Performance Assessment of Hetero-Junction Intrinsic Thin Film Photovoltaic Module Using Machine Learning Methods

Submitted to University of Mumbai in partial fulfillment

of the requirements of the degree of

Bachelor of Engineering

in

Electronics Engineering

by

Mr. Aditya Pandey	**Roll No. 18EE1016**
Mr. Prasad Kulkarni	**Roll No. 18EE1057**
Mr. Tushar Patil	**Roll No. 18EE1162**

Under the Guidance of

Dr. Vishwesh Vyawahare

Dr. Dhiraj Magare

Department of Electronics Engineering,

Ramrao Adik Institute of Technology, Navi Mumbai

April - 2022

Ramrao Adik Institute of Technology

CERTIFICATE

This is to certify that, the Project-II entitled

"Performance Assessment of Hetero-Junction Intrinsic Thin Film Photovoltaic Module Using Machine Learning Methods"

is a bonafide work done by

Mr. Aditya Pandey

Mr. Prasad Kulkarni

Mr. Tushar Patil

and is submitted in the partial fulfillment of the requirements for the award of degree of

Bachelor of Engineering
in
Electronics Engineering
to the

University of Mumbai

_____

Guide

Dr. Vishwesh Vyawahare

Project Coordinator	Head of Department	Principal
(Dr. Sharmila Petkar)	**(Dr. Vishwesh Vyawahare)**	**(Dr. Mukesh D. Patil)**

Declaration

We declare that this written submission represents our ideas in our own words and where other's ideas or words have been included. We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

.....

(Mr. Aditya D. Pandey, Roll No. 18EE1016)

.....

(Mr. Prasad P. Kulkarni, Roll No. 18EE1057)

.....

(Mr. Tushar S. Patil, Roll No. 18EE1162)

Date :

Project-II Report Approval for B. E.

This Project-II report entitled ***" Performance Assessment of Hetero-Junction Intrinsic Thin Film Photovoltaic Module Using Machine Learning Methods "*** by ***Mr. Aditya D. Pandey*** , ***Mr. Prasad P. Kulkarni*** and ***Mr. Tushar S. Patil*** is approved for the degree of ***Bachelor's Degree in Electronics Engineering, University of Mumbai***.

Internal Examiner (Guide) :

. . . . . .

External Examiner :

. . . . . .

Date :

Place :

Acknowledgments

With great pleasure, we avail this opportunity to express our profound gratitude and deep regards to our project guide **Dr. Vishwesh Vyawahare** and co-guide **Dr. Dhiraj Magare** for their spirited guidance, monitoring and constant encouragement throughout the completion of this seminar report. We have deep sense of admiration for their innate goodness and inexhaustible enthusiasm. It helped us to work in right direction to attain desired objective.

We are also thankful to **Dr. Sharmila Petkar**, Project Co-ordinator and **Dr. Vishwesh Vyawahare**, Head of Department of Electronics Engineering, RAIT, Nerul for their generous support, devoting their valuable time and helped us in all possible ways towards successful completion of this work. We thank all those who have contributed directly or indirectly to this work.

We take this privilege to express our sincere thanks to **Dr. Mukesh D. Patil**, Principal, RAIT for their support, encouragement and providing the much necessary facilities. We extend thanks to our friends who have supported in every stage of these report. We cannot end without thanking our lovely family for their encouragement.

Date

Signature

Abstract

A solar cell built of ultra-thin amorphous silicon and high-quality mono-crystalline silicon is known as a hetero-junction intrinsic thin film. It has a pyramid surface on the front that increases sunlight absorption. The operating environment has a significant impact on the performance of hetero-junction intrinsic thin-film photovoltaic modules with real I–V (current-voltage) characteristics. Changes in the environment have a significant impact on solar irradiation. Clouds also have a significant impact on the solar irradiation that a PV cell receives. In this project, we will use the Random Forest Regression machine learning algorithm to investigate the effects of sudden changes in environmental conditions on power output and module temperature of an HIT (Heterojunction with Intrinsic Thin Layer) module, where irradiance, temperature, and module efficiency parameters are taken into account when designing modules. The algorithm's output will be studied to gain a better understanding of performance variations as well as the behavior of the power output and module temperature when subjected to random influences induced by various environmental variables. The suggested algorithm is not restricted to a certain module technology or geographic location.

Contents

List of Figures

Chapter 1

Introduction

PV cells, also referred to as solar cells, are electrical components that create electricity when they are exposed to photons, or light particles. The French physicist Edmond Becquerel gave the name to the photovoltaic phenomenon, which was discovered in 1839. PV cell modules started appearing on roofs towards the end of the 1980s. As the photovoltaic industry has flourished in the 21st century, the construction of vast solar farms has continually increased in photovoltaic capacity. A photovoltaic cell is made up of several layers of materials, each with its own function. The specifically treated semiconductor layer is the most crucial layer in a solar cell. It has two separate layers (p-type and n-type) and is responsible for converting the Sun's energy into usable power via the photovoltaic effect. A layer of conducting material is present on both sides of the semiconductor, which "collects" the electricity generated. Note that the backside of the cell, which is shaded, can afford to have the conductor totally covered, whereas the front, which is illuminated, must use the conductors sparingly to prevent blocking too much of the Sun's energy from reaching the semiconductor. The anti-reflection coating is the final layer, which is only applied to the illuminated side of the cell. Reflection loss can be severe because all semiconductors are naturally reflective.

To limit the amount of solar radiation reflected off the cell's surface, one or more layers of an anti-reflection coating can be applied. Solar cells can be grouped into arrays. Homeowners have installed solar cells in considerably smaller designs on their rooftops, referred to as solar cell panels or just solar panels, to replace or supplement their normal energy source. Solar photovoltaic (PV) module temperature impacts the efficiency of PV modules, which increases the overall power output of the PV system. PV modules lose output power when they heat up, and degradation accelerates as a result. There is a minor increase in short circuit current as the module temperature rises; nevertheless, the open-circuit voltage of the module drops dramatically, lowering the module's output power.

The wind is one of the most critical environmental elements that affects the temperature of the module, and whose effect is finally translated into the PV module's performance. The PV module can be more efficient when the temperature is lower. As well as wind speed, the wind direction relative to the PV module's orientation determines what degree of temperature reduction the air circulation around the PV module causes. These models were developed considering how module temperature and several meteorological characteristics, including wind speed, wind direction, and in-plane irradiance, interrelate. Although MPPTs (maximum power point trackers) or module-level power electronics (MLPE) such as microinverters or DC-DC optimizers can be connected to create arrays with the desired peak DC voltage and current capacity, this can be done with or without the use of independent MPPTs. In arrays with series/parallel coupled cells, shunt diodes help reduce shadowing power loss.

Chapter 2

Literature Survey

Today, photovoltaic modules are commonly used. These modules are available in various shapes and sizes. Hetero-Junction Intrinsic Thin Film Module is one of them. They are the most modern and efficient photovoltaic module currently available. They are, nevertheless, still influenced by the conditions in which they labor. Wind speed, sunlight, weather, and other environmental conditions all have an impact on the module's performance. The temperature of the module is strongly affected by wind movement around it, which has an impact on the module's output power. Dr Dhiraj Magare et al. used the linear regression approach of machine learning to estimate module temperature and maximum power production in prior works such as "Wind Effect Modeling and Analysis for Estimation of Photovoltaic Module Temperature." However, because machine learning has advanced so much in recent years, researchers are continuously coming up with new ways to improve prediction accuracy. They are also always looking for methods to improve the module's efficiency in delivering power.

Chapter 3

Theory

3.1 Random Forest Regression

A random forest regression method is a regression technique utilizing ensemble learning as the supervised learning approach. A machine learning algorithm combines several different forecasts from different machine learning models to make more accurate predictions. The structure of a Random Forest is depicted in figure 3.1. The trees run in a straight line

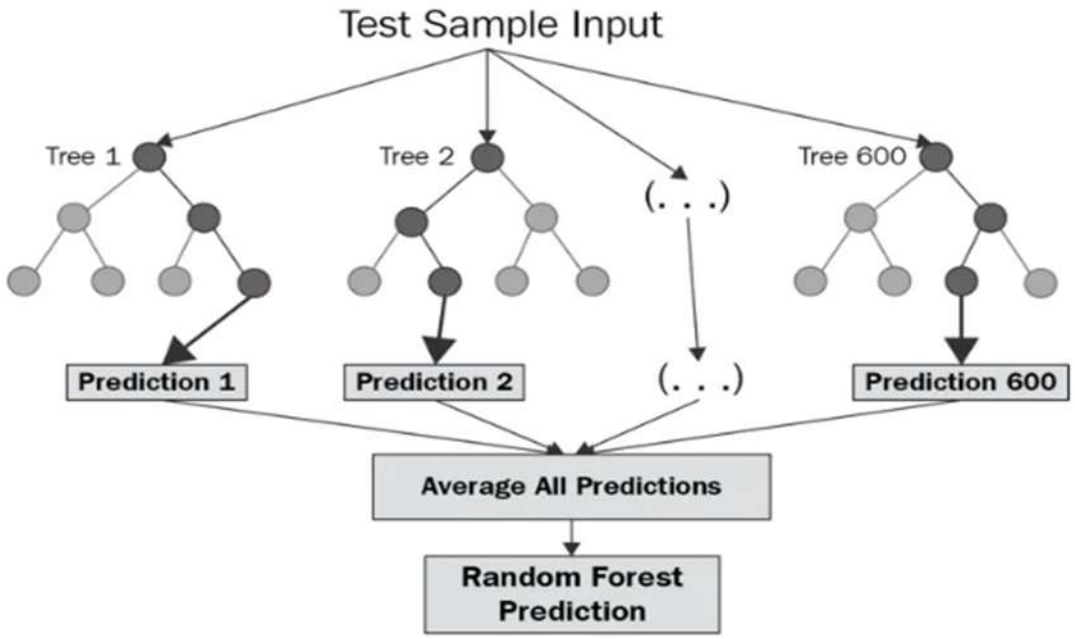

Figure 3.1: Random Forest Regression

with no contact. When a Random Forest is trained, many decision trees are constructed, and the mean of all the trees' predictions is derived. Let's go over the steps to acquire a

better knowledge of the Random Forest algorithm:

- The training set is divided into n random data points.
- Analyze the k data points to create a decision tree.
- Steps 1 and 2 should be repeated for as many trees as N is required.
- Assign the new data point to the average of all predicted y values from all your N-tree trees for each new data point.

In terms of accuracy and power, Random Forest Regression stands out. It works well on a wide range of situations, including those with non-linear relationships. Overfitting is a possibility, there is no interpretability, and we need to select which trees to include in the model. Because it exploits randomization on two levels, the ensemble of decision trees has a high level of accuracy.

- At each split, the algorithm selects a subset of features at random to be used as candidates. This avoids many decision trees from sharing the same set of features, which decorrelate individual trees.
- When generating splits, each tree takes a random sample of data from the training dataset. This adds a layer of unpredictability to the equation, preventing the trees from overfitting the data. They can't overfit the data since they can't see all of it.

3.2 Working of Photovoltaic Module

A photovoltaic cell as shown in figure 3.2 is composed of semiconductor materials that capture photons from the sun and generate an electrical charge. Solar radiation travels through the geosphere in the form of photons, elementary particles that travel at speeds of 300,000 kilometers per second. When photons impact a semiconductor material such as silicon, the electrons in its atoms are released, leaving a free space. The wayward electrons roam around aimlessly in search of a new "hole" to fill. Electrons must, however, move in the same direction to generate an electric current. Silicon comes in two forms to do this. It has one more electron than silicon, so the side facing outward is doped with phosphorus atoms, whereas the side facing inward is doped with boron atoms, which have one fewer electron than silicon. With the sandwich constructed, the excess electron layer serves as the negative terminal (n), while the shortfall on the positive side serves as the positive terminal (p). At the connection between the two layers, an electric field is formed.

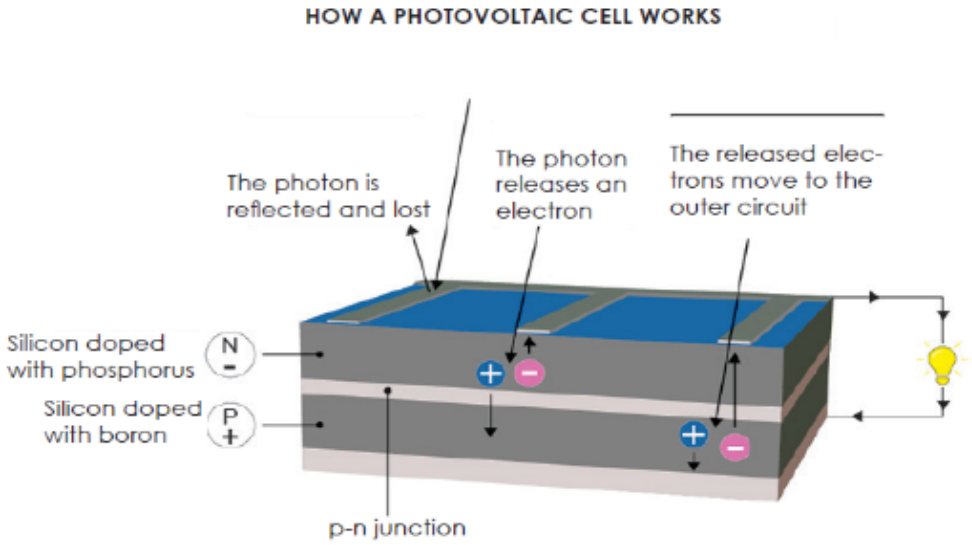

Figure 3.2: Working of PV Cell

An electric field sweeps electrons to the n-side, while holes drift to the p-side, when they are stimulated by photons. Electrical energy is transferred to the external circuit in the form of electrons and holes, which are directed to the electrical contacts on both sides. This results in a direct current. The top of the cell is coated with an anti-reflective material to reduce photon loss due to surface reflection.

3.3 Photovoltaic Cell Efficiency

Because the visible range of electromagnetic radiation contains most of the energy in sunshine and artificial light, a solar cell absorber should be effective at absorbing radiation at those wavelengths. Solar cells are rated according to their energy efficiency based on how much electricity they produce compared to the amount of light they receive. In order to test the efficiency of the cell arrays, the cells are wired together into modules, which are then assembled into arrays. A solar simulator simulates optimal sunlight conditions including 1,000 watts (W) per cubic meter at 25 degrees Celsius with the panels positioned in front of it. Peak power, or the amount of electricity generated by the system, is a proportion of the solar energy received. The efficiency of a panel of one square meter that provides 200 W of electrical power is 20 percent. PV cells have a theoretical maximum efficiency of roughly 33 percent. The Shockley-Queisser limit is the name for this limit. A solar cell's output, or how much electricity it can produce, is determined by several

factors, including the solar radiation levels in the area, the efficiency of the cell, and the type of installation. In the Paris area, incident solar radiation is 1 megawatthour per square meter per year (MWh/sq.m/y), compared to 1.70 MWh/sq.m/y in southern France and approximately 3.0 MWh/sq.m/y in the Sahara Desert. In Paris, a solar panel with a 15 percent efficiency rating will create 150.0 kWh/sq.m/y, but in the Sahara, it will generate 450.0 kWh/sq.m/y.

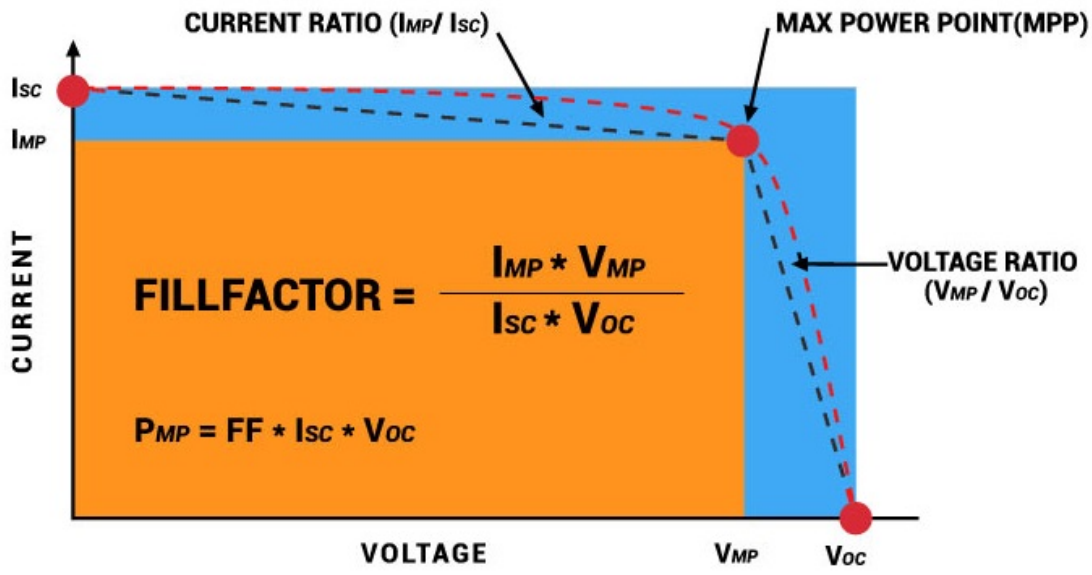

Figure 3.3: Efficiency of Solar Cell

3.4 Structure of HIT

Heterojunction with Intrinsic Thin-layer, or HIT, is an abbreviation for intrinsic thin-film heterojunction. Because Sanyo Corporation of Japan has sought for a registered trademark for HIT, it is also known as HJT or SHJ (Silicon Heterojunction solar cell). Three layers of photovoltaic material make up heterojunction solar panels as shown in figure 3.3. HJT cells combine crystalline and amorphous ”thin-film” silicon technologies into a single device. The top layer of amorphous silicon captures sunshine and light that reflects off the lower layers before it reaches the crystalline layer. The middle layer, monocrystalline silicon, is responsible for converting most of the sunlight into energy. Finally, there is an amorphous thin-film silicon layer behind the crystalline silicon. The remaining photons that pass through the first two levels are captured in this final layer. When these technologies are used together, more energy may be gathered than if they were used separately, with efficiency of 25 percent or higher. The basic structure of an HIT solar cell is shown

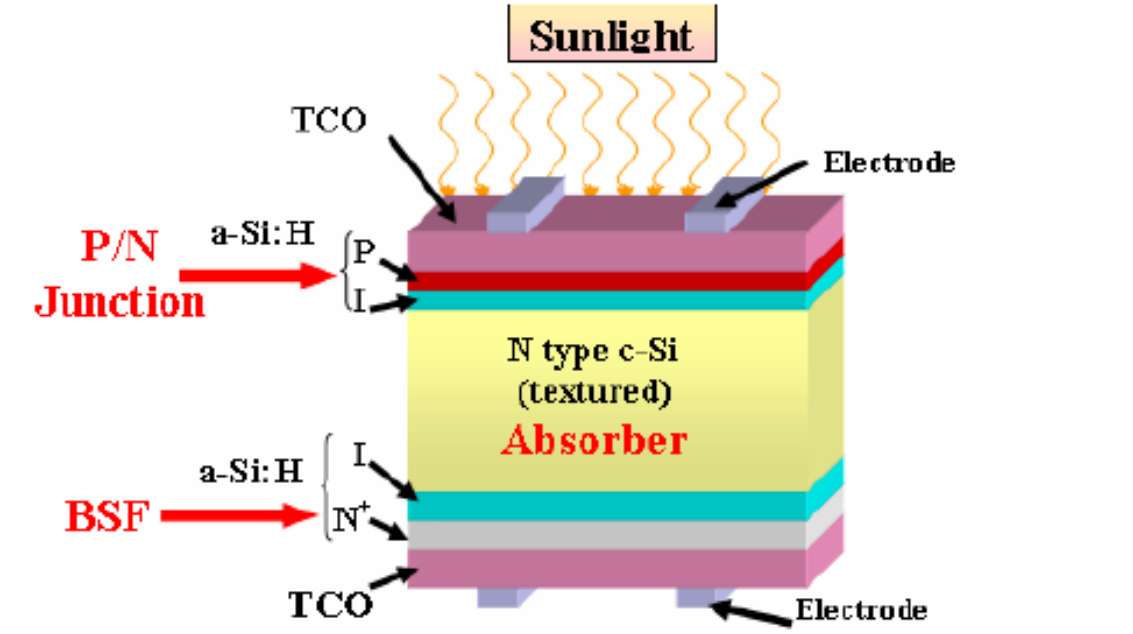

Figure 3.4: Structure of HIT PV Cell

in the diagram below, which is characterized by a pi-type a-Si: H film (film thickness 5-10 nm) on the light irradiation side and an n-type a-Si: H film (film thickness 5- l0nm) sandwiching a crystalline silicon wafer, forming transparent electrodes and collector electrodes on both sides to form an asymmetrical HIT solar cell.

HIT combines the greatest features of crystalline silicon with those of amorphous silicon thin film to create a high-power hybrid cell that outperforms the industry's standard, PERC. There is the potential for significant cost reductions because the HIT manufacturing process requires four less steps than PERC technology. While PERC has long been a popular choice in the industry, its complicated manufacturing process can't compete with HIT. Furthermore, PERC lacks HIT's high-temperature performance advantage.

3.5 Fabrication of HIT

The fabrication order differs from one group to the next. The absorber layer of HIT cells is usually made of high-quality CZ/FZ grown c-Si wafers. The surface of the wafer is textured with alkaline etchants such as NaOH or (CH3)4NOH to generate pyramids of 5-10m height. After that, peroxide and HF solutions are used to clean the wafer. It is followed by the deposition of an intrinsic alkali Si passivation layer, which is usually done by PECVD or Hot-wire CVD. For deposition, silane gas (SiH4) is diluted with H2. The deposition temperature and pressure are 300°C and 0.1-1 Torr, respectively.

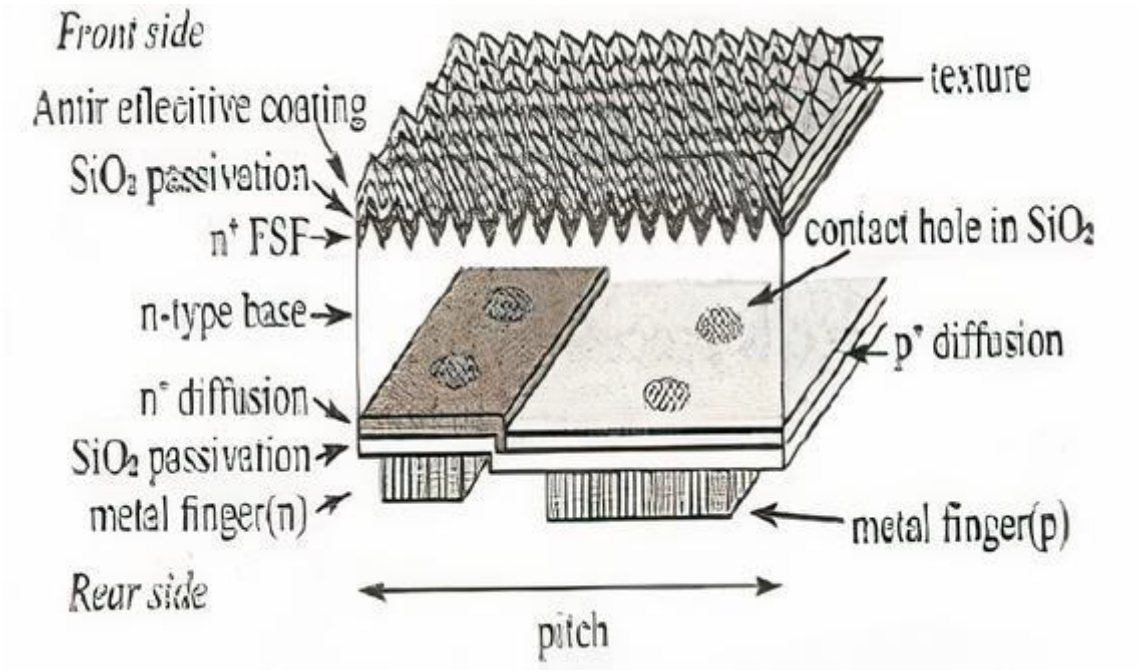

Figure 3.5: Fabrication of PV Cell

To avoid the production of faulty epitaxial Si, this stage must be precisely controlled. Deposition and annealing cycles, as well as H2 plasma treatment, were proven to offer excellent surface passivation. A-Si p-type layers are deposited by mixing phosphene gas with silane, whereas a-Si n-type layers are deposited by mixing diborane with silane. On a c-Si wafer deposited directly with doped a-Si layers, very poor passivation properties are observed. Dopant-induced defect formation in a-Si layers is most likely to blame. Because a-Si has a high lateral resistance, Bi-facial designs typically have front and back layers of Indium Tin Oxide (ITO) respectively that are transparent conductive oxides (TCOs). It's usually placed on the backside of a fully metallized cell to prevent back metal diffusion and to match the impedance of reflected light. For bi-facial design, a 50-100m thick silver/aluminum grid is placed using stencil printing for the front and back contacts.

Chapter 4

Design and Implementation Details

4.1 Design Methodology

This project proposes a prediction model using Random Forest Regression machine learning algorithm that can predict module temperature and maximum power output of a given photovoltaic module under different environmental conditions. A data log consisting of 18000+ data points of a whole year was used to train the model. Out of these, 80 percent data points have been used for training the model and 20 percent for testing.

The data set consists of parameters such as short circuit current (Isc), open circuit voltage (Voc), current at maximum power (Ipmax), voltage at maximum power (Vpmax), maximum power (Pmax), module temperature (Temp_mid_avg), solar irradiance (Gt), wind speed (WS) and ambient temperature (Tamb). The parameters have been measured at an interval of 10 minutes. Out of these parameters, Temp_mid_avg and Pmax are dependent parameters. Gt, WS and Tamb are independent parameters for predicting temp_mid_avg. Isc and Voc are independent parameters for predicting Pmax. Temperature coefficient of maximum power for HIT module is -0.33 percent/°C Linear regression machine learning algorithm has been used before, but it is a very basic algorithm and limited to only linear relationships. It also doesn't work efficiently with large data types. The PV module's maximum output power was measured and compared to the predicted maximum output power. Random Forest Regression was used to get the anticipated output power. Random forest uses ensemble technique which means that it combines multiple models.

4.1.1 Required Libraries

Various libraries have been used for achieving the prediction model. Pandas is used for data processing and manipulation. Sklearn is used for training and testing of the data.

Matplotlib and seaborn are used for data visualization.

4.1.2 Data Visualisation

Graphs have been plotted to compare the various parameters. Using this comparison, correlations between the parameters were observed.

4.1.3 Training and testing

Random Forest regression model operates by making several decision trees at the time of training and the final output is given by the mean value of all the decision trees. It picks random data points from the data set and then the decision tree is made according to the picked data set. The number of trees are then chosen and the same process of picking random data points and making decision trees is continued.

4.2 Code Snippets

4.2.1 Yearly Prediction for 2010

```
x_pmax = df[['Isc', 'Voc']]
y_pmax = df[['Pmax']]
```

```
x_mod = df[['Gt', 'WS', 'Tamb']]
y_mod = df[['Temp_Mid_avg']]
```

Figure 4.1: Setting independent and dependent variables

```
X_train, X_test, y_train, y_test = train_test_split(
    x_mod, y_mod, test_size=0.2)
regressor = RandomForestRegressor()
regressor.fit(X_train, y_train.values.ravel())

y_pred = regressor.predict(X_test)
daf=pd.DataFrame({'Actual':y_test.values.flatten(), 'Predicted':y_pred,'Error':abs(y_pred-y_test.values.flatten())})
daf
```

Figure 4.2: Training data for predicting Module Temperature

```
X_train_pmax, X_test_pmax, y_train_pmax, y_test_pmax = train_test_split(
    x_pmax, y_pmax, test_size=0.2)
regressor_pmax = LinearRegression()
regressor_pmax.fit(X_train_pmax, y_train_pmax.values.ravel())

y_pred_pmax = regressor_pmax.predict(X_test_pmax)
daf_pmax=pd.DataFrame({'Actual':y_test_pmax.values.flatten(), 'Predicted':y_pred_pmax,'Error':abs(y_pred_pmax-y_test_pmax.values.flatten())})
daf_pmax
```

Figure 4.3: Training data for predicting Maximum Power

```
sns.set(rc={'figure.figsize':(25,10)})


ax = sns.distplot(y_test, hist=False, kde_kws = {'color':'#0000ff',
                    'linewidth':4, 'linestyle':'-', 'alpha':0.9}, label="Actual Value")
sns.distplot(y_pred, hist=False, kde_kws = {'color':'#DC143C',
                    'linewidth':4, 'linestyle':'--', 'alpha':0.9}, label="Predicted Values" , ax=ax)
plt.xlabel('Actual Values')
plt.ylabel('Predicted Values')


plt.title('Actual vs Predicted values')


plt.show()
```

Figure 4.4: Code for plotting Actual vs Predicted graph of Module Temperature

```
sns.set(rc={'figure.figsize':(25,10)})


ax = sns.distplot(y_test_pmax, hist=False, kde_kws = {'color':'#0000ff',
                    'linewidth':4, 'linestyle':'-', 'alpha':0.9}, label="Actual Value")
sns.distplot(y_pred_pmax, hist=False, kde_kws = {'color':'#DC143C',
                    'linewidth':4, 'linestyle':'--', 'alpha':0.9}, label="Predicted Values" , ax=ax)


plt.title('Actual vs Predicted values')


plt.show()
```

Figure 4.5: Code for plotting Actual vs Predicted graph of Maximum Power

4.2.2 Season-Wise Prediction for 2010

```
df = pd.read_csv(r"C:\Users\adity\Documents\Final Year Project\filteredData_train.csv")
df1 =df[(df["ModuleMonth"] >= 2) & (df["ModuleMonth"] < 6)]
df2 =df[(df["ModuleMonth"] >= 6) & (df["ModuleMonth"] < 10)]
df3 =df[(df["ModuleMonth"] >= 10) & (df["ModuleMonth"]) < 13 & (df["ModuleMonth"] == 1)]
df1
```

Figure 4.6: Code for splitting data into seasons

```
X_train_season1, X_test_season1, y_train_season1, y_test_season1 = train_test_split(
    x_mod_season1, y_mod_season1, test_size=0.2)
regressor = RandomForestRegressor(n_estimators=100, random_state=1)
regressor.fit(X_train_season1, y_train_season1.values.ravel())

y_pred_season1 = regressor.predict(X_test_season1)
daf_season1=pd.DataFrame({'Actual':y_test_season1.values.flatten(), 'Predicted':y_pred_season1,'Error':abs(y_pred_season1-y_test_season1.values.flatten())})
daf_season1
print("---------------------------------------")
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test_season1, y_pred_season1)))
```

Figure 4.7: Training data for predicting Module Temperature for Summer Season

```
X_train_season2, X_test_season2, y_train_season2, y_test_season2 = train_test_split(
    x_mod_season2, y_mod_season2, test_size=0.2)
regressor2 = RandomForestRegressor(n_estimators=100, random_state=1)
regressor2.fit(X_train_season2, y_train_season2.values.ravel())

y_pred_season2 = regressor2.predict(X_test_season2)
daf_season2=pd.DataFrame({'Actual':y_test_season2.values.flatten(), 'Predicted':y_pred_season2,'Error':abs(y_pred_season2-y_test_season2.values.flatten())})
daf_season2
print("---------------------------------------")
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test_season2, y_pred_season2)))
```

Figure 4.8: Training data for predicting Module Temperature for Rainy Season

```
X_train_season3, X_test_season3, y_train_season3, y_test_season3 = train_test_split(
    x_mod_season3, y_mod_season3, test_size=0.2)
regressor3 = RandomForestRegressor(n_estimators=100, random_state=1)
regressor3.fit(X_train_season3, y_train_season3.values.ravel())

y_pred_season3 = regressor3.predict(X_test_season3)
daf_season3=pd.DataFrame({'Actual':y_test_season3.values.flatten(), 'Predicted':y_pred_season3,'Error':abs(y_pred_season3-y_test_season3.values.flatten())})
daf_season3
print("---------------------------------------")
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test_season3, y_pred_season3)))
```

Figure 4.9: Training data for predicting Module Temperature for Winter Season

```
X_train_pmax_season1, X_test_pmax_season1, y_train_pmax_season1, y_test_pmax_season1 = train_test_split(
    x_pmax_season1, y_pmax_season1, test_size=0.2)
regressor_pmax_season1 = RandomForestRegressor(n_estimators=100, random_state=42)
regressor_pmax_season1.fit(X_train_pmax_season1, y_train_pmax_season1.values.ravel())

y_pred_pmax_season1 = regressor_pmax_season1.predict(X_test_pmax_season1)
daf_pmax_season1=pd.DataFrame({'Actual':y_test_pmax_season1.values.flatten(), 'Predicted':y_pred_pmax_season1,'Error':abs(y_pred_pmax_season1-y_test_pmax_season1.values.flatten())})

print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test_pmax_season1, y_pred_pmax_season1)))
```

Figure 4.10: Training data for predicting Maximum Power for Summer Season

```
X_train_pmax_season2, X_test_pmax_season2, y_train_pmax_season2, y_test_pmax_season2 = train_test_split(
    x_pmax_season2, y_pmax_season2, test_size=0.2)
regressor_pmax_season2 = RandomForestRegressor(n_estimators=100, random_state=42)
regressor_pmax_season2.fit(X_train_pmax_season2, y_train_pmax_season2.values.ravel())

y_pred_pmax_season2 = regressor_pmax_season2.predict(X_test_pmax_season2)
daf_pmax_season2=pd.DataFrame({'Actual':y_test_pmax_season2.values.flatten(), 'Predicted':y_pred_pmax_season2,'Error':abs(y_pred_pmax_season2-y_test_pmax_season2.values.flatten())})

print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test_pmax_season2, y_pred_pmax_season2)))
```

Figure 4.11: Training data for predicting Maximum Power for Rainy Season

```
X_train_pmax_season3, X_test_pmax_season3, y_train_pmax_season3, y_test_pmax_season3 = train_test_split(
    x_pmax_season3, y_pmax_season3, test_size=0.2)
regressor_pmax_season3 = RandomForestRegressor(n_estimators=100, random_state=42)
regressor_pmax_season3.fit(X_train_pmax_season3, y_train_pmax_season3.values.ravel())

y_pred_pmax_season3 = regressor_pmax_season3.predict(X_test_pmax_season3)
daf_pmax_season3=pd.DataFrame({'Actual':y_test_pmax_season3.values.flatten(), 'Predicted':y_pred_pmax_season3,'Error':abs(y_pred_pmax_season3-y_test_pmax_season3.values.flatten())})

print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test_pmax_season3, y_pred_pmax_season3)))
```

Figure 4.12: Training data for predicting Maximum Power for Winter Season

Chapter 5

Results and Observations

5.1 Yearly Results

In order to predict the maximum power and temperature of the module, a prediction model was used on the data log from the National Institute of Solar Energy (NISE), Gurgaon. The predicted values were compared to the actual values given in the data set by plotting the "predicted vs actual value" curves. The difference in the values, i.e., error was calculated. RMSE values for each of the dependent parameters was also calculated.

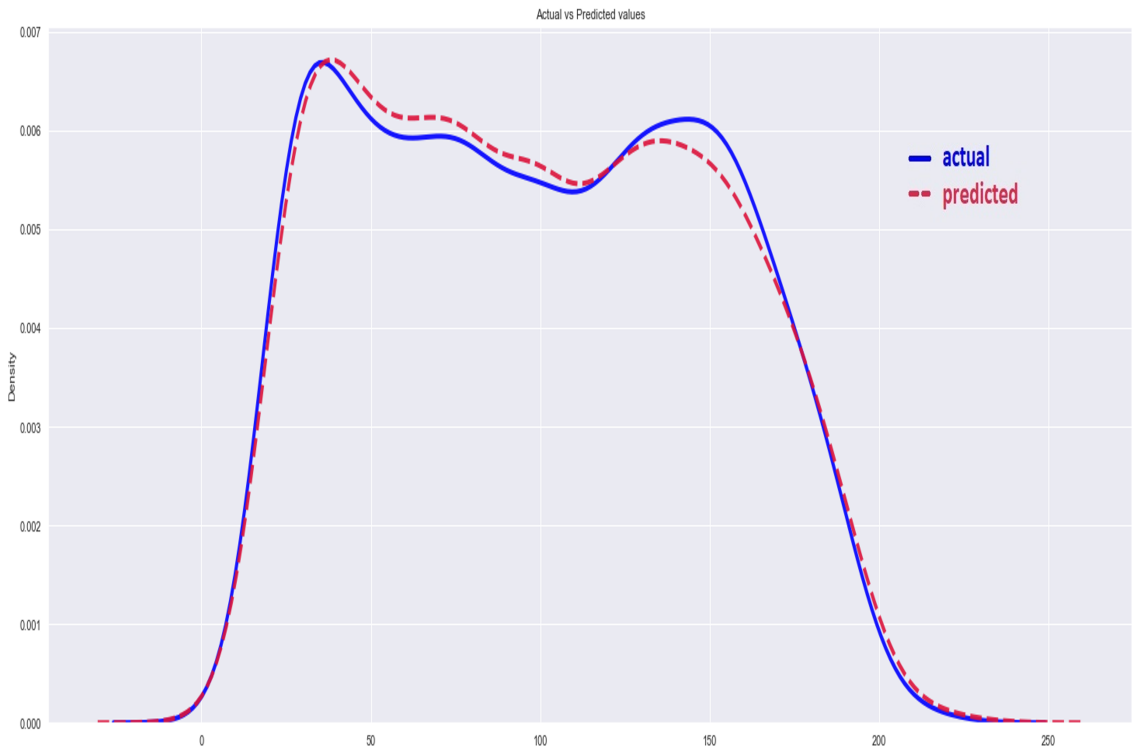

Figure 5.1: Actual vs Predicted for Maximum Power Using Linear Regression

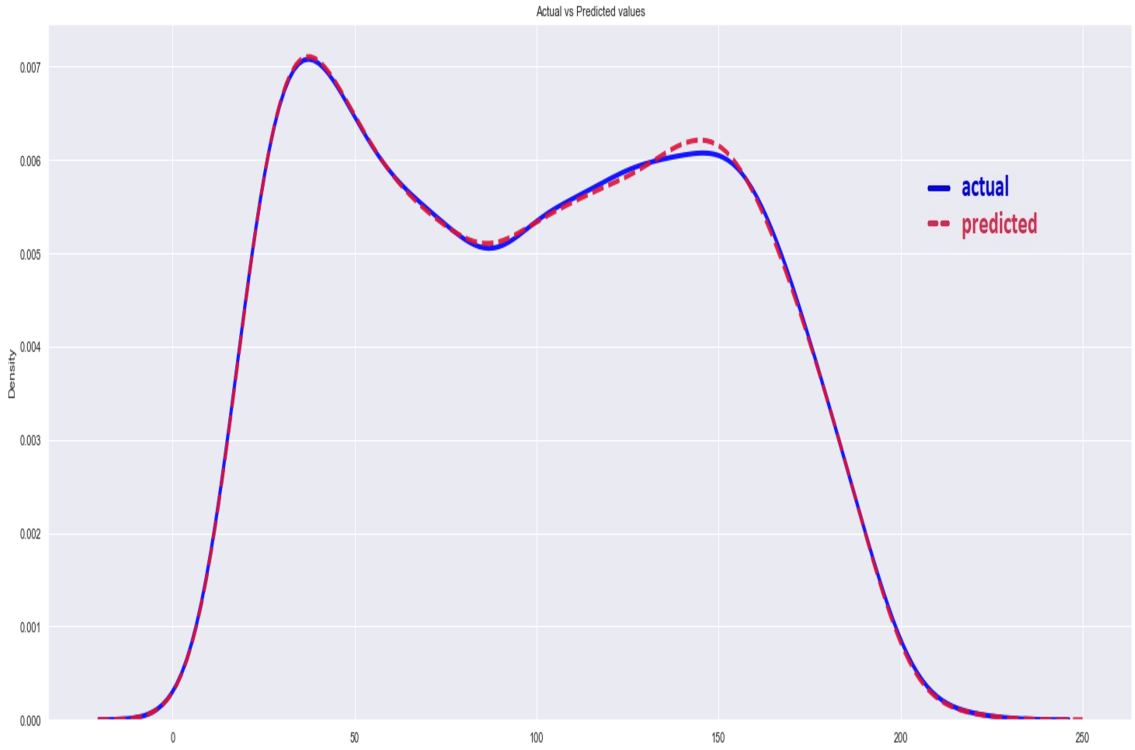

Figure 5.2: Actual vs Predicted for Maximum Power Using Random Forest Regression

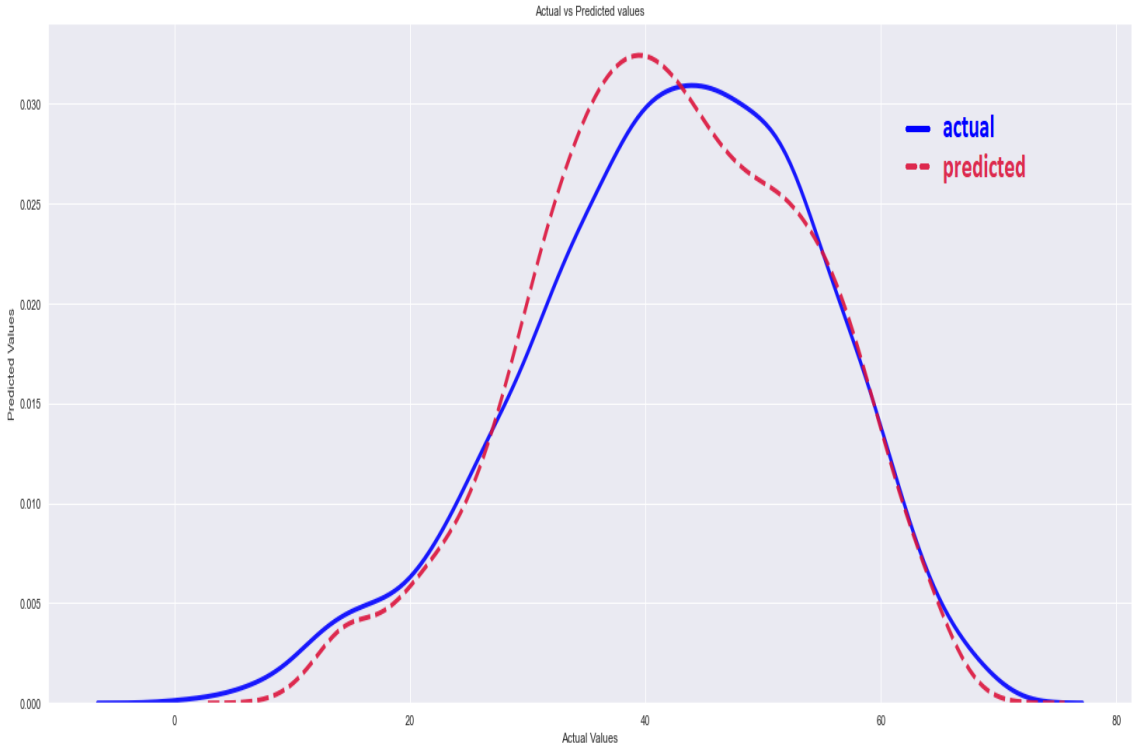

Figure 5.3: Actual vs Predicted for Module Temperature Using Linear Regression

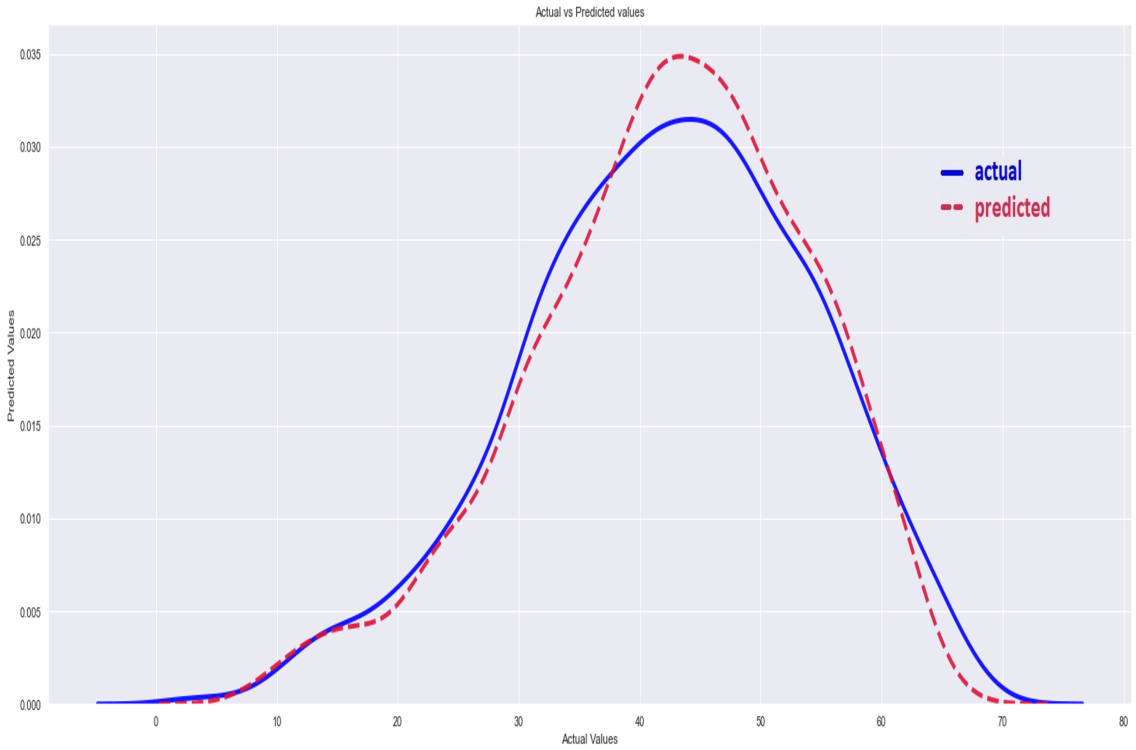

Figure 5.4: Actual vs Predicted for Module Temperature Using Random Forest Regression

As it is clearly observable from figure 5.1 and figure 5.2, random forest regression is much more accurate and efficient in predicting the module temperature than linear regression. Similarly, from figure 5.3 and figure 5.4, it can be observed that random forest regression is also more accurate and efficient in predicting the maximum power output of the module. Temperature of the module has a non-linear relationship with its inde-

Error Metric	Linear Regression (Module Temperature)	Linear Regression (Maximum Power)	Random Forest Regression (Module Temperature)	Random Forest Regression (Maximum Power)
RMSE	4.3	3.5	3.85	2.7

Figure 5.5: Table 1. Comparison of RMSE values

pendent parameters, so the RMSE value is higher as compared to maximum power which has a linear relationship with its independent parameters. The RMSE values for linear and random forest regression is given in table . The random forest regression has a much lower RMSE value than linear regression. Thus, random forest provides much better prediction and is more efficient. The comparison of RMSE values is given in table 1.

5.2 Season-Wise Results

5.2.1 Module Temperature

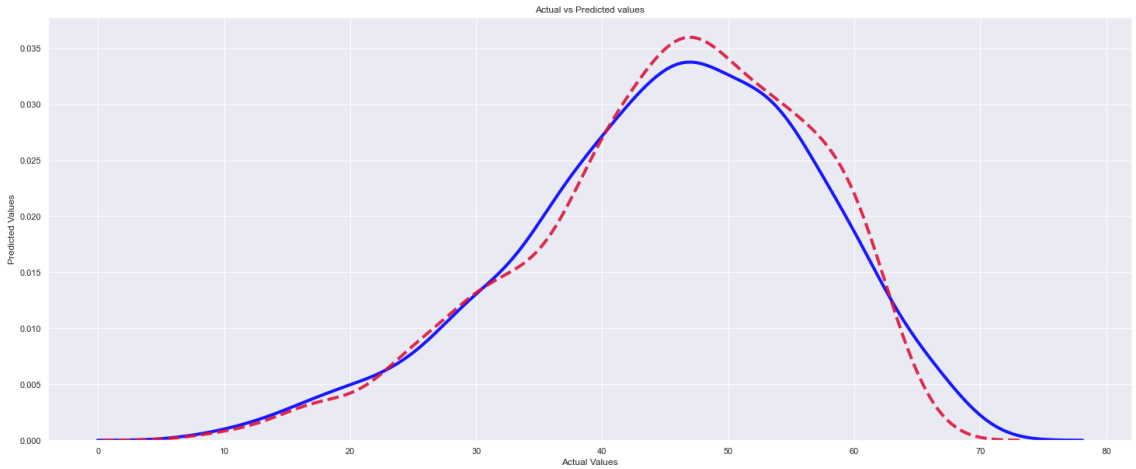

Figure 5.6: Actual vs Predicted for Module Temperature in Summer Season

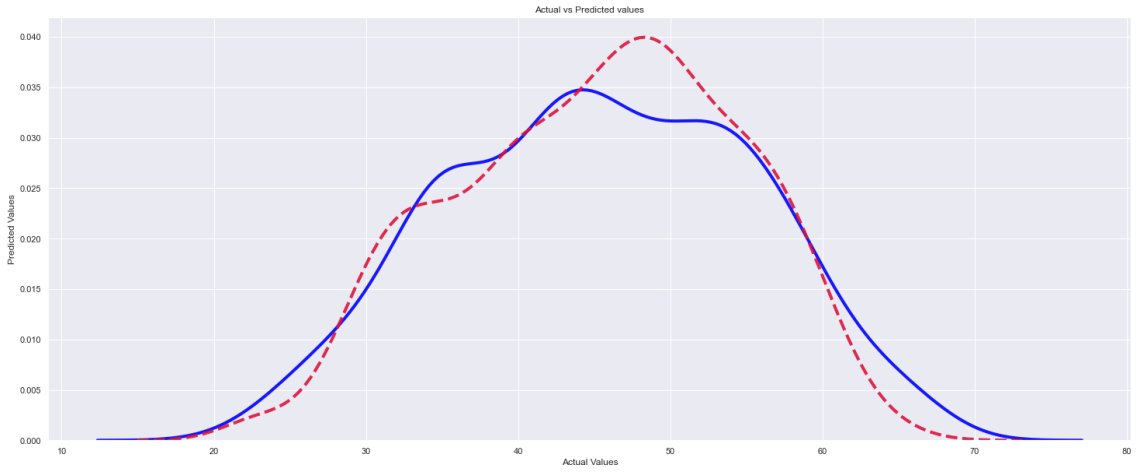

Figure 5.7: Actual vs Predicted for Module Temperature in Rainy Season

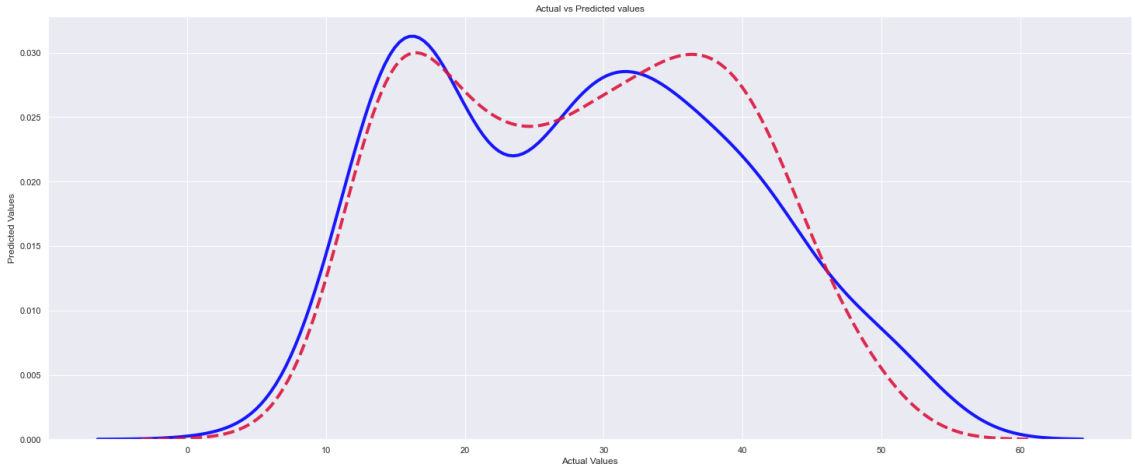

Figure 5.8: Actual vs Predicted for Module Temperature in Winter Season

5.2.2 Maximum Power

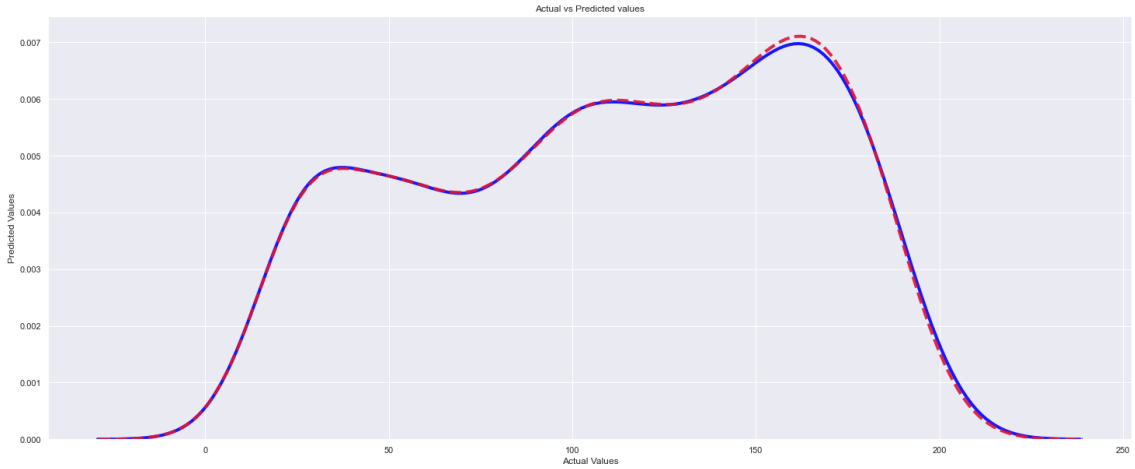

Figure 5.9: Actual vs Predicted for Maximum Power in Summer Season

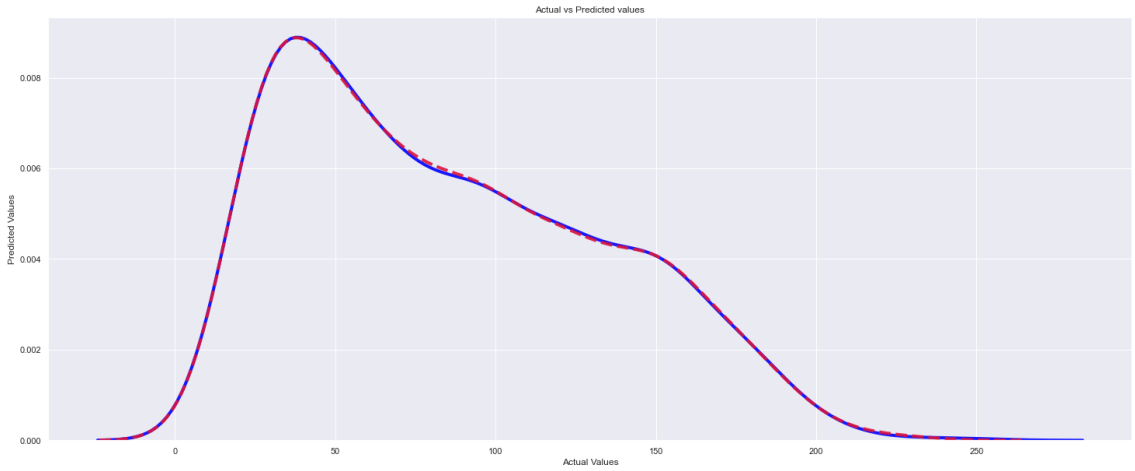

Figure 5.10: Actual vs Predicted for Maximum Power in Rainy Season

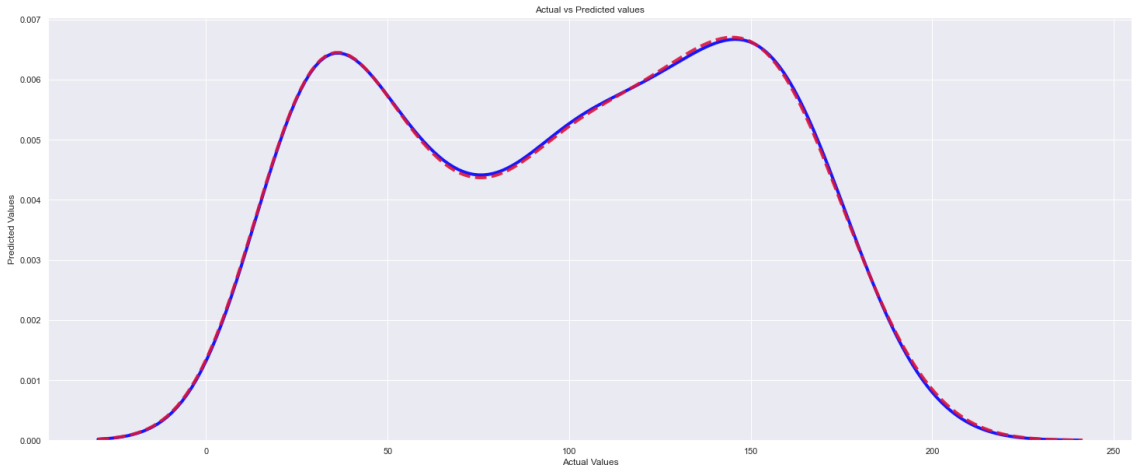

Figure 5.11: Actual vs Predicted for Maximum Power in Winter Season

Chapter 6

Conclusion

The assessment of PV module temperature and maximum power was included in this project. With the machine learning model of random forest regression, a simple model for predicting PV module temperature and maximum power has been developed that considers wind speed and direction, in-plane irradiance, and module efficiency. Overall, the model's results show a good match between actual and predicted module temperatures and maximum power outputs for HIT PV cells. The methodology given in this project can be used to anticipate module temperatures of HIT PV cells under varying weather conditions with great consistency. The RMSE values of both linear regression and random forest regression were compared in this project. It was observed that linear regression has much higher RMSE value than random forest regression which makes the latter more accurate and efficient in predicting module temperature and maximum power. Actual vs predicted graphs were plotted as well and again random forest regression was more accurate.

Chapter 7

Publications and Achievements

Publication

- ICACC 2022

Chapter 8

Plagiarism Report

Bibliography

[1] Green, M. A., 1982, Solar Cells: Operating Principles, Technology, and System Applications, Prentice Hall, Englewood Cliffs, NJ.

[2] King, D. L., Boyson, W. E., and Kratochvil, J. A., 2002, "Analysis of Factors Influencing the Annual Energy Production of Photovoltaic Systems," 29th IEEE Photovoltaic Specialists Conference, New Orleans, LA, May 19–24, pp. 1356–1361.

[3] Schwingshackl, C., Petitta, M., Wagner, J. E., Belluardo, G., Moser, D., Castelli, M., Zebisch, M., and Tetzlaff, A., 2013, "Wind Effect on PV Module Temperature: Analysis of Different Techniques for an Accurate Estimation," Energy Procedia, 40, pp. 77–86.

[4] Ross, R. G., Jr., and Smokler, M. I., 1986, "Flat- Plate Solar Array Project Final Report—Vol. VI: Engineering Sciences and Reliability," Jet Propulsion Laboratory, Pasadena, CA, Report No. DOE/JPL-1012-125.

[5] Nordmann, T., and Clavadetscher, L., 2003, "Understanding Temperature Effects on PV System Performance," Third World Conference on Photovoltaic Energy Conversion, Osaka, Japan, May 11–18, pp. 2243–2246.

[6] Duffie, J. A., and Beckman, W. A., 2006, Solar Energy Thermal Processes, 3rd ed., Wiley, Hoboken, NJ.

[7] Bahaidarah, H., Rehman, S., Subhan, A., Gandhidasan, P., and Baig, H., 2015, "Performance Evaluation of a PV Module Under Climatic Conditions of Dhahran, Saudi Arabia," Energy Explor. Exploit., 33(6), pp. 909–930.

[8] Jakhrani, A. Q., Othman, A. K., Rigit, A. R. H., and Samo, S. R., 2011, "Determination and Comparison of Different Photovoltaic Module Temperature Models for Kuching, Sarawak," IEEE First Conference Clean Energy and Technology (CET), Kuala Lumpur, Malaysia, June 27–29, pp. 231–236.

[9] Bora, B., Sastry, O. S., Kumar, A., Renu, M., Bangar, M., and Prasad, B., 2016, "Estimation of Most Frequent Conditions and Performance Evaluation of Three Photovoltaic Technology Modules," ASME J. Sol. Energy Eng., 138(5), p. 054504.

[10] IEC, 2009, "Procedures for Temperature and Irradiance Corrections to Measured I-V Characteristics," International Electrotechnical Commission, Geneva, Switzerland, Standard No. IEC 60891:2009.

[11] IEC, 2011, "Photovoltaic (PV) Module Performance Testing and Energy Rating—Part 1: Irradiance and Temperature Performance Measurements and Power Rating," International Electrotechnical Commission, Geneva, Switzerland, Standard No. IEC 61853-1:2011.

[12] Magare, D. B., Sastry, O. S., Gupta, R., Betts, T. R., Gottschalg, R., Kumar, A., Bora, B., and Singh, Y. K., 2016, "Effect of Seasonal Spectral Variations on Performance of Three Different Photovoltaic Technologies in India," Int. J. Energy Environ. Eng., 7(1), pp. 93–103.

[13] Rosell, J. I., and Ibanez, M., 2006, "Modelling Power Output in Photovoltaic Modules for Outdoor Operating Conditions," Energy Convers. Manage., 47(15–16), pp. 2424–2430.

[14] Magare, D., Sastry, O., Gupta, R., Bora, B., Singh, Y., and Mohammed, H. (December 22, 2017). "Wind Effect Modeling and Analysis for Estimation of Photovoltaic Module Temperature." ASME. J. Sol. Energy Eng. February 2018; 140(1): 011008.

[15] E. Skoplaki, J.A. Palyvos, On the temperature dependence of photovoltaic module electrical performance: A review of efficiency/power correlations, Solar Energy, Volume 83, Issue 5, 2009, Pages 614-624, ISSN 0038-092X

[16] Haddad, Sofiane Rabhi, Abdelhamid Soukkou, Ammar HASSAN ALI, Mohamed El hajjaji, Ahmed. (2018). Experimental Analysis and Assessment of a HIT Photovoltaic Module.