Abstract

This paper describes the analysis of solar forcing parameters in the ionosphere and proposes a software solution to analyses and plot different parameters associated with the phenomenon.

List of Figures

List of Tables

Contents

Introduction

A large number of studies are carried out in the ionospheric region of the atmosphere in order to study about solar activities. The solar activity is the physical phenomena that occurs at solar magnetic field which fluctuates on time.

What are solar forcing parameters

Why solar forcing parameters are important

What is correlation

How solar forcing parameters are correlated

Where all we can use correlation between solar forcing parameters

What are the uses of correlation between solar forcing parameters

* Motivation
* Problem Statement
* Goal of the thesis

Theory

* Ionosphere
* Correlation of solar forcing parameters
  + TEC
  + Soalar radio flux
  + Solar wind speed
* Requirements of analysis software
* Existing Tools
* Data
  + Datasets
  + preprocessing

Software Development

Modern software tools are required to analyze large datasets. The data set can be raw data or unstructured data, which has to be formatted according to the requirements of the analysis. Data scientists extract information from these data sets for data driven decision-making. The accuracy of data is critical because inaccurate data leads to faulty conclusions. Frequent use of the same analytics for multiple scientific researches brings the requirement of a software tool to save time and cost of the research.

A good software product is developed through certain steps and guidelines, which are called software development life cycle (SDLC). SDLC is a used by software industries to design, develop, test and maintain quality software products. (Yadav, 2015) The software to analyze scientific data has been developed by the popular SDLC method called agile software development methodology. It is a modern software development methodology having flexibility and incorporates levels of practicality into the final product. Agile methodology focuses on keeping the code simple and delivering functional parts of the application as it finishes. Agile development is a kind of incremental software development with rapid development cycles. (Yadav, 2015)

SDLC – software development life cycle steps

* Developing a tool to analyze scientific data
  + Design

The analyzing process is divided into 4 independent steps, which are setting up the project and loading the data, data cleaning and formatting, computations, visualization. Each step in the process flow has its own data configurations to handle different data sets and different operations. The configurations files are written in JSON format and saved as json file, which makes it easier to handle in the software. There will be independent folder locations for each process step and these folders contains the configuration files.

* + - Workflow
      * Setting up the project and loading the data

Workflow start with setting up the project. Here in this step a project folder is created in project location with name of the project. All the intermediate files and final results are saved into this folder during the process workflow. Project name and project location is saved into the configuration file for easy reference during the program execution. The raw data for the analysis is available in different file formats like comma-separated values (CSV), text files and other formats that may make the read process slower. To make the data handling and computations faster, data files are read and converted to a binary file format called hierarchical data format (HDF) in this process step.

* + - * Data cleaning and formatting

Accuracy of the analysis performed is highly depended on the quality of data. Raw data consist of data from different sources, duplicates, corrupted data or incorrect data. In this step raw data files are read from the specified location and making it ready for operations. It consist of ensuring the time series contains number of data points, merging data columns from different data sources and interpolating data points if needed.

* + - * + Read data files
        + File formats(hdf)
        + Save formatted files
      * Computations
        + Mean
        + interpolation
        + Relative difference
        + Correlation with box window
        + Correlation with gaussian window
        + Confidence interval
      * Visualization
        + line graphs
        + colour maps
        + interactive plots
    - UI
      * Design
      * Libraries
  + Build
    - Python
    - Dependencies
      * Numpy
      * Pandas
      * Pycdf
      * matplotlib
    - File Formats
      * Cdf
      * Hdf
      * Csv
      * Txt
      * json
    - Metadata configurations
  + Test
  + Deployment
    - Deployment in cloud machine
  + Maintenance
    - repository

Results

Conclusion

Reference

Appendix