학사학위논문

(국문) 인공지능을 이용한 태양광 발전소 최적 입지선정

(영문) Optimal Location Analysis for Solar Power Plants

신 선 웅

한양대학교 공과대학

2021년 12월 14일

학사학위논문

(국문) 인공지능을 이용한 태양광 발전소 최적 입지선정

(영문) Optimal Location Analysis for Solar Power Plants

지도교수 정 재 원

이 논문을 공과대학 학사학위논문으로 제출합니다.

2021년 12월 14일

한양대학교 공과대학

건축공학부

신 선 웅

이 논문을 신선웅의 학사학위 논문으로 인준함

2021년 12월 14일

심 사 위 원 : 정 재 원 (인)

(지 도 교 수)

한양대학교 공과대학

Optimal Location Analysis for Solar Power Plants Using AI

Sunwoong Shin
Student of Hanyang University

December 14, 2021

Contents

1. INTRODUCTION

1.1 Background

Renewable energy is expanding many businesses and research related to solar, hydro and wind energy, and domestic supply is accelerating further under government policy [6]. Among them, solar energy accumulates energy in solar cells using solar energy system and converts it into electricity to make electricity. These systems do not produce hazardous substances, do not require fuel, do not require any other noise and are easy to install and use for a long period of time. However, the exact evaluation of solar energy facilities has yet to be made, indicating that the problem of operational and power generation efficiency is [6]. Therefore, to increase the efficiency of solar energy, it is necessary to study proper regional evaluation, solar potential calculation, and performance prediction. Existing research was mainly conducted to evaluate the potential of solar power systems and maximize performance and efficiency, and recently, research has been proceeded to analyze the impact of solar power systems on the weather environment [13]. However, these studies did not use solar energy generation data effectively, and the subsequent use of shared solar energy generation data together would yield more accurate results than ever before. In addition, there are more weather stations that can observe solar radiation data than in the past, allowing active use of solar radiation data. Moreover, given geographical factors, large areas are needed to build power generation facilities. To build a solar power plant, a larger area is needed than limited domestic land, and it is essential to select the optimal power plant location that maximizes power generation efficiency. The territory of Korea consists of a vast mountainous region, accounting for 63% of the total land. Photovoltaic plants generally require a gentle slope of flat land, but these flat land are commonly used for urban and other commercial purposes. Flat land is steadily rising, forming high prices. Therefore, it would be important to select the appropriate areas based on geographical factors.

1.2 Research scope and method

In this study, we will collect weather data from Korean Weather Station across the country and utilize them as variables along with geographical and economic data. It is intended to analyze and evaluate the appropriate area of solar energy generation system installation using solar power generation data. After selecting a machine learning model suitable for training turbulent data, the machine starts training with local data and analyze the importance of how much each variable affects the results. After calculating the location score of solar power plants in unknown areas with variables that can significantly affect the results, the results will be analyzed compared to the actual power generation of solar power plants to examine whether the results are reasonable.

2. RELATED WORK

Prior to proceeding with the study, existing papers need to be reviewed. Studies have been conducted on the optimal location of solar power by geographical and climatic factors. Based on existing studies, it is necessary to select the optimal location for solar power plants through more logical and empirical analysis.

2.1 Installation Type

To examine the power generation performance and efficiency of solar power systems, power generation characteristics and performance by installation type were studied, and the effects of the operating temperature range and loss of solar cell modules were studied [4]. In addition, there are studies that analyze solar radiation only, such as the impact of shading on the distribution of buildings in urban areas [3]. However, in prior studies, there is a limitation that is difficult to compare with other regions because it is analyzed only in small areas. In addition, in Korea, there are many mountainous terrain and solar observation stations are not densely distributed, so an analysis of the unmeasured land is needed.

2.2 Geographical Conditions

In the case of gradients, it is usually installed in areas within 10 degrees of gradients when installing solar power plants [7]. According to the Ministry of Environment's pre-environmental review manual, freezing in winter and slope collapse are feared. Also, vertical views from surrounding roads can interfere with driver's eyes [8]. The south, southeast, southwest, and flat terrain is good for sunlight. In the case of the shading corridor, Azimuth was applied at 230° and Altitude at 10° as of the winter solstice day (December 22nd of the solar calendar) of the Korea Astronomical Research Institute. In the case of winter solstice, the year is the shortest of the year, and when the sun shines well, it shines well for 365 days.

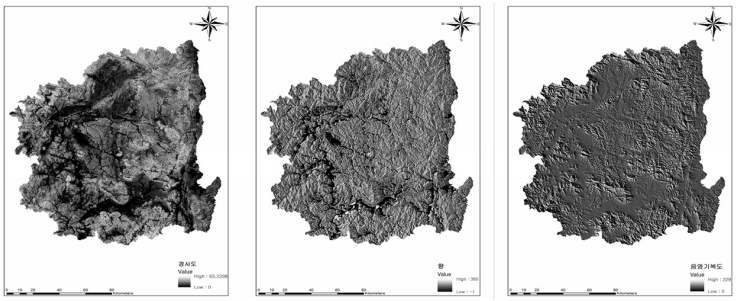
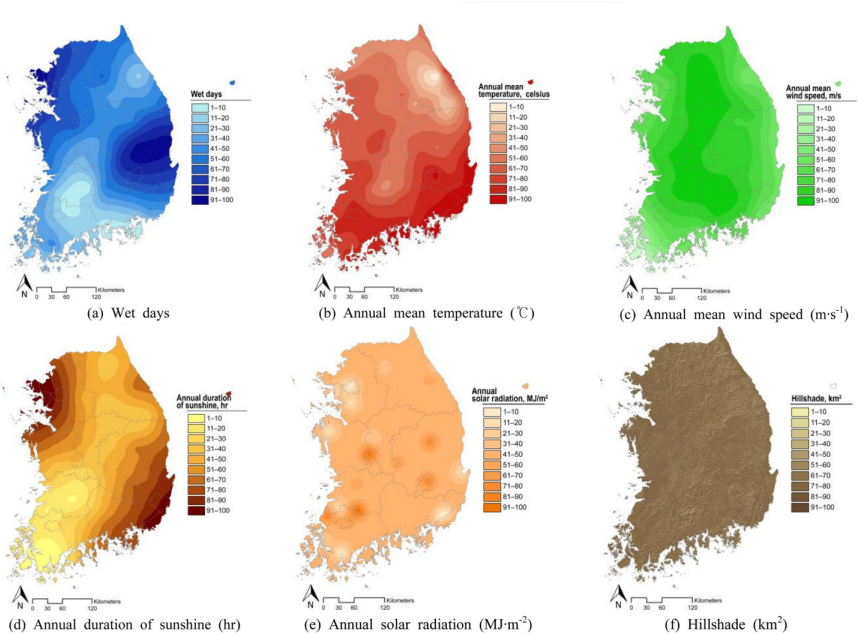


Figure 1: Terrain factor of solar power plants

Shading corridor is a map of the degree of shading by gradient and elevation, and gradient and elevation are reported as factors that greatly influence solar power [11].

2.3 Weather Conditions

The most important thing in solar power plants is climate. The factors commonly used in previous studies are solar radiation, mean temperature, wind speed, duration of sunshine, humidity. We could see how these factors correlate with solar power generation, but we did not know the close relationship between each factor. [10].



(a) Wet days (b) Annual mean temperature (℃) (c) Annual mean wind speed ($m \cdot s^{-1}$)

(d) Annual duration of sunshine (hr) (e) Annual solar radiation ($MJ \cdot m^{-2}$) (f) Hillshade (km^2)

Figure 2: Spatial distribution of the proxy variables

2.4 Economic Conditions

Climate factors such as solar radiation and average temperature are the most important to maximize the efficiency of solar power plants, but land purchase costs are also important because a lot of solar panels are needed to generate a lot of electricity. In the case of land purchase costs, the Ministry of Land, infrastructure and transport's official land price was expressed using the Creeping interpolation technique using Inc.Biz-gis' GIS DB score data (as of 2011) [9].

3. DATA ANALYSIS

The basis of data management is to collect, store, and organize the right data. The key to big data is not data collection, but how to use the collected data. The process of preprocessing to find and write high-quality data necessary for the paper is inevitable.

3.1 Weather Observation Data

In this study, to analyze the efficiency of solar power production based on the weather data of the entire Republic of Korea, the entire weather data provided by the Korean Weather Station was collected and analyzed. The collected data are weather data based on latitude and longitude, which can identify overall weather factors in Korea. It was necessary to preprocess weather data. Data that does not affect the value of solar power was deleted in advance. Empty data were loaded with zero values, and all transport data had to be linearly interpolated due to the large number of missing values.

```
def make_sub_array(start_date, location):
    queryParams = '?' + urllib.parse.urlencode(
        {quote_plus('ServiceKey') : ServiceKey1,
         quote_plus('pageNo') : '1',
         quote_plus('numOfRows') : '960',
         quote_plus('dataType') : 'JSON',
         quote_plus('dataCd') : 'ASOS',
         quote_plus('dateCd') : 'HR',
         quote_plus('startDt') : start_date,
         quote_plus('startHh') : '01',
         quote_plus('endDt') : datetime.strftime(pd.to_datetime(
    start_date)+pd.DateOffset(hours=960), '%Y%m%d'),
         quote_plus('endHh') : '00',
         quote_plus('stnIds') : location})

    response = urlopen(url + queryParams).read()
    response = json.loads(response)

    if response['response']['header']['resultMsg']==('NO_DATA'):
        return 'NoData'
    for i, data in enumerate(response['response']['body']['items']['
    item']):
        if i==0:
            obs_array = np.delete(
                np.array(list(data.values())), [1, 5, 7, 9, 11, 13, 17,
     19, 21, 24, 26, 27, 28, 30, 31, 33]).reshape(-1, 1)
        else:
            obs_array = np.hstack(
                [obs_array, np.delete(np.array(list(data.values())),[1,
     5, 7, 9, 11, 13, 17, 19, 21, 24, 26, 27, 28, 30, 31, 33]).reshape
    (-1, 1)])

    return obs_array.T
```

Listing 1: Custom crawling function python code

```
start_date_list = list(pd.date_range(start='20170101 01:00:00', end='
    20200701 00:00:00', freq='960H').astype(str))
for i, date in enumerate(start_date_list):
    start_date_list[i] = date.split(' ')[0].replace('-', '')

def make_df(location):
    obs_array = make_sub_array(start_date_list[0], location)
    if type(obs_array)==str:
        return 'NoData'
    sentence = ' ' + obs_array[0][2] + print(f'{sentence:=^56}')
```

```
    for i, date in enumerate(tqdm(start_date_list[1:])):
        obs_array = np.vstack([obs_array, make_sub_array(date, location
    )])

    obs_array[:, [4, 12, 13, 14]] = np.where(obs_array[:, [4, 12, 13,
    14]]=='', 0.0, obs_array[:, [4, 12, 13, 14]])
    obs_array = np.where(obs_array=='', np.nan, obs_array)

    obs_df = pd.DataFrame(obs_array, columns=col_list)
    obs_df[col_float_list] = obs_df[col_float_list].astype('float16')
    obs_df['time'] = pd.to_datetime(obs_df['time'])
    obs_df[['cloud']] = obs_df[['cloud']].interpolate()

    return obs_df
```

Listing 2: Custom concatenating function python code

```
obs_dict_local = {}
obs_dict_local_list = [105, 129, 192]
for loc_num in obs_dict_local_list:
    obs_dict_local[loc_num] = make_df(str(loc_num))
obs_dict_global = {}
obs_dict_global_list = np.setdiff1d(np.arange(90, 300), [105, 129, 152,
     175, 192])
for loc_num in tqdm(obs_dict_global_list):
    obs_dict_global[loc_num] = make_df(str(loc_num))
```

Listing 3: Main python code

We sampled the observation data of Busan. As shown in below table, we used features that can affect the power. Observation time, location number, location name, temperature, precipitation, wind speed, wind direction, humidity, vapor pressure, dew point, air pressure, sea pressure, sunshine, solar radiation, rainfall, snowfall, electricity, cloud opacity, visibility are features we chose.

	time	loc_num	loc_name	temp	precipitation	wind_speed	wind_direction	humidity	pressure_vapor	dew_point	pressure_local	pressure_sea	sunshine	radiation	snow	cloud	air_opacity	temp_surf
0	2017-01-01 01:00:00	159	busan	3.500000	0.000000	3.599609	360.0	67.0	5.300781	-2.000000	1020.5	1029.0	0.0	0.000000	0.0	0.0	1438.0	0.500000
1	2017-01-01 02:00:00	159	busan	3.599609	0.000000	4.000000	360.0	67.0	5.300781	-1.900391	1020.5	1029.0	0.0	0.000000	0.0	0.0	1572.0	0.399902
2	2017-01-01 03:00:00	159	busan	3.000000	0.000000	1.500000	360.0	69.0	5.199219	-2.099609	1021.0	1030.0	0.0	0.000000	0.0	0.0	1407.0	-0.500000
3	2017-01-01 04:00:00	159	busan	2.800781	0.000000	0.399902	0.0	67.0	5.000000	-2.699219	1020.5	1030.0	0.0	0.000000	0.0	0.0	1392.0	-0.799805
4	2017-01-01 05:00:00	159	busan	2.699219	0.000000	3.300781	320.0	68.0	5.000000	-2.599609	1019.5	1028.0	0.0	0.000000	0.0	0.0	1335.0	-0.399902
30715	2020-07-03 20:00:00	159	busan	20.593750	0.000000	3.599609	50.0	81.0	19.593750	17.203125	1000.0	1008.0	0.0	0.029999	0.0	2.0	1704.0	21.296875
30716	2020-07-03 21:00:00	159	busan	20.296875	0.000000	2.500000	50.0	86.0	20.406250	17.796875	1000.5	1008.5	0.0	0.000000	0.0	10.0	1587.0	21.203125
30717	2020-07-03 22:00:00	159	busan	19.796875	1.299805	2.000000	20.0	91.0	20.906250	18.203125	1001.0	1009.0	0.0	0.000000	0.0	10.0	536.0	20.796875
30718	2020-07-03 23:00:00	159	busan	18.906250	3.800781	3.300781	50.0	95.0	20.593750	18.000000	1000.0	1008.0	0.0	0.000000	0.0	10.0	1244.0	20.296875
30719	2020-07-04 00:00:00	159	busan	18.796875	0.000000	2.000000	70.0	92.0	19.906250	17.406250	1000.0	1008.0	0.0	0.000000	0.0	7.0	1953.0	20.000000

Figure 3: Sampled observation data

Correlation Analysis is a technique for analyzing whether there is an alignment relationship between two variables. The correlation coefficient resulting from the correlation analysis is a number with a range of ”-1 to +1”, representing the linearity of the two variables. A large correlation coefficient means that the linearity between the

two variables is very high and a low correlation coefficient means that the linearity between the two variables is weak. Correlation analysis of the variables confirms that air pressure and sea pressure, vapor pressure and dew point, air temperature and surface temperature come close to 1, which means that they are almost identical and therefore do not need to be regressed between them.

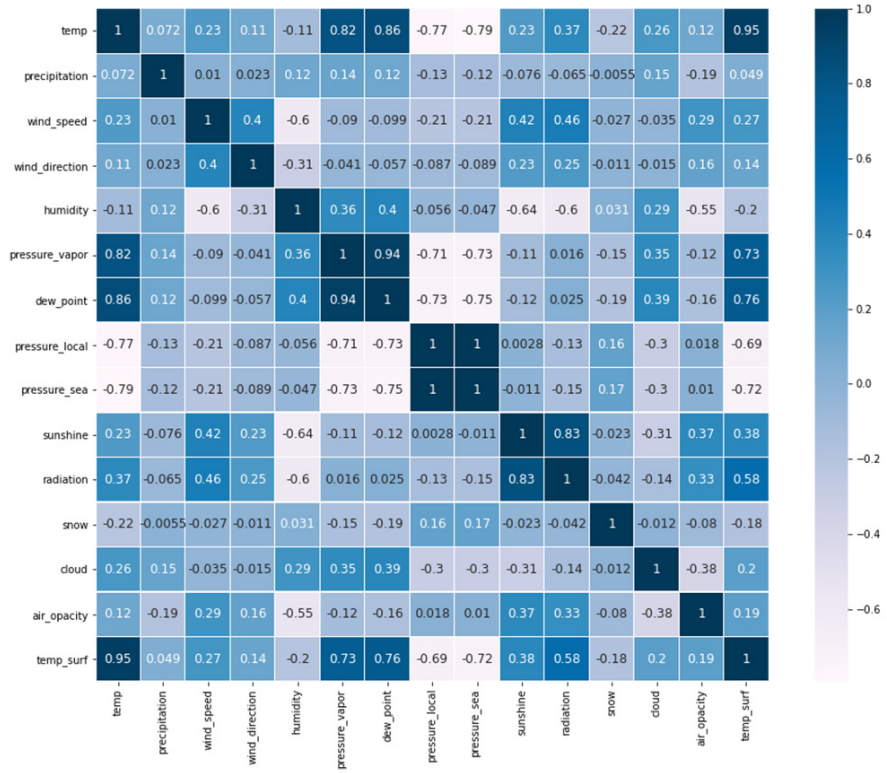
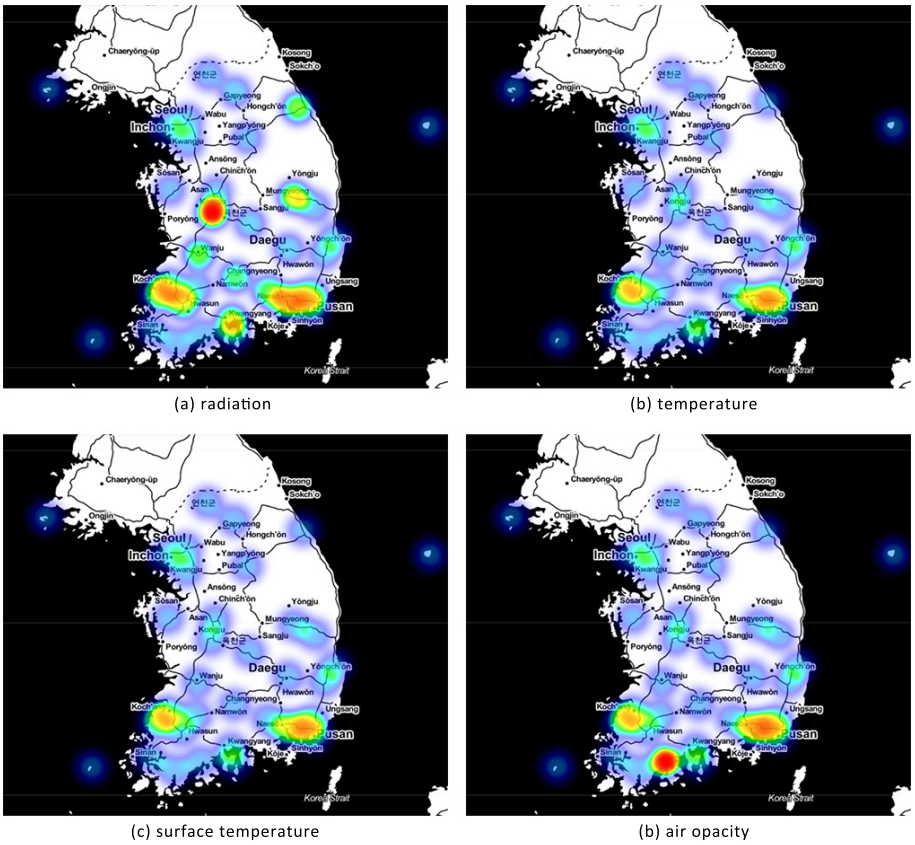


Figure 4: Correlation of weather observation features

We made a heatmap by sampling variables that are more likely to affect power generation relatively much. When analyzing the solar radiation, temperature, and visibility heatmap, it is predicted that the power generation in the south will be greater than in the north. As the correlation analysis suggests, the heatmap of air temperature heatmap and surface temperature heatmap is almost identical. Since the region with zero solar radiation data was removed and visualized, there are many areas with no data at all. So, it is necessary to use the kriging interpolation method to predict the value of the unknown's attributes using the ambient value [12]. Unlike Inverse Distance Weighted (IDW), which uses functions with respect to simple distances, statistical distances are used to predict values of unknown points by linear combinations between point infor-

mation. It also has more accurate features than other interpolation methods because it reflects the correlation strength between neighboring values, not just using distance.



(a) radiation (b) temperature

(c) surface temperature (b) air opacity

Figure 5: Sampled spacial distribution heatmap

3.2 Solar Power Generation Data

Target values are needed for model learning, and solar power generation data could be easily received from public data portals. However, there were constraints: there should not be many missing values, and the weather data provided by the Korea Weather Station should match the region. Power generation data exploratory data analysis was essential, and insight was sought through various visualizations. By visualizing amount of power generation per month, data trend of amount of power generation analyzed. It was possible to confirm that regional power generation was independent, and different aspects could be seen every day.

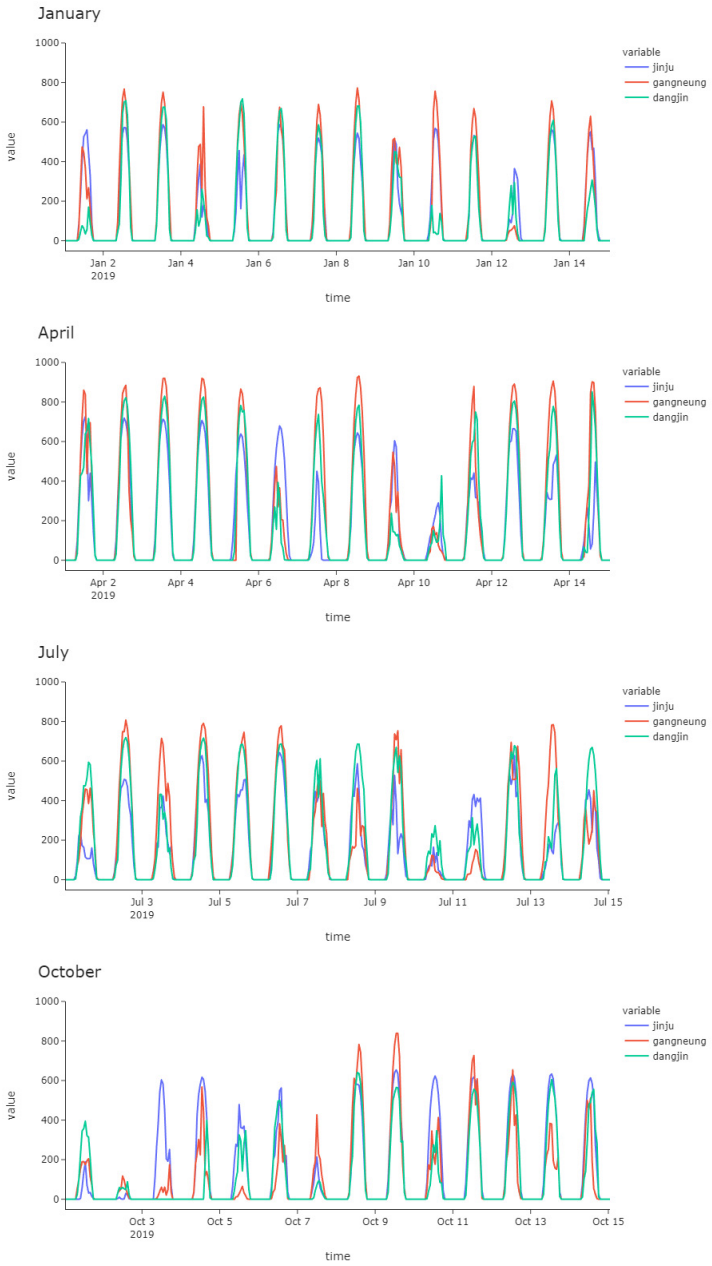


Figure 6: Energy plot

After analyzing the monthly power generation, we expected that there would be more power generation in July and August, when solar radiation was the highest, but

we could see that power generation was higher in April and May. The drop in land temperature caused by rainfall is thought to be one of the causes.

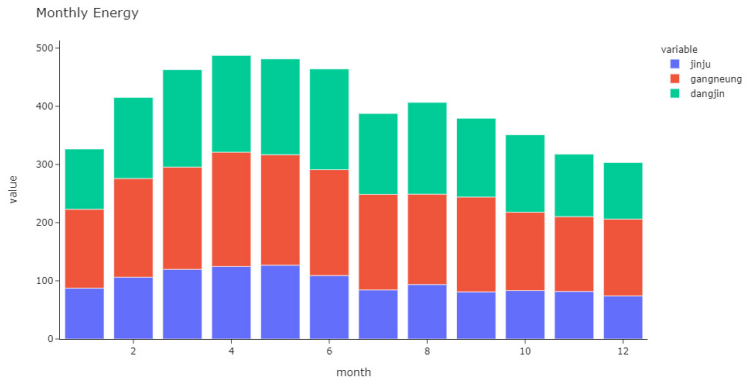


Figure 7: Monthly energy plot

Hourly power generation was highest between 1:00 and 2:00, and before 7:00 a.m. and after 8:00 p.m., power generation converged to zero.

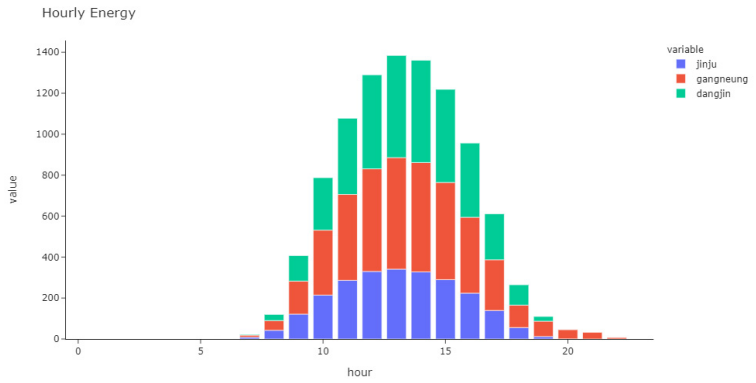


Figure 8: Hourly energy plot

We drew a box plot of solar power generation in each region. The mean and variance were different, but after normalizing to the maximum power generation(Jinju: 905MW, Gangneung: 1065MW, Dangjin: 1000MW), the mean and variance of each region were almost identical. Judging from this, we could see that the maximum power generation of each solar power plant is linearly proportional.

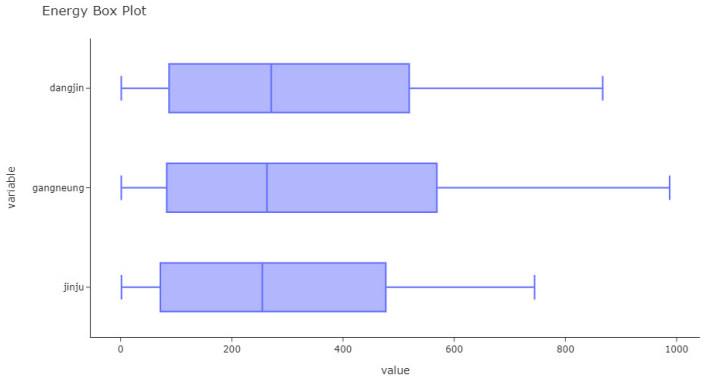


Figure 9: Energy box plot

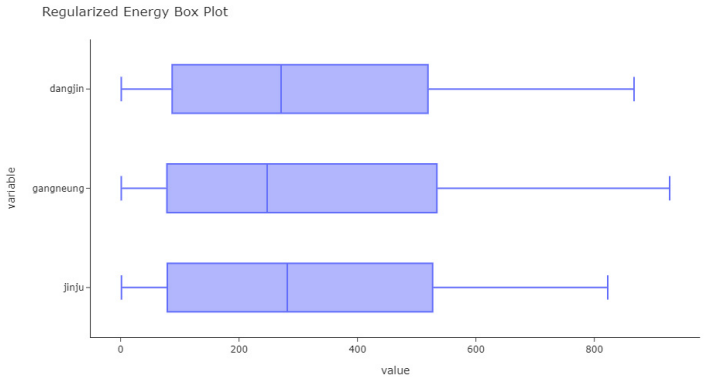


Figure 10: Regularized energy box plot

By default, pairplot shows us grid of Axes such that each numeric variable in data will by shared across the y-axes across a single row and the x-axes across a single column [2]. The diagonal plots are treated differently: a univariate distribution plot is drawn to show the marginal distribution of the data in each column. By analyzing pairplot, each generation has regional independence, so we could conclude that it's suitable for model training.

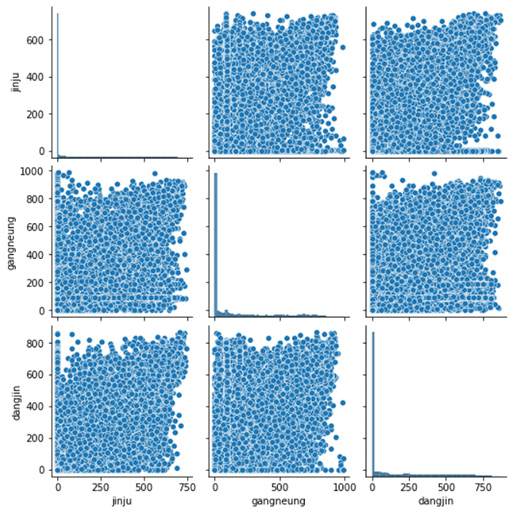


Figure 11: Energy pairplot

3.3 Economic Data

We could get each land's official price in National Spatial Data Infrastructure Portal. The official land price is based on the standard land price announced annually by the Minister of Land, Infrastructure and Transport, and calculates the price ratio according to the difference in land characteristics supplied by the Minister of Land, Infrastructure and Transport. We get the latest data (2021-07-27) consisting of 15 text file. The figure below is sample text file which has more than 900,000 rows.

```
pd.read_csv(economy_path_list[0], sep='|', engine='python', encoding='cp949')
```

	PNU	BASE_YEAR	STDMT	PNILP	PJJI_YN	PANN_YMD	ETC_CNTN	COL_ADM_SECT_CD
0	1111010600100070038	2021	1	7460000	1	20210531	NaN	11110
1	1111010600100070040	2021	1	2008000	0	20210531	NaN	11110
2	1111010600100070045	2021	1	4535000	0	20210531	NaN	11110
3	1111010600100080000	2021	1	4916000	0	20210531	NaN	11110
4	1111010600100090001	2021	1	4820000	0	20210531	NaN	11110
...	...	...	...	...	...	...	...	...
909146	1174010800104370032	2021	1	4375000	0	20210531	NaN	11740
909147	1174010800104370033	2021	1	4375000	0	20210531	NaN	11740
909148	1174010800104370034	2021	1	1364000	0	20210531	NaN	11740
909149	1174010800104370036	2021	1	5748000	0	20210531	NaN	11740
909150	1174010800104380008	2021	1	5244000	0	20210531	NaN	11740

909151 rows × 8 columns

Figure 12: Sampled official price text file

We could figure out that the first 4 digits of the PNU code and the first 4 digits of the legal dong code were the same.

```
legal_code = pd.read_csv('../data/legal_code.csv', dtype="0")
legal_code
```

	법정동코드	시도명	시군구명	읍면동명	동리명	생성일자	말소일자
0	1100000000	서울특별시	NaN	NaN	NaN	19880423	NaN
1	1111000000	서울특별시	종로구	NaN	NaN	19880423	NaN
2	1111010100	서울특별시	종로구	청운동	NaN	19880423	NaN
3	1111010200	서울특별시	종로구	신교동	NaN	19880423	NaN
4	1111010300	서울특별시	종로구	궁정동	NaN	19880423	NaN
...	...	...	...	...	...	...	...
20558	5013032022	제주특별자치도	서귀포시	표선면	하천리	20060701	NaN
20559	5013032023	제주특별자치도	서귀포시	표선면	성읍리	20060701	NaN
20560	5013032024	제주특별자치도	서귀포시	표선면	가시리	20060701	NaN
20561	5013032025	제주특별자치도	서귀포시	표선면	세화리	20060701	NaN
20562	5013032026	제주특별자치도	서귀포시	표선면	토산리	20060701	NaN

20563 rows × 7 columns

Figure 13: Legal code

So we mapped those two csv files. The result is below figure. We have same location name columns so later, we can map this csv files to the observation data we just created in Weather Observation Data.

	PNU	price	loc_name
0	1111	5789.0	Jongno
1	1117	6553.0	Yongsan
2	1120	4353.0	Seongdong
3	1121	4157.0	Gwangjin
4	1123	3587.0	Dongdaemun
...	...	...	...
256	4889	20.0	Gyeongsangnamdo
257	4889	20.0	Hapcheon
258	5011	223.0	Jeju
259	5013	137.0	Jeju
260	5013	137.0	Seogwipo

261 rows × 3 columns

Figure 14: Economy data

4. MODELING

We have weather observation data and daily solar power generation data for the region. In other words, both training data and target data exist. This means that solar power generation can be predicted through supervised machine learning, and it will be possible to find the optimal location for a solar power plant using this and other additional data (like economy and geographic data).

4.1 LightGBM

We used new GBDT algorithm with GOSS and EFB LightGBM [5]. This framework uses a leaf-wise tree growth algorithm, which is unlike many other tree-based algorithms that use depth-wise growth. Leaf-wise tree growth algorithms tend to converge faster than depth-wise ones. However, they tend to be more prone to overfitting. Lightgbm has many advantages:

- Faster training speed and higher efficiency
- Lower memory usage
- Better accuracy
- Support of parallel and GPU learning
- Capable of handling large-scale data

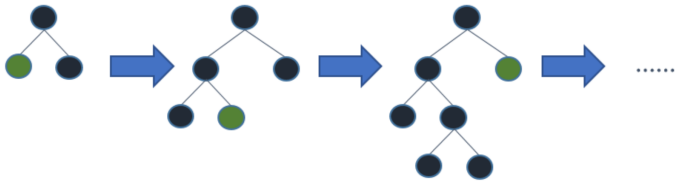


Figure 15: Leaf-wise tree growth

4.2 Hyperparameters Tuning

GridSearchCV is a function that comes in Scikit-learn's(or SK-learn) model selection package. So an important point here to note is that we need to have Scikit-learn library installed on the computer. This function helps to loop through predefined hyperparameters and fit your estimator (model) on your training set. So, in the end, we can select the best parameters from the listed hyperparameters [1].

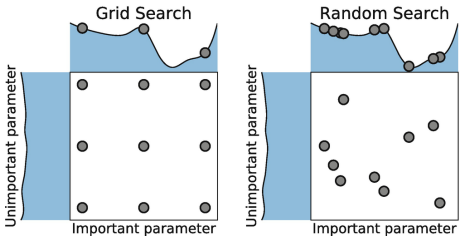


Figure 16: GridSearchCV

We know this is an old method, but it is more efficient than using the latest hyperparameter tuning technology like bayesian optimization because the amount of data was vast.

4.3 Local Training

For more efficient model learning, custom mattresses have been created to prevent actual solar power generation from being reflected in learning when it is less than 10% of maximum power generation. This is because overfitting can occur if you include very little power generation in your learning.

```
def nmae_10(y_pred, dataset):
    y_true = dataset.get_label()

    absolute_error = abs(y_true - y_pred)
    absolute_error /= 1000

    target_idx = np.where(y_true>=100)
    nmae = 100 * absolute_error[target_idx].mean()

    return 'score', nmae, False
```

Listing 4: Evaluate Function

```
def sola_nmae(answer, pred):
    absolute_error = np.abs(answer - pred)
    absolute_error /= 1000

    target_idx = np.where(answer>=100)
    nmae = 100 * absolute_error[target_idx].mean()

    return nmae
```

Listing 5: Score Function

To see how well the model trained during the training, we set the verbose-eval parameter to 500. The final loss value is distributed about 5-6 for each region, so we could see that the training was done well.

```
======================================== Gangneung Training Start ========================================
[LightGBM] [Info] Total Bins 1990
[LightGBM] [Info] Number of data points in the train set: 6759, number of used features: 17
[LightGBM] [Info] Start training from score 299.112368
Training until validation scores don't improve for 200 rounds
[500]   valid_0's score: 6.19167
[1000]  valid_0's score: 6.04034
[1500]  valid_0's score: 5.99863
[2000]  valid_0's score: 5.95411
[2500]  valid_0's score: 5.92756
Early stopping, best iteration is:
[2559]  valid_0's score: 5.9218
======================================== Gangneung Training Finish ========================================
```

Figure 17: Gangneung Training Process

```
========================================== Dangjin Training Start ==========================================
[LightGBM] [Info] Total Bins 2281
[LightGBM] [Info] Number of data points in the train set: 11621, number of used features: 18
[LightGBM] [Info] Start training from score 313.387058
Training until validation scores don't improve for 200 rounds
[500]   valid_0's score: 6.14567
[1000]  valid_0's score: 5.77038
[1500]  valid_0's score: 5.58683
[2000]  valid_0's score: 5.4754
[2500]  valid_0's score: 5.40042
[3000]  valid_0's score: 5.3485
[3500]  valid_0's score: 5.31184
[4000]  valid_0's score: 5.27776
[4500]  valid_0's score: 5.24603
[5000]  valid_0's score: 5.22724
[5500]  valid_0's score: 5.21488
[6000]  valid_0's score: 5.19617
Early stopping, best iteration is:
[6090]  valid_0's score: 5.19242
========================================== Dangjin Training Finish ==========================================
```

Figure 18: Dangjin Training Process

```
========================================== Jinju Training Start ==========================================
[LightGBM] [Info] Total Bins 1983
[LightGBM] [Info] Number of data points in the train set: 6801, number of used features: 17
[LightGBM] [Info] Start training from score 233.619984
Training until validation scores don't improve for 200 rounds
[500]   valid_0's score: 5.87133
[1000]  valid_0's score: 5.46206
[1500]  valid_0's score: 5.25439
[2000]  valid_0's score: 5.09687
[2500]  valid_0's score: 5.04748
Early stopping, best iteration is:
[2362]  valid_0's score: 5.04584
========================================== Jinju Training Finish ==========================================
```

Figure 19: Jinju Training Process

4.4 Feature Importance

We should analyze the importance of features to determine which variables have significantly affected the value of power generation. Normalized feature importance could be extracted by taking the average by obtaining a percentage value of feature importance for each region. Solar radiation features were selected as the most important features,

followed by temperature and articles. Due to the large number of missing precipitation data, the precipitation feature came out as a relatively unimportant feature. In any case, it affects solar radiation, temperature, and humidity when it rains, so it is okay to have a low importance of the feature of precipitation.

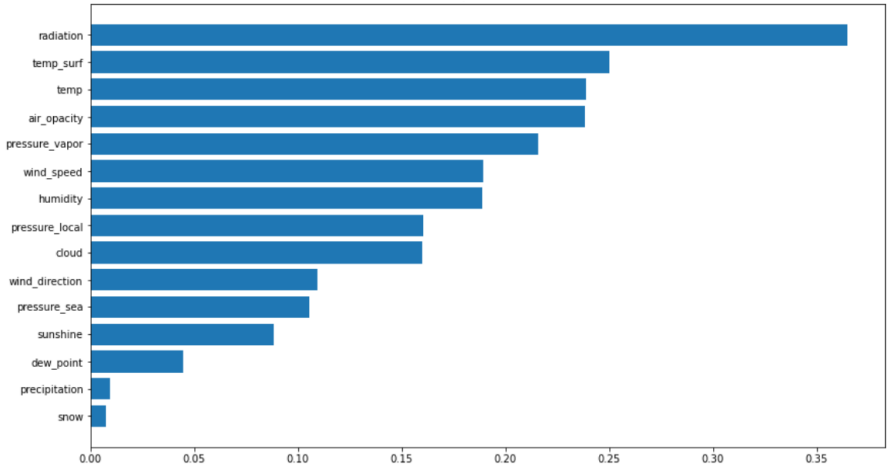


Figure 20: Feature importance plot

4.5 Global Prediction

Having conducted local training, we can derive solar power location scores from other regions using the trained models. We trained 3 different models by using Gangneung, Dangjin, Jinju observation data. If we input the unknown region's observation data to each model and sum up, we can get expected solar power generation data. Using the method, the optimal location score could be calculated by applying features from 40 different regions where solar radiation data exist to three models used in local training.

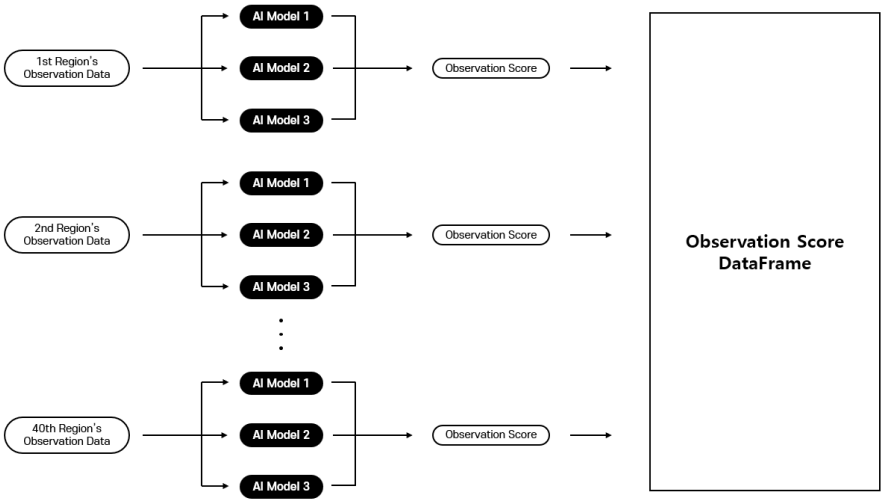


Figure 21: Global Prediction Scheme

5. RESULTS

From the previous model training, we could get the average model of the three AI model. Furthermore, we added the official land price to the data.

5.1 Visualization

Having conducted global prediction, we can derive solar power location scores from other regions using the trained models. The optimal location score could be calculated by applying features from 40 different regions where solar radiation data exist to three models used in local training. Furthermore, we added the official land price to the data. The first image is the dataframe and heatmap that represents observation score. The second image is the dataframe and heatmap that represents economic score. And the last image is the dataframe and heatmap that represents total score which equals to the observation score divided by cube root of the economy score.

location name	latitude	longitude	score
Gwangyangsi	34.943	127.691	509
Yeonggwanggun	35.284	126.478	503
Uiryeonggun	35.323	128.288	503
Daejeon	36.372	127.372	500
Andong	36.573	128.707	499
Hamyanggun	35.511	127.745	496
Jeonju	35.822	127.155	493
Gwangju	35.173	126.892	493
Chupungnyeong	36.220	127.995	493
Chuncheon	37.903	127.736	491
Hongseong	36.658	126.688	491
Daegu	35.885	128.619	490
Changwon	35.170	128.573	490
Wonju	37.338	127.947	487
Cheongsonggun	36.435	129.040	484
Cheongju	36.639	127.441	483
Daegwallyeong	37.677	128.718	482
Busan	35.105	129.032	479
Mokpo	34.817	126.382	471
Bukchuncheon	37.948	127.755	468
Bukgangneung	37.805	128.855	468
Gangjingun	34.645	126.784	467
Suwon	37.272	126.985	466
Incheon	37.478	126.625	466
Yeosu	34.739	127.741	461
Seoul	37.571	126.966	461
Gimhaesi	35.230	128.891	457
Gyeongjusi	35.817	129.201	452
Sunchanggun	35.371	127.129	451
Pohang	36.032	129.380	449
Jeju	33.514	126.530	446
Heuksando	34.687	125.451	445
Gochanggun	35.427	126.697	441
Bukchangwon	35.227	128.673	441
Gochang	35.348	126.599	436
Gosan	33.294	126.163	433
Boseonggun	34.763	127.212	428
Yangsansi	35.307	129.020	411
Baengnyeongdo	37.974	124.712	372
Ulleungdo	37.481	130.899	364
Cheorwon	38.148	127.304	323

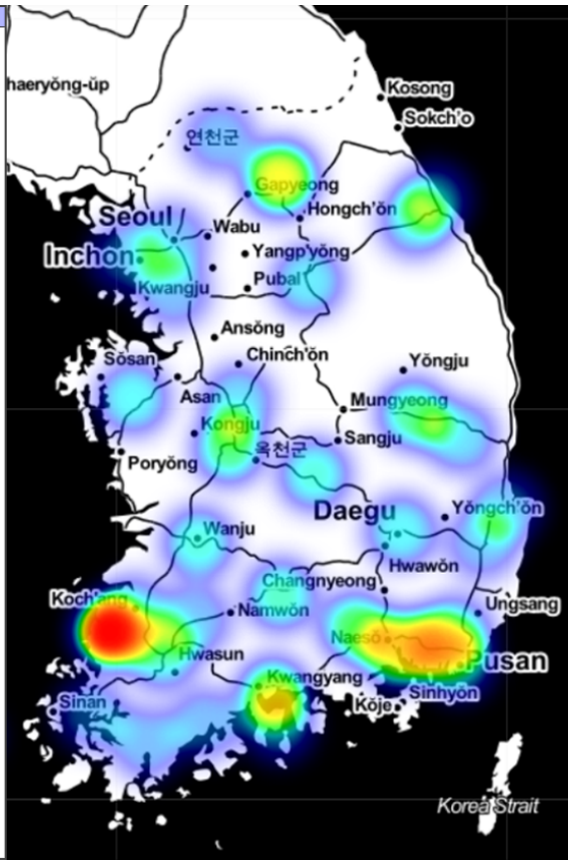


Figure 22: Observation score heatmap

Dataframe and heatmap that represents regional observation data. This dataframe includes 40 regions. Considering the amount of radiation and temperature, the observation scores would be high mainly in the southern region. However, the scores derived through machine learning showed that other variables played a significant role.

location name	latitude	longitude	price
Gwangyang	34.9434	127.6914	71.0
Yeonggwang	35.2837	126.4778	18.0
Uiryeong	35.3226	128.2881	22.0
Daejeon	36.372	127.3721	464.0
Andong	36.5729	128.7073	62.0
Hamyang	35.5114	127.7454	26.0
Jeonju	35.8215	127.155	368.0
Gwangju	35.1729	126.8916	215.0
Chuncheon	37.9026	127.7357	149.0
Hongseong	36.6576	126.6877	45.0
Changwon	35.1702	128.5728	320.0
Wonju	37.3375	127.9466	146.0
Cheongsong	36.4351	129.0401	21.0
Cheongju	36.6392	127.4407	163.0
Busan	35.1047	129.032	323.0
Mokpo	34.8173	126.3815	251.0
Gangjin	34.6446	126.7841	13.0
Suwon	37.2723	126.9853	1268.0
Incheon	37.4777	126.6249	58.0
Yeosu	34.7393	127.7406	116.0
Seoul	37.5714	126.9658	3934.0
Gimhae	35.2298	128.8908	239.0
Gyeongju	35.8174	129.2009	108.0
Sunchang	35.3713	127.1286	13.0
Pohang	36.032	129.38	168.0
Jeju	33.5141	126.5297	223.0
Gochang	35.4266	126.697	16.0
Boseong	34.7634	127.2123	14.0
Yangsan	35.3074	129.0201	245.0
Cheorwon	38.1479	127.3042	30.0

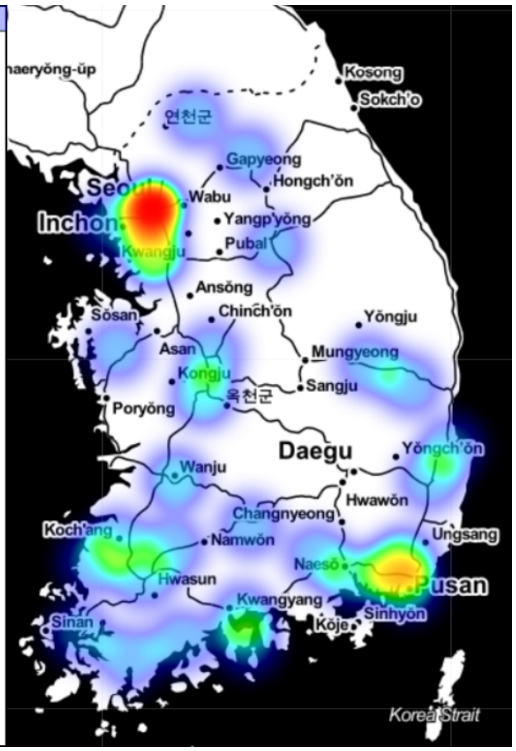


Figure 23: Economy score heatmap

Dataframe and heatmap that represents regional economic data. This dataframe includes 30 regions. The data is the average of the official land prices in each region. In the process of mapping with the previous Observation Score DataFrame, some regions that don't match were removed.

location name	latitude	longitude	score	price	final score
Gwangyang	34.9434	127.6914	509.0	2.9	175.35
Yeonggwang	35.2837	126.4778	503.0	2.06	244.2
Uiryeong	35.3226	128.2881	503.0	2.17	232.25
Daejeon	36.372	127.3721	500.0	4.64	107.73
Andong	36.5729	128.7073	499.0	2.81	177.83
Hamyang	35.5114	127.7454	496.0	2.26	219.65
Jeonju	35.8215	127.155	493.0	4.38	112.56
Gwangju	35.1729	126.8916	493.0	3.83	128.75
Chuncheon	37.9026	127.7357	491.0	3.49	140.54
Hongseong	36.6576	126.6877	491.0	2.59	189.57
Changwon	35.1702	128.5728	490.0	4.23	115.85
Wonju	37.3375	127.9466	487.0	3.48	140.1
Cheongsong	36.4351	129.0401	484.0	2.14	226.09
Cheongju	36.6392	127.4407	483.0	3.57	135.18
Busan	35.1047	129.032	479.0	4.24	112.99
Mokpo	34.8173	126.3815	471.0	3.98	118.33
Gangjin	34.6446	126.7841	467.0	1.9	245.94
Suwon	37.2723	126.9853	466.0	5.97	78.09
Incheon	37.4777	126.6249	466.0	2.76	168.86
Yeosu	34.7393	127.7406	461.0	3.28	140.47
Seoul	37.5714	126.9658	461.0	7.92	58.21
Gimhae	35.2298	128.8908	457.0	3.93	116.23
Gyeongju	35.8174	129.2009	452.0	3.22	140.21
Sunchang	35.3713	127.1286	451.0	1.9	237.51
Pohang	36.032	129.38	449.0	3.6	124.72
Jeju	33.5141	126.5297	446.0	3.86	115.41
Gochang	35.4266	126.697	441.0	2.0	220.5
Boseong	34.7634	127.2123	428.0	1.93	221.26
Yangsan	35.3074	129.0201	411.0	3.96	103.88
Cheorwon	38.1479	127.3042	323.0	2.34	138.01

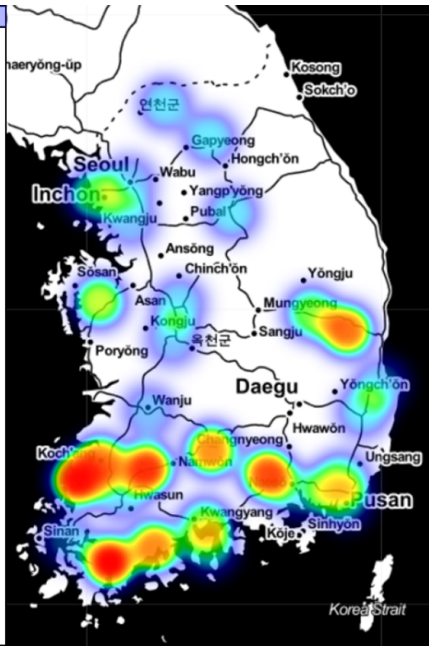


Figure 24: Best location heatmap

5.2 Optimal Location

As expected, regions with high official land prices had low final scores. And places in the southern region with low land prices had the highest final score. Compared to the actual normalized solar power generation, the difference is not so much.

5.3 Evaluation

Comparing with the actual solar power generation energy in several regions, it was found that there was no significant difference. In addition, considering economic factors, we were able to select the optimal areas for solar power plants more accurately than other previous papers.

6. FUTURE WORK

In this paper, we made AI model using just three local areas (gangneung, jinju, dangjin) because it's difficult to get solar power generation data (target data). In the future, we'll increase the local area as much as possible. By doing that, we can get a more accurate AI model and a more accurate observation score. The the way we get the global data in Korea, we could predict other countries like USA, England, China... We could suggest the optimal location for solar power plants for them just by getting observation data of their regions.

References

[1] Lars Buitinck et al. “API design for machine learning software: experiences from the scikit-learn project”. In: *arXiv preprint arXiv:1309.0238* (2013).

[2] Nicholas Cox. “PAIRPLOT: Stata module for plots of paired observations”. In: (2007).

[3] Md Ziaul Hassan et al. “Forecasting day-ahead solar radiation using machine learning approach”. In: *2017 4th Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE)*. IEEE. 2017, pp. 252–258.

[4] Ju-Hee Jang et al. “A preliminary research of the bifacial PV system under installation conditions”. In: *Journal of the Korean Solar Energy Society* 38.6 (2018), pp. 51–63.

[5] Guolin Ke et al. “Lightgbm: A highly efficient gradient boosting decision tree”. In: *Advances in neural information processing systems* 30 (2017), pp. 3146–3154.

[6] Chang Ki Kim et al. “Derivation of typical meteorological year of Daejeon from satellite-based solar irradiance”. In: *Journal of the Korean Solar Energy Society* 38.6 (2018), pp. 27–36.

[7] YH Kwon, JY Kim, and MJ Lee. “Environmental considerations in the siting of solar and wind power plants”. In: *Korea Environment Institute* (2008).

[8] J Lee, HM Chung, and SS Lee. “Analysis on the location of the sunray energy power plants”. In: *Korea Knowledge Information Technology Society* 3.3 (2008), pp. 31–37.

[9] Kirim Lee and L. Hee. “Solar Power Plant Location Analysis Using GIS and Analytic Hierarchy Process”. In: 2015.

[10] Sung-Hun Lee et al. “Economic evaluation method for photovoltaic system development using insolation data analysis”. In: *Journal of the Korean Institute of Illuminating and Electrical Installation Engineers* 25.10 (2011), pp. 38–46.

[11] JI Park, MH Park, and SY Choi. “A study on GIS based suitability analysis of solar photovoltaic power generation using correlation analysis”. In: *The Korean Society of Cadastre* 28 (2012), pp. 91–107.

[12] Wim CM Van Beers and Jack PC Kleijnen. “Kriging interpolation in simulation: a survey”. In: *Proceedings of the 2004 Winter Simulation Conference, 2004.* Vol. 1. IEEE. 2004.

[13] Sung-Wook Yun et al. “A Study for Planning Optimal Location of Solar Photovoltaic Facilities using GIS”. In: (2019).