A Minor Project Report

On

**Solar Radiation prediction using sarimax and ann(hybrid)**

SUBMITTED IN PARTIAL FULFILLMENT FOR THE AWARD OF DEGREE OF

**Bachelor of Technology**

**IN**

**Electronics and Communication Engineering**



**Submitted By:** **Under the Guidance Of**

**NAME(s) (ENROLMENT) SUPERVISOR NAME**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA (U.P.)**

**November, 2019**

**CERTIFICATE**

This is to certify that the minor project report entitled, “**----------**” submitted by **-------** in partial fulfillment of the requirements for the award of Bachelor of Technology Degree in **Electronics and Communication Engineering** of the Jaypee Institute of Information Technology, Noida is an authentic work carried out by them under my supervision and guidance. The matter embodied in this report is original and has not been submitted for the award of any other degree.

**Signature of Supervisor:**

**Name of the Supervisor:**

**ECE Department,**

**JIIT, Sec-128,**

**Noida-201304**

**Dated:**

**DECLARATION**

We hereby declare that this written submission represents our own ideas in our own words and where others' ideas or words have been included, have been 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.

Place:

Date:

Name:

Enrollment:

Name:

Enrollment:

Name:

Enrollment:

**Abstract**

Solar radiation prediction has a great importance in electricity generation from solar energy and helps to size photovoltaic power systems. Therefore, the Global Horizontal Irradiance (GHI) was predicted at 1-hour duration in this paper.

Direct Normal Irradiance (DNI), Direct Horizontal Irradiance (DHI), Temperature, Pressure, Relative Humidity, Dew Point, Wind Direction and Wind Speed parameters were used as atmospheric input variables for **time series model** by **Auto Regressive Integrated Moving Average(ARIMA) model**. ARIMA is one of the popular linear models in time series forecasting.

As time series also have non linearity hence, a hybrid model is made afterwards using residuals of ARIMA as input for Artificial Neural Networks. Statistical error measures such as the mean error (ME), the mean square error (MSE) and the mean absolute error (MAE) were calculated to compare the two methods.

The results showed that the Hybrid models predict the solar radiations with a higher accuracy than the SARIMAX model in the four examined sites.

**TABLE OF CONTENTS**

**CHAPTER1: INTRODUCTION6**

1.1. Introduction to problem6

1.2. Our Approach9

**CHAPTER 2:Software and Language10**

2.1. Why Python10

2.2.Why Spyder is the best IDE10

**CHAPTER 3:Rcognizing the type of problem and data preprocessing11**

3.1. Regression11

3.2. Classification11

3.3 Dataset Preprocessing………………………………………………………………………..12

3.3.1 Train and Test Split………………………………………………………………..12

**CHAPTER 4: Algorithm Used13**

4.1. ARIMA13

4.2. SARIMAX15

4.3. Artificial Neural Network16

4.3.1. Hybrid SARIMAX-ANN17

**CHAPTER 5: Result Analysis20**

5.1. SARIMAX20

5.2. Hybrid SARIMAX ANN22

**CHAPTER 6: Conclusion and Future Scope24**

6.1. Conclusions24

6.2.Future Scopes24

**References25**

**Appendix26**

**Chapter 1**

**Introduction**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**1.1 Introduction**

Solar radiation prediction is an important problem with direct applications in renewable energy. Solar is one of the most important green sources of energy, that is currently under expansion in many countries of the world, especially in those with more solar potential, such as Rajasthan. An accurate estimation of the energy production in solar energy systems involves the accurate prediction of solar radiation, depending on different atmospheric variables.

We have accessed the data of **four cities (Jaiselmer, Jodhpur, Bikaner, Barmer)** of **Rajasthan** which provides data for the year 2010-2014.

The dataset counts to 43801 rows for each city. Our dataset consists of 14 columns ,out of which 8 columns are treated as independent features(**X**) which are Direct Normal Irradiance (DNI), Diffuse Horizontal Irradiance (DHI), Temperature, Pressure, Relative Humidity, Dew Point, Wind Direction and Wind Speed as inputs ([Khatib](https://www.sciencedirect.com/science/article/abs/pii/S1364032112000767) [1]et al., 2012; [Inman](https://www.sciencedirect.com/science/article/pii/S0360128513000294) [2]et al., 2013; [Sozen](https://www.sciencedirect.com/science/article/pii/S0196890404000172) [3]et al., 2004; [Voyant](https://www.sciencedirect.com/science/article/abs/pii/S0360544210005955) [4]et al., 2011) and one as dependent feature(**Y**) which is Global Horizontal Irradiance(GHI).

These nine columns for our training model are:

1)**Diffuse Horizontal Irradiance** (DHI): Diffuse Horizontal Irradiance is the amount of radiation received per unit area by a surface that does not arrive on a direct path from the sun, but has been scattered by molecules and particles in the atmosphere and comes equally from all directions

2)**Direct Normal Irradiance** (DNI): Direct Normal Irradiance is the amount of solar radiation received per unit area by a surface that is always held perpendicular (or normal) to the rays that come in a straight line from the direction of the sun.

3)**Global Horizontal Irradiance** (GHI)(**Y)**: Global Horizontal Irradiance is the total amount of shortwave radiation received from above by a surface horizontal to the ground. This value is of particular interest to photovoltaic installations and includes both Direct Normal Irradiance (DNI) and Diffuse Horizontal Irradiance (DHI).

**Global Horizontal (GHI) = Direct Normal (DNI) X cos(θ) + Diffuse Horizontal (DHI)**

4)**Dew Point**: Dew-point temperature refers to the effect of water vapor on solar radiation. It can cool the panel down but can also decrease the area for solar radiation intensity.

5)**Temperature**: The degree or intensity of heat present in a substance or object. Excessive heat can cause decrease in solar output.

6)**Pressure**: It is the continuous physical force exerted on or against solar panel by something in contact with it.

7)**Relative Humidity**: The amount of water vapour present in air expressed as a percentage of the amount needed for saturation at the same temperature. High relative humidity is associated with low global solar radiation and low relative humidity is associated with high global solar radiations.

8)**Wind Direction**: Cooler solar panel improves efficiency

9)**Wind Speed**: When solar panel is too hot, it reduces efficiency, on the other hand cooler solar panel improves efficiency

In recent years, several works have been developed to try to predict solar radiation using machine learning techniques and environmental parameters. They used different input geographical and atmospheric parameters like latitude, longitude, temperature, wind speed and direction, daily global irradiation, sunshine duration or precipitation ([Mellit and Kalogirou](https://www.sciencedirect.com/science/article/pii/S0360128508000026)[5] ,2008; [Mubiru](https://www.sciencedirect.com/science/article/abs/pii/S0960148108000074)[6], 2008). According to [Bilgili and Ozoren](https://link.springer.com/article/10.1007%2Fs00703-011-0137-9)[7] (2011), sunshine duration, air temperature and relative humidity are the most widely used meteorological parameters to predict daily solar radiation and its components.

An initial category to distinguish is time-series models, which can be further divided into Box and Jenkins techniques, smoothing techniques, Kalman-ﬁltering theory, and spectral analysis. Early applications of Box and Jenkins techniques in the ﬁeld of traffic forecasting were implemented by [Ahmed and Cook](https://trid.trb.org/view/148123)[8] and [Nihan and Holmesland](https://sci-hub.se/https:/link.springer.com/article/10.1007/BF00167127)[9]. More advanced techniques have been applied recently, including autoregressive integrated moving average (ARIMA) models with intervention x-variables (ARIMAX) by [Williams, B. M., and L. A. Hoel](https://sci-hub.se/https:/ascelibrary.org/doi/abs/10.1061/(ASCE)0733-947X(2003)129:6(664))[10], seasonal ARIMA models (SARIMA) by [Williams, B. M](https://sci-hub.se/https:/ascelibrary.org/doi/abs/10.1061/(ASCE)0733-947X(2003)129:6(664))[11], seasonal ARIMA with intervention x-variables(SARIMAX) by [SI Vagropoulos, GI Chouliaras](https://sci-hub.se/https:/ieeexplore.ieee.org/abstract/document/7514029/)[12].

Neural network models are the second category of techniques that can be identiﬁed. [Y Jiang](https://sci-hub.se/https:/www.sciencedirect.com/science/article/pii/S0301421508003133)[13] was among the first who applied ANN on solar radiation prediction. [Amit Kumar Yadav , Hasmat Malik](https://sci-hub.se/https:/www.sciencedirect.com/science/article/pii/S1364032113008228)[14] wrote about the most relevant input parameters for artificial neural network based solar radiation prediction models.

A suitable combination of linear and nonlinear models provides a more accurate prediction model than an individual linear or nonlinear model for forecasting time series data originating from various applications. The linear autoregressive integrated moving average (ARIMA) and nonlinear artificial neural network (ANN) models are explored in this paper to devise a new hybrid ARIMA–ANN model for the prediction of time series data. The results obtained from all of these data sets show that for both one-step-ahead and multistep-ahead forecasts, the proposed hybrid model has higher prediction accuracy.

A hybrid ARIMA–ANN model was proposed by [Zhang](https://www.sciencedirect.com/science/article/abs/pii/S0925231201007020?via%3Dihub) [15], which was shown to give more accurate predictions than the individual models. On Wolf’s sunspot data, Canadian lynx data, and exchange rate time series data, this hybrid model was shown to outperform individual ARIMA and ANN models in the case of one-step-ahead prediction. Another hybrid ARIMA–ANN method was proposed by [Khashei and Bijari](https://www.sciencedirect.com/science/article/abs/pii/S1568494610002759?via%3Dihub) [16], which was shown to give better performance for one-step-ahead forecasting than the method proposed by Zhang. The hybrid method proposed by Zhang was also used for [electricity price forecasting](http://dx.doi.org/10.1049/iet-gtd.2012.0263) in [17] and for [water quality time series prediction](https://www.sciencedirect.com/science/article/abs/pii/S0952197609001390?via%3Dihub) in [18].

For error detection,we have used MAAPE and RMSE values. The mean absolute percentage error (MAPE) is one of the most widely used measures of forecast accuracy, due to its advantages of scale-independency and interpretability. However, MAPE has the significant disadvantage that it produces infinite or undefined values for zero or close-to-zero actual values. In order to address this issue in MAPE, we have used a new measure of forecast accuracy called the mean arctangent absolute percentage error (MAAPE) by [Sungil Kima and Heeyoung Kim](https://sci-hub.se/https:/www.sciencedirect.com/science/article/pii/S0169207016000121)[19]. MAAPE has been developed through looking at MAPE from a different angle. In essence, MAAPE is a slope as an angle, while MAPE is a slope as a ratio, considering a triangle with adjacent and opposite sides that are equal to an actual value and the difference between the actual and forecast values, respectively. MAAPE inherently preserves the philosophy of MAPE, overcoming the problem of division by zero by using bounded influences for outliers in a fundamental manner through considering the ratio as an angle instead of a slope.

**1.2 Our Approach**

Fig 1.1 Process Flow Chart

**Chapter 2**

**Software and Language**

In this chapter, a point to point depiction of our dataset and preprocessing of dataset is explained.

**2.1 Why Python?**

It is popular in machine learning because of many inter-related reasons. Python is simple, elegant, consistent, and math-like Python code has been described as readable pseudo-code. It is easy to pick up due to its consistent syntax and the way it mirrors human language and/or their mathematical counterparts. The latter (much due to libraries such as Numpy) is something one will appreciate if he were to implement a machine. Any person has to atleast spent 1 - 2 days writing Java code from scratch to do the things for a ML competition whereas it would took barely 1 min to fire up in Python. Python is math-like in that some "objects" that are very much part of a mathematician's vocabulary are part of the language without you having to install / import them, and they resemble their equivalent mathematical counterparts. With carefully chosen variable/function names, the code can read like math or english.

**2.2 Why Spyder is the Best Python IDE?**

In Anaconda distribution, Spyder could perfectly be one's first approach. It’s an open source cross-platform IDE for data science. It integrates the essentials libraries for data science, such as **NumPy**, **SciPy**, **Matplotlib** and **IPython**, besides that, it can be extended with plugins. Different of most of IDEs around the web, Spyder was built specifically for data science.

We have also used libraries like **Statsmodels** for **Sarimax** and **Keras** for **Artificial Neural Networks**  where:

Statsmodels: [statsmodels](https://www.statsmodels.org/stable/about.html#module-statsmodels) is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration.

Keras:Keras is an open-source neural-network library written in Python. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular.

**Chapter 3**

**Recognizing the type of problem and dataset preprocessing**

**3.1 Regression**

* A regression problem requires the prediction of a quantity.
* A regression can have real valued or discrete input variables.
* A problem with multiple input variables is often called a multivariate regression problem.

**3.2 Classification**

* A classification problem requires that examples be classified into one of two or more classes.
* A classification can have real-valued or discrete input variables.
* A problem with two classes is often called a two-class or binary classification problem.
* A problem with more than two classes is often called a multi-class classification problem.
* A problem where an example is assigned multiple classes is called a multi-label classification problem.

**Since our dataset contains time series quantities , our problem is Regression time-series problem**

**3.3 Dataset Preprocessing**

Our dataset is clean and we just have to split it into test and train values.

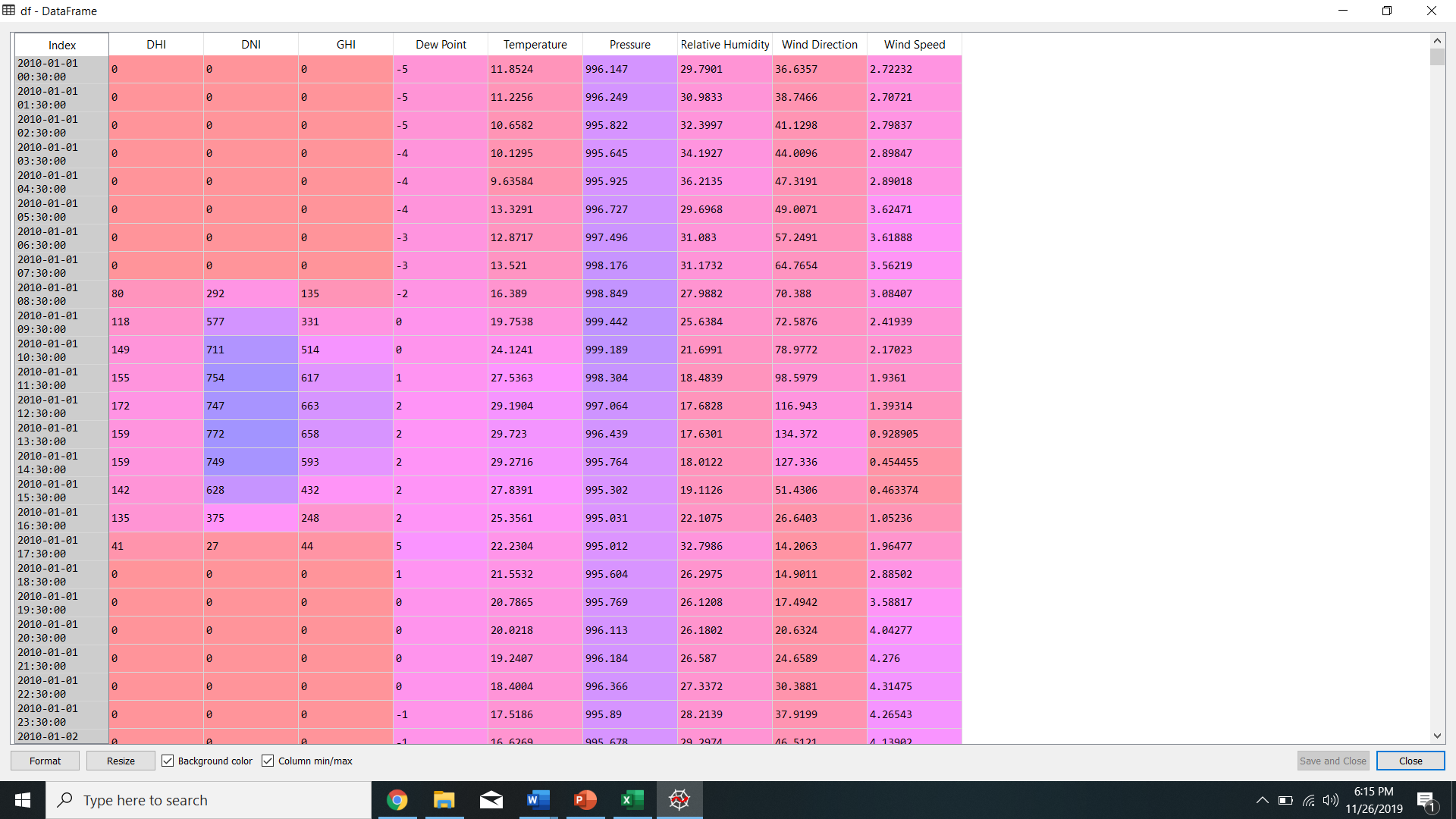


Fig: Dataset

**3.3.1 Train and Test split** The dataset is splitted into two sets- Train set and Test set. The algorithm are trained on the training set and after training the algorithm will be tested on the test set. Since we are using a big dataset we will cut it into 90% and 10%

**Train Set-** About 90% of the dataset is used for training the algorithm.

**Test Set-**The remaining 10% of dataset is used for testing the algorithm and predicting the results.

**Chapter 4**

**Algorithms Used**

In this chapter, a point by point depiction of different regression machine learning algorithms are explained.

**4.1 Auto Regressive Integrated Moving Average (ARIMA)**

ARIMA is a simple stochastic time series model that we can use to train and then forecast future time points. ARIMA can capture complex relationships as it takes error terms and observations of lagged terms. These models rely on regressing a variable on past values.

# ****AR****IMA is Auto Regressive — (AR)

Auto Regressive (AR) property of ARIMA is referred to as **P**. Past time points of time series data can impact current and future time points. ARIMA models take this concept into account when forecasting current and future values.

ARIMA uses a number of lagged observations of time series to forecast observations.

A weight is applied to each of the past term and the weights can vary based on how recent they are.

ARIMA relies on Auto Regression. Autoregression is a process of regressing a variable on past values of itself. Autocorrelations gradually decay and estimate the degree to which white noise characterizes a series of data.

# ARIMA is Integrated — (I)

If a trend exists then time series is considered non stationary and shows seasonality. Integrated is a property that reduces seasonality from a time series. ARIMA models have a degree of differencing which eliminates seasonality.

**D** property of ARIMA represents degree of differencing.

# ARIMA is Moving Average — (MA)

Error terms of previous time points are used to predict current and future point’s observation. Moving average (MA) removes non-determinism or random movements from a time series. The property **Q** represents Moving Average in ARIMA. It is expressed as MA(x) where x represents previous observations that are used to calculate current observation.

Moving average models have a fixed window and weights are relative to the time. This implies that the MA models are more responsive to current event and are more volatile.

**P (Auto Regressive), D (Integrated) and Q (Moving Average)** are the three properties of ARIMA model

Coefficients are calculated recursively. Model is chosen such that the estimated results calculated from the model are closer to the actual observed values.

This process is iterative in nature.

**The forecasting equation is constructed as follows**

First, let ydenote the dth difference of Y, which means:

If d=0:  yt  =  Yt

If d=1:  yt  =  Yt - Yt-1

If d=2:  yt  =  (Yt - Yt-1) - (Yt-1 - Yt-2)  =  Yt - 2Yt-1 + Yt-2

Note that the second difference of Y (the d=2 case) is not the difference from 2 periods ago.  Rather, it is the *first-difference-of-the-first difference*, which is the discrete analog of a second derivative, i.e., the local acceleration of the series rather than its local trend.

In terms of y, the general forecasting equation is:

ŷt   =   μ + ϕ1 yt-1 +…+ ϕp yt-p - θ1et-1 -…- θqet-q

Here the moving average parameters (θ’s) are defined so that their signs are negative in the equation, following the convention introduced by Box and Jenkins.  Some authors and software (including the R programming language) define them so that they have plus signs instead.

 When actual numbers are plugged into the equation, there is no ambiguity, but it’s important to know which convention your software uses when you are reading the output.

To identify the appropriate ARIMA model for Y, you begin by determining the order of differencing (d) needing to stationarize the series and remove the gross features of seasonality, perhaps in conjunction with a variance-stabilizing transformation such as logging or deflating. If you stop at this point and predict that the differenced series is constant, you have merely fitted a random walk or random trend model.

However, the stationarized series may still have autocorrelated errors, suggesting that some number of AR terms (p ≥ 1) and/or some number MA terms (q ≥ 1) are also needed in the forecasting equation.

## **What’s Wrong with ARIMA**

Autoregressive Integrated Moving Average, or ARIMA, is a forecasting method for univariate time series data. What if there were other variables that could relate with the target variable?

As its name suggests, it supports both an autoregressive and moving average elements. The integrated element refers to differencing allowing the method to support time series data with a trend.

A problem with ARIMA is that it does not support seasonal data. That is a time series with a repeating cycle.

ARIMA expects data that is either not seasonal or has the seasonal component removed, e.g. seasonally adjusted via methods such as seasonal differencing.

That’s why, we have used **SARIMAX**

**4.2 Seasonal Auto Regressive Integrated Moving Average with Exogenous variable (SARIMAX)**

The ARIMA part of the SARIMAX is the same. Rather, two more properties are also included that makes it more special and good for predicting linear models.

**Seasonality**

Seasonality in a time series is a regular pattern of changes that repeats over S time periods, where S defines the number of time periods until the pattern repeats again.

For example, there is seasonality in monthly data for which high values tend always to occur in some particular months and low values tend always to occur in other particular months. In this case, S = 12 (months per year) is the span of the periodic seasonal behavior. For quarterly data, S = 4 time periods per year.

**Exogenous Variables: -**

An exogenous variable is one whose value is determined outside the model and is imposed on the model. In other words, variables that affect a model without being affected by it.

### **Seasonal Elements**

There are four seasonal elements that are not part of ARIMA that must be configured; they are:

**P**: Seasonal autoregressive order.

**D**: Seasonal difference order.

**Q**: Seasonal moving average order.

**m**: The number of time steps for a single seasonal period.

**The SARIMA model is specified (p,d,q)×(P,D,Q)s(p,d,q)×(P,D,Q)s.**

ϕp(L)ϕ~P(Ls)ΔdΔDsyt=A(t)+θq(L)θ~Q(Ls)ζtϕp(L)ϕ~P(Ls)ΔdΔsDyt=A(t)+θq(L)θ~Q(Ls)ζt

In terms of a univariate structural model, this can be represented as

ytϕp(L)ϕ~P(Ls)ΔdΔDsut=ut+ηt=A(t)+θq(L)θ~Q(Ls)ζtyt=ut+ηtϕp(L)ϕ~P(Ls)ΔdΔsDut=A(t)+θq(L)θ~Q(Ls)ζt

where ηtηt is only applicable in the case of measurement error (although it is also used in the case of a pure regression model, i.e. if p=q=0).

In terms of this model, regression with SARIMA errors can be represented easily as

ytϕp(L)ϕ~P(Ls)ΔdΔDsut=βtxt+ut=A(t)+θq(L)θ~Q(Ls)ζtyt=βtxt+utϕp(L)ϕ~P(Ls)ΔdΔsDut=A(t)+θq(L)θ~Q(Ls)ζt

this model is the one used when **exogenous** regressors are provided.

Note that the reduced form lag polynomials will be written as:

Φ(L)≡ϕp(L)ϕ~P(Ls)Θ(L)≡θq(L)θ~Q(Ls)

We have performed SARIMAX on our dataset and following results were observed in figures

**4.3 Artificial Neural Network (ANN)**

ANNs Approach to Time-Series Modeling ANNs are flexible computing frameworks for modeling a broad range of nonlinear problems.

One significant advantage of the ANN models over other classes of nonlinear model is that ANNs are universal approximators that can approximate a large class of functions with a high degree of accuracy. Their power comes from the parallel processing of the information from the data.

No prior assumption of the model form is required in the model building process. Instead, the network model is largely determined by the characteristics of the data. The network architecture of ANNs consists of the three-layer fully connected feed forward neural network.

The model is characterized by a network of three layers of simple processing units connected by acyclic links.

The relationship between the output 𝑦𝑡 and the inputs **(𝑦𝑡−1 , 𝑦𝑡−2 , ⋯ , 𝑦𝑡−𝑝)** has the following mathematical representation as in

**𝑦𝑡 = ∝0+ ∝𝑗 𝑔 (𝛽0𝑗 + 𝛽𝑖𝑗 𝑦𝑡−1 𝑝 𝑖=1 𝑞 𝑗=1) +∈𝑡**

**∝𝑗 (𝑗 = 1,2,3, … , 𝑞) and 𝛽𝑖𝑗 (𝑖 = 1,2,3, . . ; = 1,2,3, … , 𝑞)** are the model parameters, often called the connection weights, p is the number of input nodes, and q is the number of hidden nodes.

The logistic function is often used as the hidden-layer transfer function, and is given by Eq.

**𝑔 𝑥 = 1 /1 + exp −𝑥 6 .**

Hence, the ANN model in fact, performs a nonlinear functional mapping from the past observations 𝑦𝑡−1 , 𝑦𝑡−2 , ⋯ , 𝑦𝑡−𝑝 to the future value 𝑦𝑡 , given by Eq.

**𝑦𝑡 = 𝑓 𝑦𝑡−1 , 𝑦𝑡−2 , ⋯ , 𝑦𝑡−𝑝 , 𝜔 +∈𝑡**

Where 𝜔 is a vector of all parameters, and 𝑓 𝑦𝑡−1 , 𝑦𝑡−2 , ⋯ , 𝑦𝑡−𝑝 , 𝜔 is a function determined by the network structure and connection weights and ∈ is the random error.

The choice 𝑞 of is data dependent and there is no systematic rule in deciding this parameter.

In addition to choosing an appropriate number of hidden nodes, another important task of ANN modeling of a time series is the selection of the number of lagged observations p and the dimension of the input vector.

This is perhaps the most important parameter to be estimated in an ANN model because it plays a major role in determining the (nonlinear) autocorrelation structure of the time series. However, there is no theory that can be used to guide the selection of p.

**4.4 Hybrid Model**

ARIMA and ANN are good in linear and nonlinear domains but none of them is a universal model that is suitable for all circumstances. The approximation of ARIMA model for complex nonlinear problems may not yield good results similarly, ANN will give mixed results in the linear domain.

Since it is difficult to completely know the characteristics of the data in a real problem, a hybrid methodology that has both linear and nonlinear modeling capabilities can be a good strategy for practical use. By combining different models, different aspects of the underlying patterns may be captured.

Practically a time series may be considered to have a linear component and a non-linear component as shown in the Eq.

**𝑦𝑡 = 𝑙𝑡 + 𝑛𝑡**

Where 𝑙𝑡 is the linear part and 𝑛𝑡 is the nonlinear part. The two parts can be estimated separately from the data. First the linear part is determined that is separated from the time series. This residual is fitted with the ANN model.

This is fundamental strategy behind this model. Let 𝑒𝑡 be the residuals which can be obtained from the time series by subtracting forecasted value 𝑙 𝑡 from sarimax model as shown in Eq.

**𝑒𝑡 = 𝑦𝑡 − 𝑙 𝑡**

If there are linear correlations left in residuals, then the linear models are not sufficient in forecasting the data. The residual analysis is sufficient enough to capture the nonlinear patterns in the data.

Therefore, even if a model has passed the diagnostic checking, the model may still not be adequate in that nonlinear relationships have not been appropriately modeled. Any significant nonlinear pattern in the residuals will indicate the limitation of the ARIMA.

By modeling residuals using ANNs, nonlinear relationships can be discovered.

**Chapter 5**

**Result Analysis**

The predicted plots are shown below

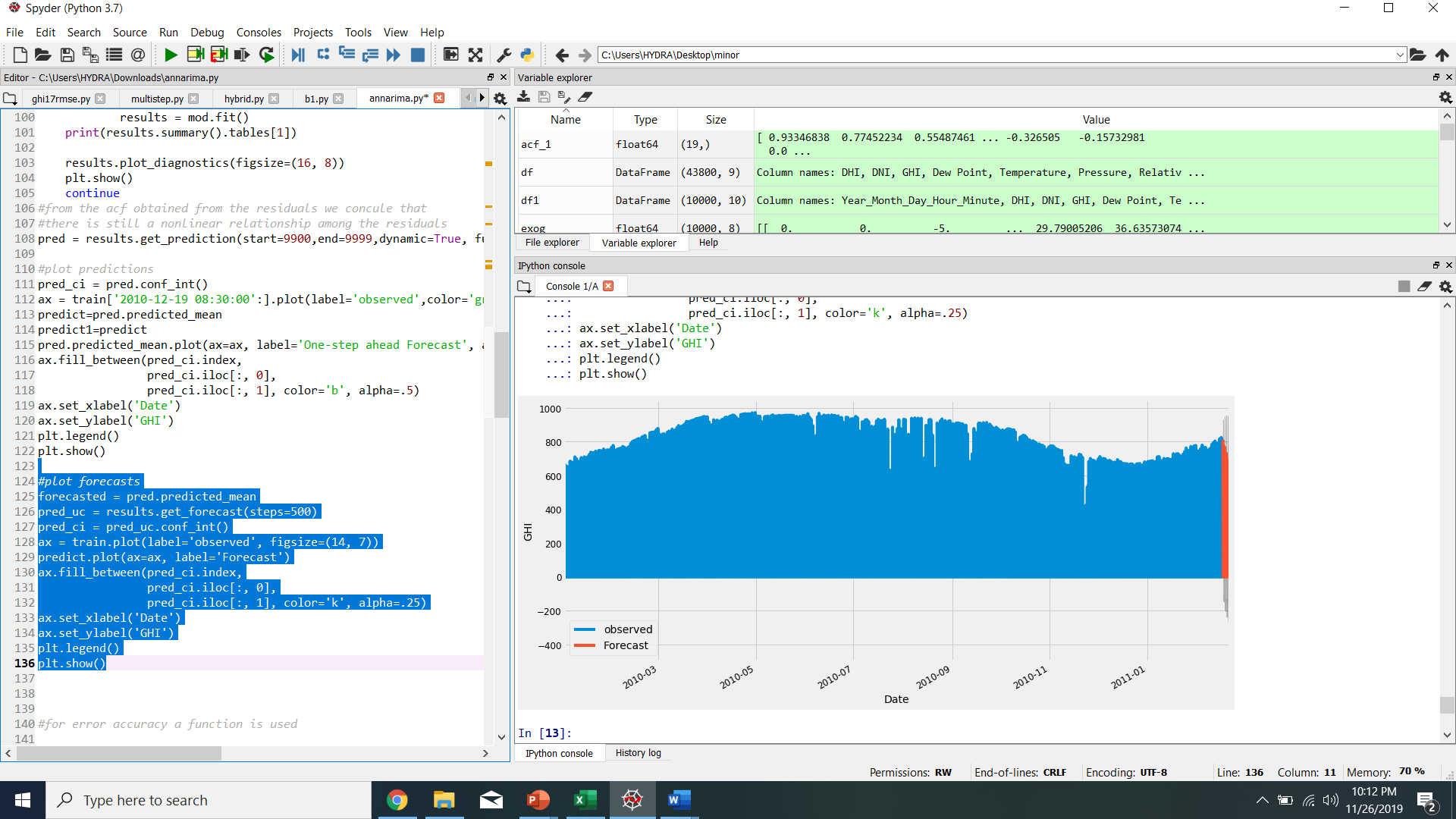


Fig 1: Barmer SARIMAX GHI forecasted

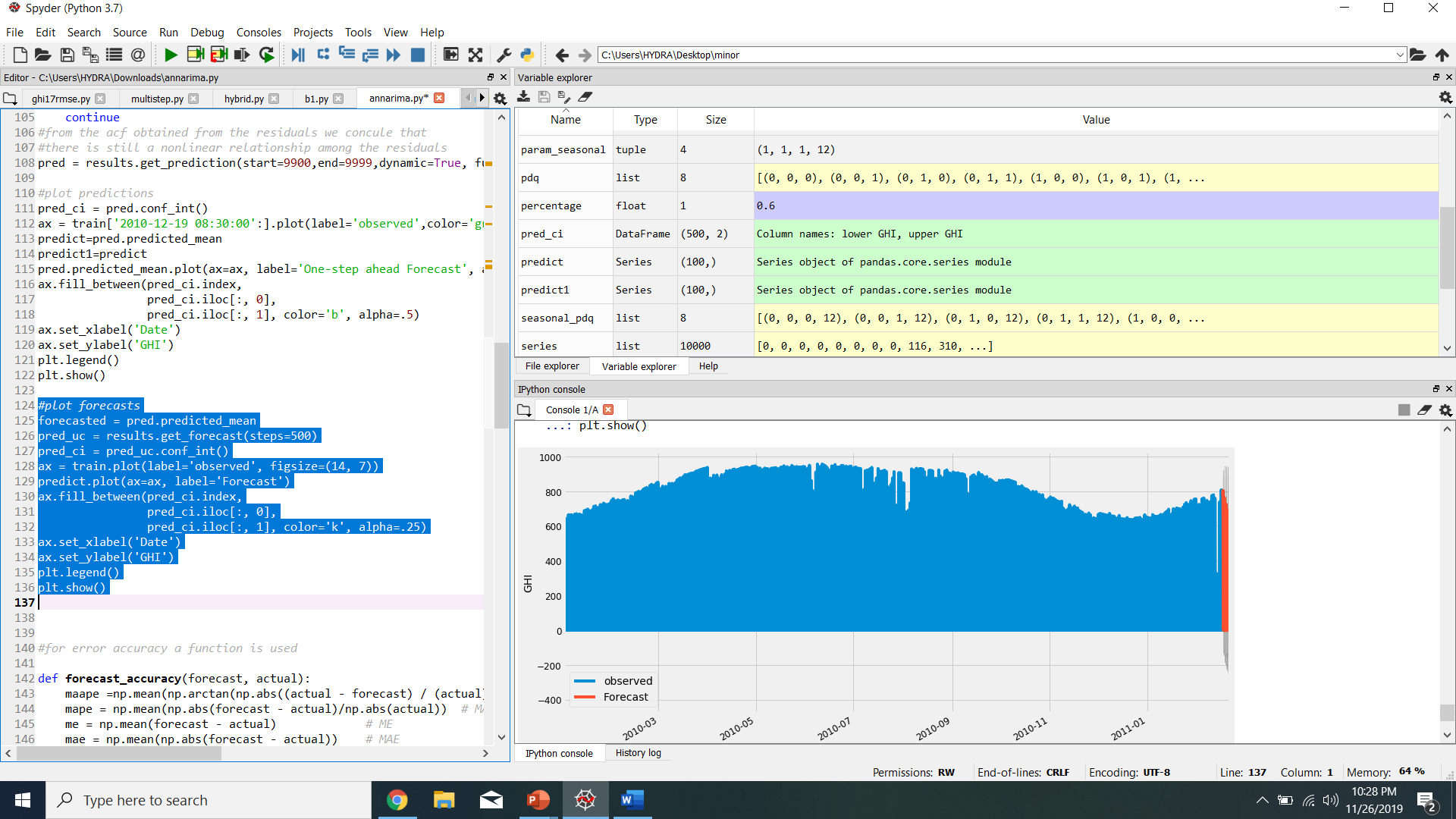


Fig 2: Bikaner SARIMAX forecasted

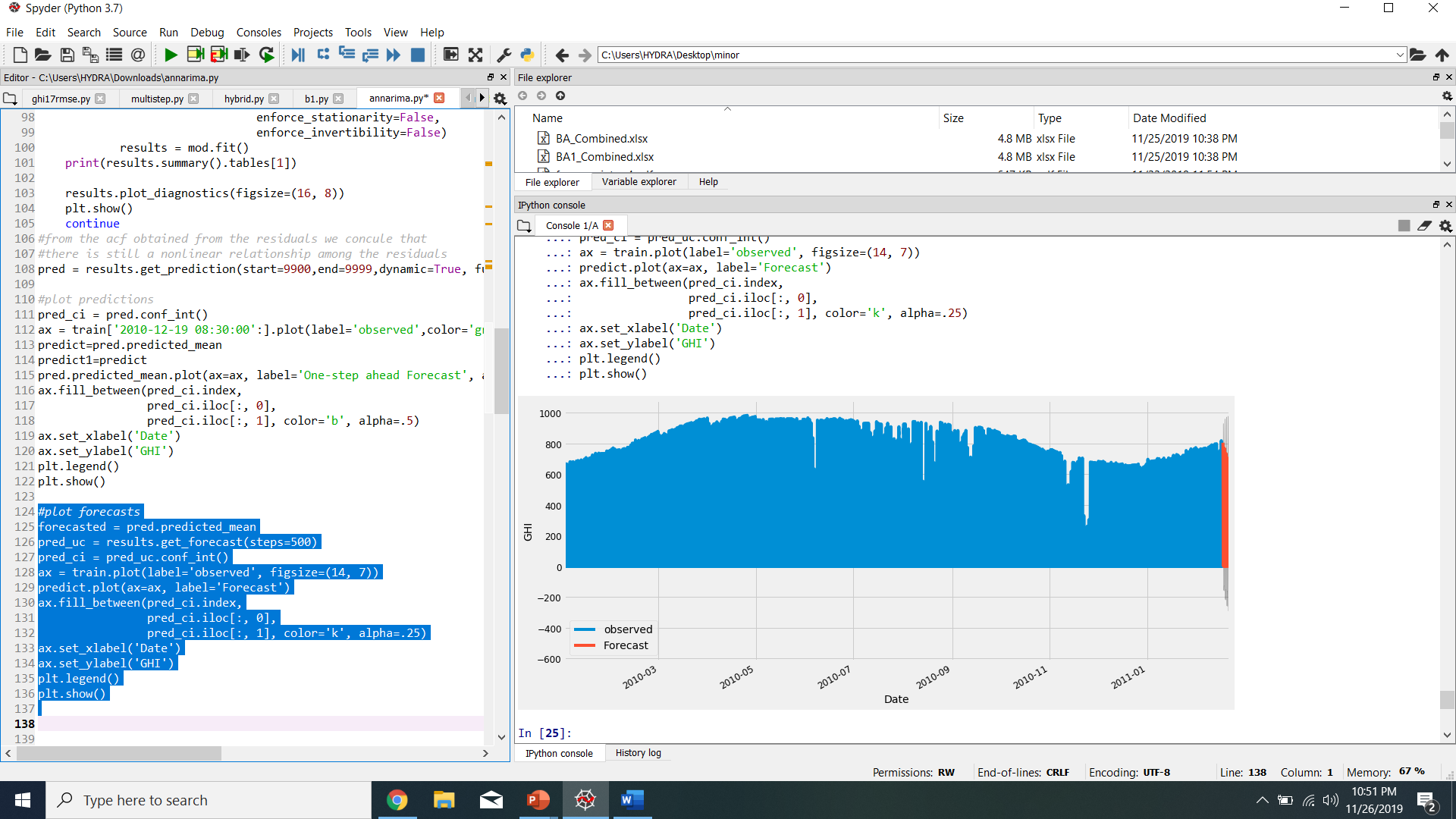


Fig 3: Jodhpur SARIMAX GHI FORECASTED

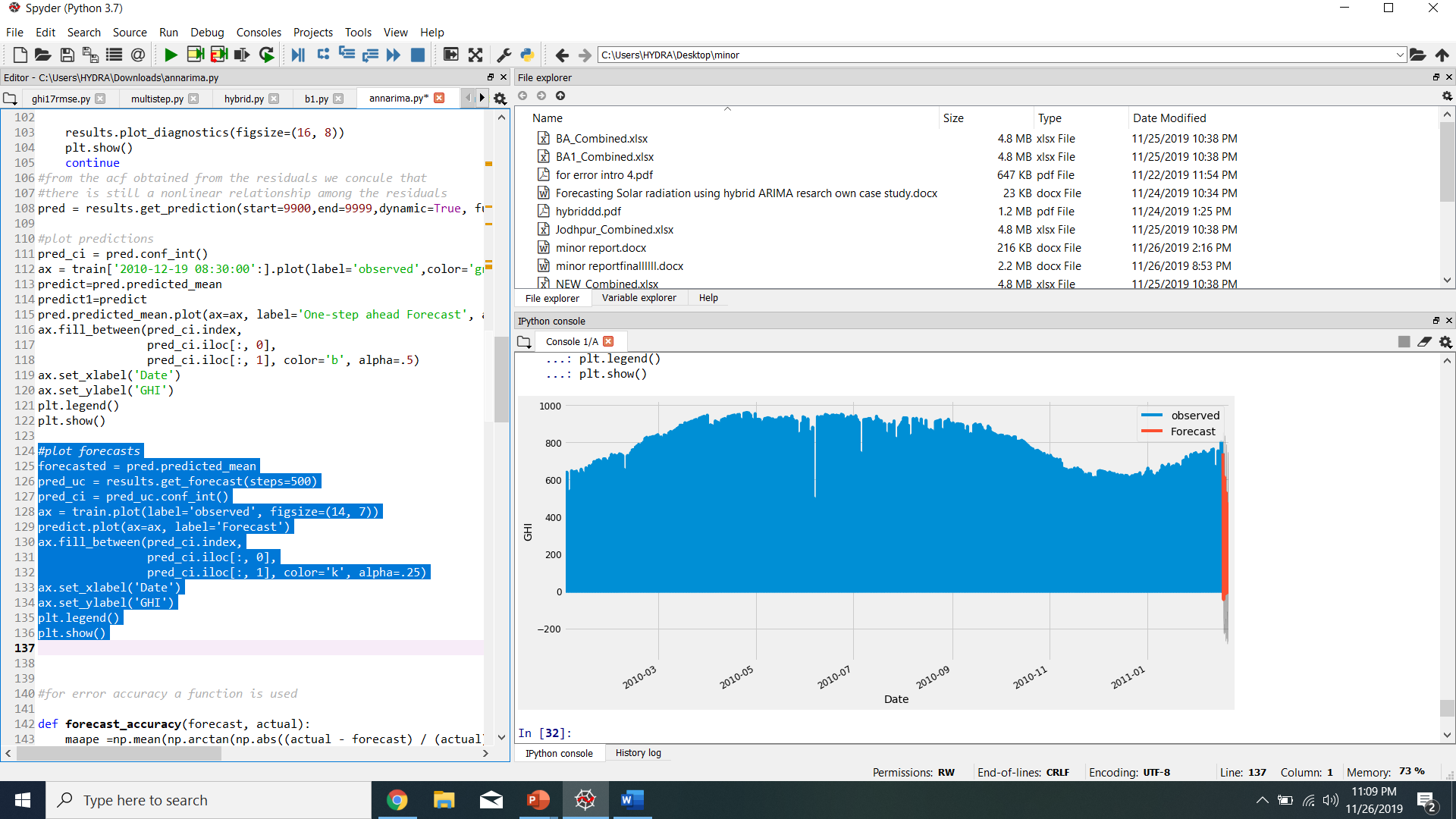


Fig4 : Jaiselmer SARIMAX GHI FORECASTED

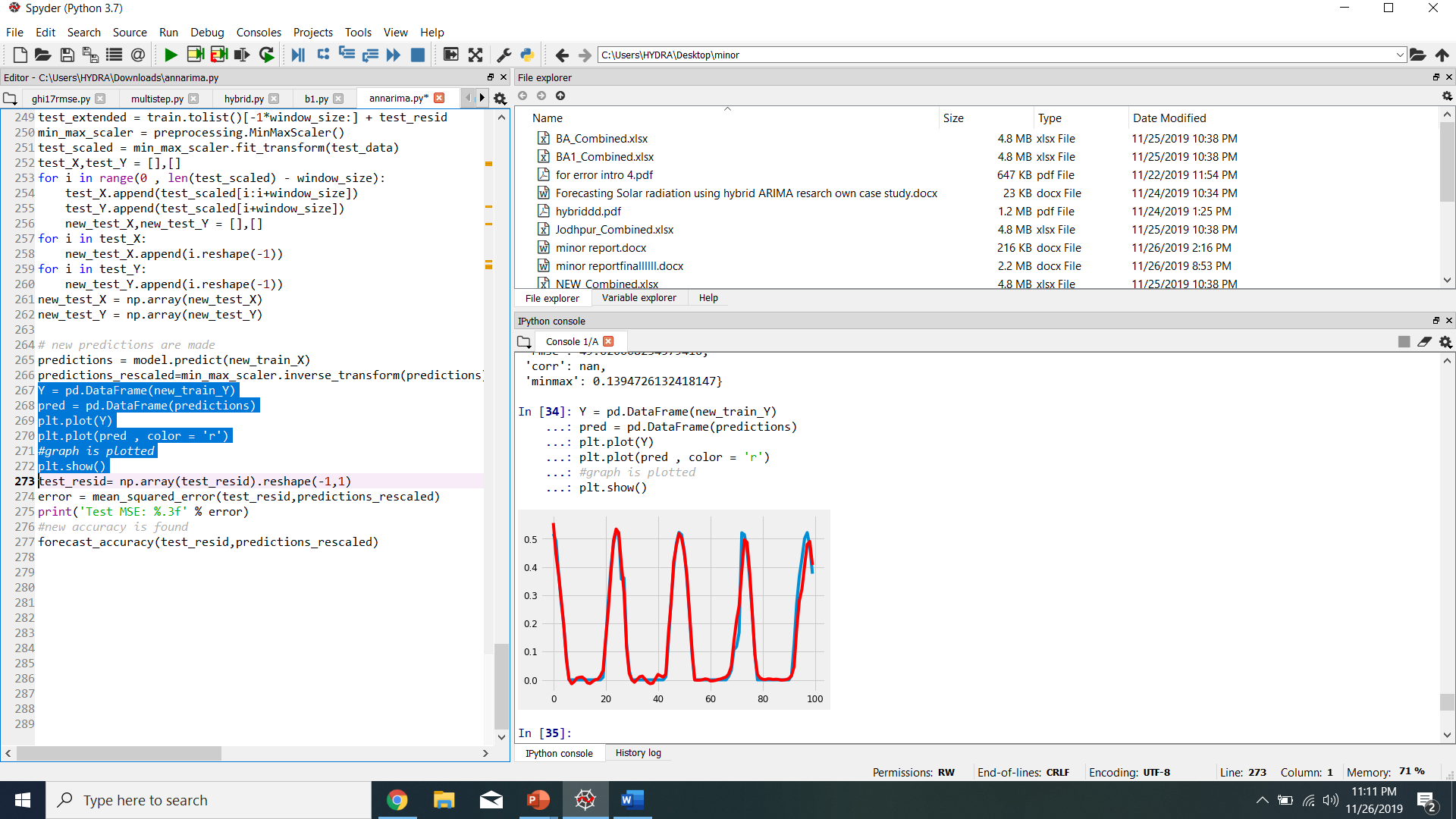


Fig 5: JAISELMER HYBRID(TEST RESIDUAL – PREDICTIONS RESCALED)

Fig6:Barmer hybrid (test residuals and predictions rescaled)

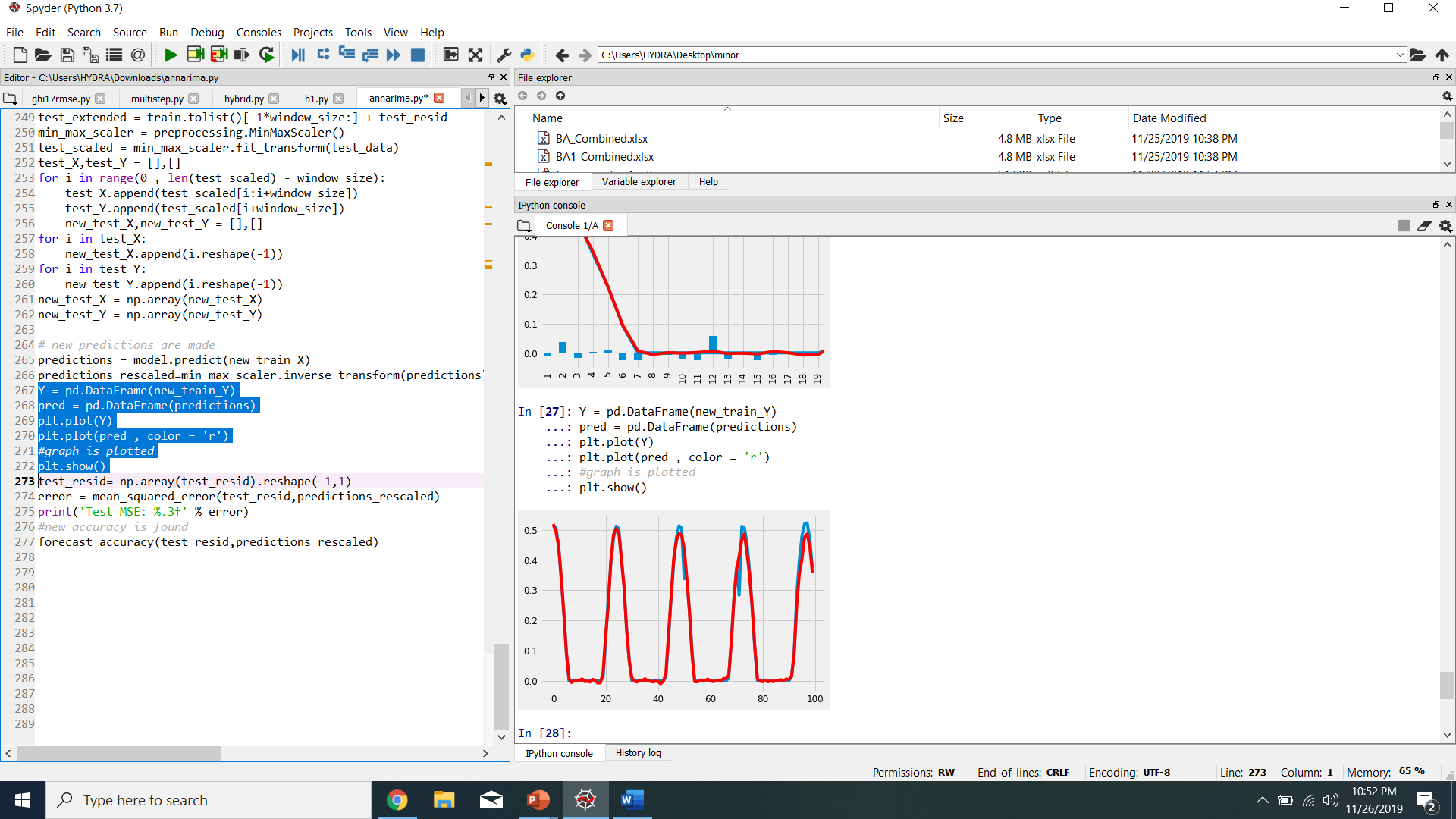


Fig 7: JODHPUR HYBRID (TEST RESIDUALS AND PREDICTIONS RESCALED)

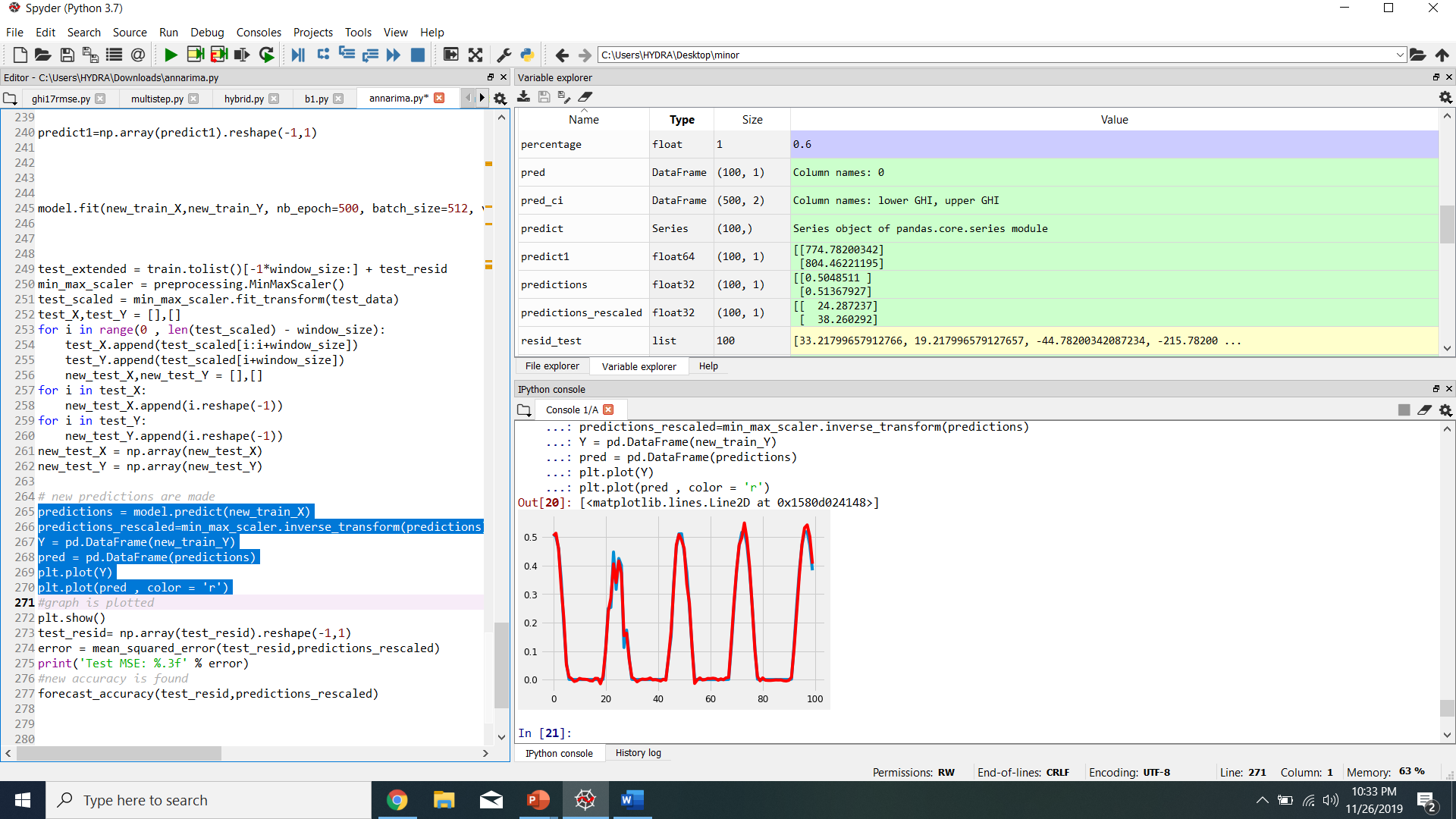
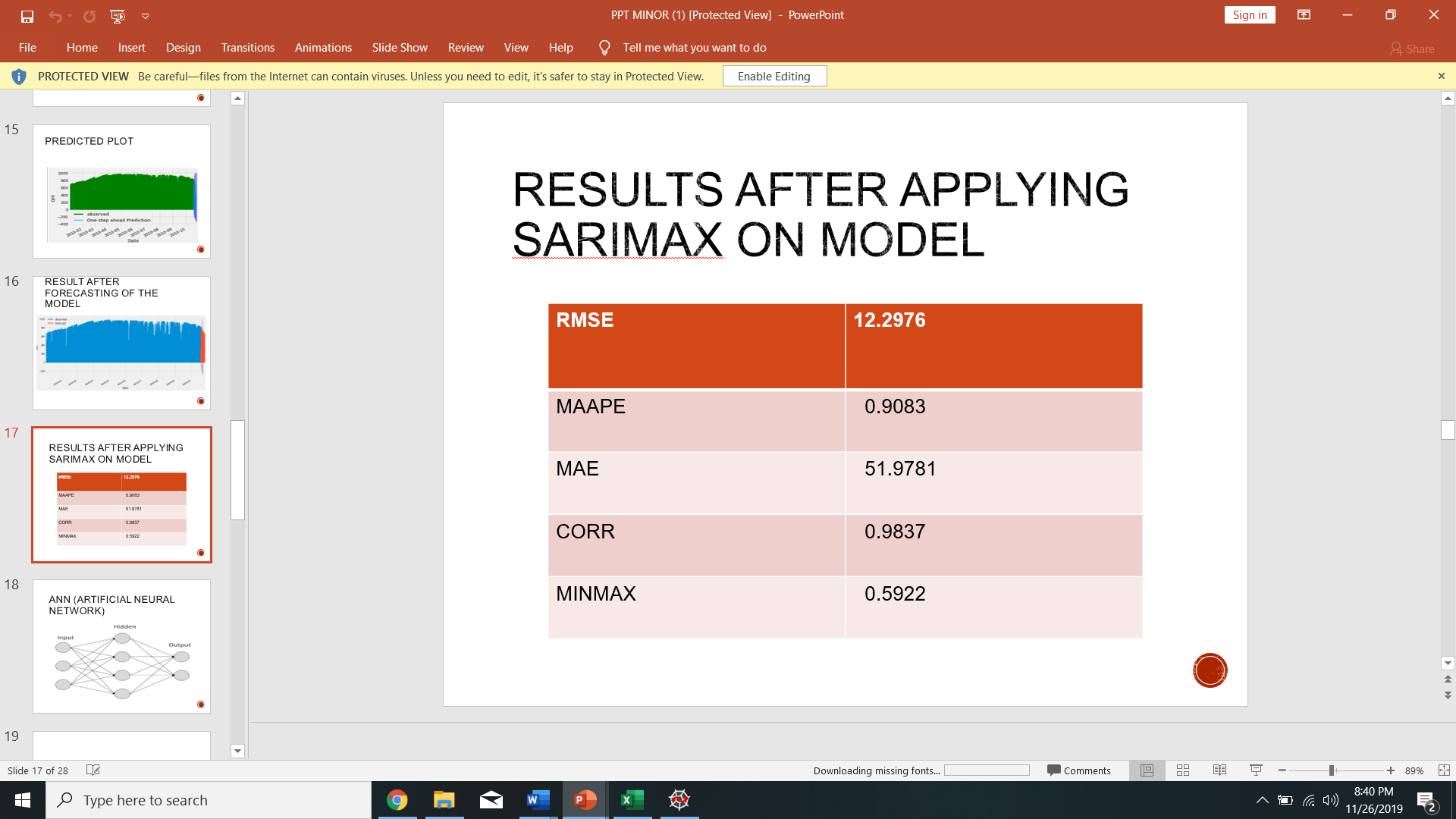
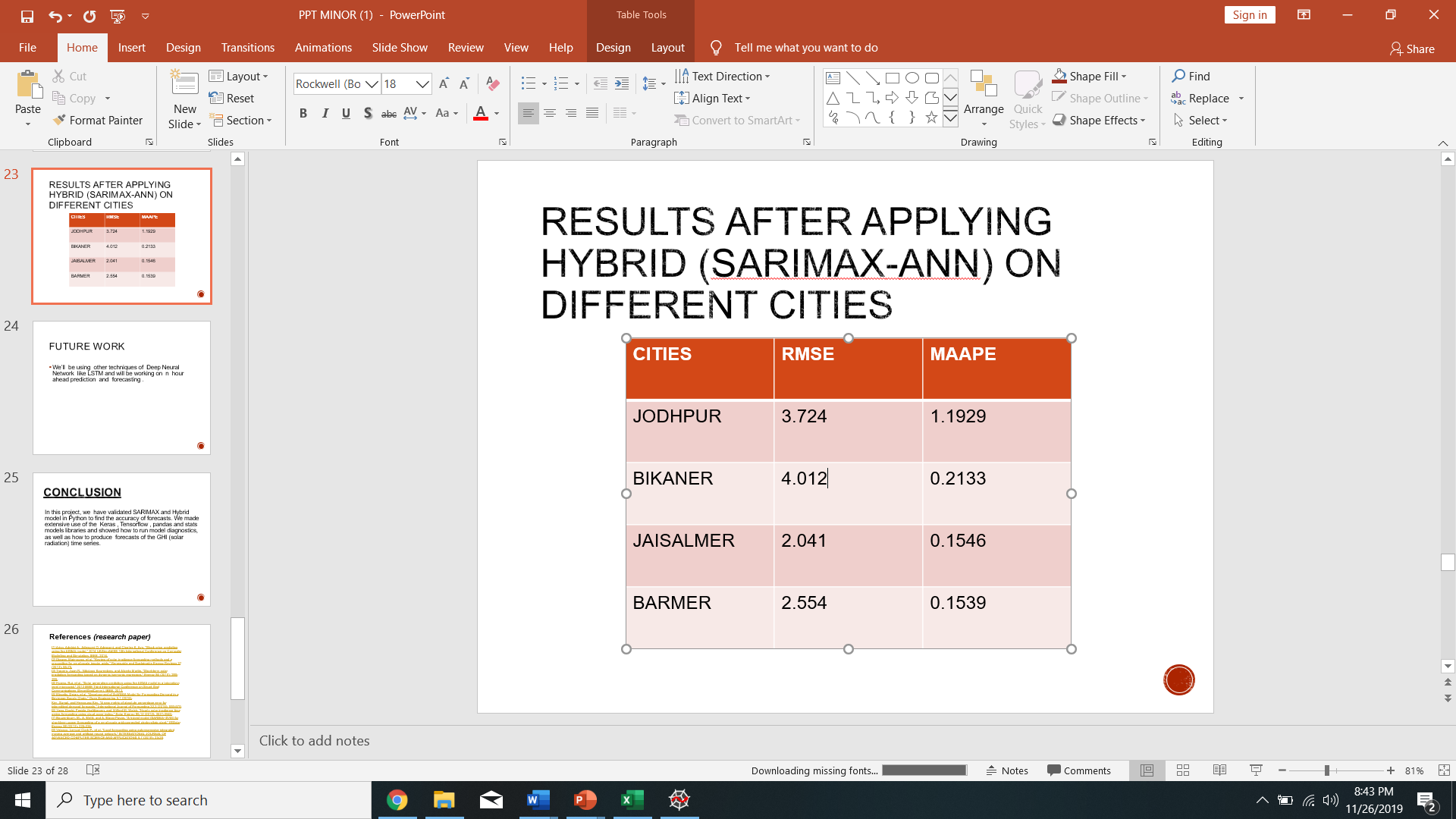


Fig 8: Bikaner hybrid (test residuals and predictions rescaled)

The mean absolute arctangent percentage error (MAAPE)and root mean square error (RMSE) are used evaluate the accuracy of forecasting.





**Chapter 6**

**Conclusion and Future Scope**

**­­­­­­­­­­­­­­­­­­­­­­­­­­­­**

**6.1 Conclusion**

In this project, we have validated SARIMAX and hybrid SARIMAX-ANN model in Python to find the accuracy of forecasts.

Also we were able to produce predictions of next 100 Global Horizontal Index(GHI) values.

**6.2 Future Scope**

Following things can be done:

* Use deep learning methods like RNN and CNN
* To predict multi hour ahead rather than one hour ahead only
* To train and test on different datasets

**References**

[1] Khatib, Tamer, Azah Mohamed, and Kamaruzzaman Sopian. "A review of solar energy modeling techniques." *Renewable and Sustainable Energy Reviews* 16.5 (2012): 2864-2869.

[2] Inman, Rich H., Hugo TC Pedro, and Carlos FM Coimbra. "Solar forecasting methods for renewable energy integration." *Progress in energy and combustion science* 39.6 (2013): 535-576.

[3] Sözen, Adnan, Erol Arcaklioğlu, and Mehmet Özalp. "Estimation of solar potential in Turkey by artificial neural networks using meteorological and geographical data." *Energy Conversion and Management* 45.18-19 (2004): 3033-3052.

[4] Voyant, Cyril, et al. "Optimization of an artificial neural network dedicated to the multivariate forecasting of daily global radiation." *Energy* 36.1 (2011): 348-359.

[5] Mellit, Adel, and Soteris A. Kalogirou. "Artificial intelligence techniques for photovoltaic applications: A review." *Progress in energy and combustion science* 34.5 (2008): 574-632.

[6] Mubiru, J. "Predicting total solar irradiation values using artificial neural networks." *Renewable Energy* 33.10 (2008): 2329-2332.

[7] Salcedo-Sanz, S., et al. "Daily global solar radiation prediction based on a hybrid Coral Reefs Optimization–Extreme Learning Machine approach." *Solar Energy* 105 (2014): 91-98.

[8] Ahmed, Mohammed S., and Allen R. Cook. *Analysis of freeway traffic time-series data by using Box-Jenkins techniques*. No. 722. 1979.

[9Nihan, Nancy L., and Kjell O. Holmesland. "Use of the Box and Jenkins time series technique in traffic forecasting." *Transportation* 9.2 (1980): 125-143.

[10] Williams, Billy M., and Lester A. Hoel. "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results." *Journal of transportation engineering* 129.6 (2003): 664-672.

[11] Zhang, G. Peter. "Time series forecasting using a hybrid ARIMA and neural network model." *Neurocomputing* 50 (2003): 159-175.

[12] Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, *50*, 159-175.

[13] Wei, Liyang, et al. "A study on several machine-learning methods for classification of malignant and benign clustered microcalcifications." *IEEE transactions on medical imaging* 24.3 (2005): 371-380.

[14] Yadav, Amit Kumar, Hasmat Malik, and S. S. Chandel. "Selection of most relevant input parameters using WEKA for artificial neural network based solar radiation prediction models." *Renewable and Sustainable Energy Reviews* 31 (2014): 509-519.

[15] Zhang, G. Peter, ed. *Neural networks in business forecasting*. IGI global, 2004.

[16] Khashei, Mehdi, and Hehdi Bijari. "Exchange rate forecasting better with hybrid artificial neural networks models." *Journal of Mathematical and Computational Science* 1.1 (2012): 103-125.

[17] Khashei, Mehdi, and Mehdi Bijari. "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting." *Applied Soft Computing* 11.2 (2011): 2664-2675.

[18] Sheela, K. Gnana, and Subramaniam N. Deepa. "Review on methods to fix number of hidden neurons in neural networks." *Mathematical Problems in Engineering* 2013 (2013).

[19] Kim, Sungil, and Heeyoung Kim. "A new metric of absolute percentage error for intermittent demand forecasts." *International Journal of Forecasting* 32.3 (2016): 669-679.

**Appendix**

