predict the missing values better. Besides, the models with high to low performance is: random forest; decision tree; sparse GP with RBF, Matern32, Matern52, ExQuad, OU kernels. The adding of elevation data input makes no contribution to the model performance as it adds no extra information. Compared to other machine learning method, GP have higher time complexity. Moreover, the laptop memory is limited, leading to limitation of inducing data and sparse GP's low performance.

References

[1] H. Liu, Y.-S. Ong, X. Shen, and J. Cai, "When Gaussian Process Meets Big Data: A Review of Scalable GPs," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 31, no. 11, pp. 4405–4423, Nov. 2020, doi: 10.1109/TNNLS.2019.2957109.

[2] R. Bostanabad, Y.-C. Chan, L. Wang, P. Zhu, and W. Chen, "Globally approximate gaussian processes for big data with application to data-driven metamaterials design," *Journal of Mechanical Design*, vol. 141, no. 11, 2019.

[3] A. Smola and P. Bartlett, "Sparse greedy Gaussian process regression," *Advances in neural information processing systems*, vol. 13, 2000.

[4] E. Pekel, M. C. Akkoyunlu, M. T. Akkoyunlu, and S. Pusat, "Decision tree regression model to predict low-rank coal moisture content during convective drying process," *International Journal of Coal Preparation and Utilization*, vol. 40, no. 8, pp. 505–512, 2020.

[5] M. Belgiu and L. Drăguţ, "Random forest in remote sensing: A review of applications and future directions," *ISPRS journal of photogrammetry and remote sensing*, vol. 114, pp. 24–31, 2016.

[6] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, no. 2, pp. 197–227, 2016.

[7] S. Khandelwal, R. Goyal, N. Kaul, and A. Mathew, "Assessment of land surface temperature variation due to change in elevation of area surrounding Jaipur, India," *The Egyptian Journal of Remote Sensing and Space Science*, vol. 21, no. 1, pp. 87–94, 2018.

[8] I. Roman, R. Santana, A. Mendiburu, and J. A. Lozano, "Evolution of Gaussian Process kernels for machine translation post-editing effort estimation," *Annals of Mathematics and Artificial Intelligence*, vol. 89, no. 8, pp. 835–856, 2021.

[9] B. Mulgrew, "Applying radial basis functions," *IEEE signal processing magazine*, vol. 13, no. 2, pp. 50–65, 1996.

[10]Z. Zhu, J. Zhang, and J. Zou, "A multi-kernel based Gaussian process dynamic model for human motion modeling," in *2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, 2017, pp. 457–462.

[11]M. C. Jones and D. Signorini, "A comparison of higher-order bias kernel density estimators," *Journal of the American Statistical Association*, vol. 92, no. 439, pp. 1063–1073, 1997.

[12]Y. Lin and L. D. Brown, "Statistical properties of the method of regularization with periodic Gaussian reproducing kernel," *The Annals of Statistics*, vol. 32, no. 4, pp. 1723–1743, 2004.

[13]M. Masjed-Jamei and I. Area, "Error bounds for Gaussian quadrature rules using linear

kernels," *International Journal of Computer Mathematics*, vol. 93, no. 9, pp. 1505–1523, 2016.

[14]M. Caputo and M. Fabrizio, "Applications of new time and spatial fractional derivatives with exponential kernels," *Progress in Fractional Differentiation & Applications*, vol. 2, no. 1, pp. 1–11, 2016.

[15]J. Fang, S. Liang, Z. Meng, and Q. Zhang, "Gaussian process with graph convolutional kernel for relational learning," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 353–363.

[16]C. Archambeau, D. Cornford, M. Opper, and J. Shawe-Taylor, "Gaussian process approximations of stochastic differential equations," in *Gaussian Processes in Practice*, 2007, pp. 1–16.