Pattern Recognition of Space Plasma Regimes

A Thesis

Presented to

the Faculty of the Department of Computer Science

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

By

Pengfei Guan

May 2013

Pattern Recognition of Space Plasma Regimes

Pengfei Guan

APPROVED:

Dr. Ricardo Vilalta, Chairman
Dept. of Computer Science

Dr. Shishir Shah
Dept. of Computer Science

Dr. Saurabh Prasad
Dept. of Electrical & Computer Engineering

Dean, College of Natural Sciences and Mathematics

Acknowledgements

I would like to express my sincere gratitude to my advisor Dr. Vilalta for his extraordinary support and motivation. Not only did he help me in improving my skills as a student and a researcher, but also as a professional. I would like to thank him for all his valuable advice and the precious time that he spent on me. I would also like to thank Dr. Shah, and Dr. Prasad for being willing to serve on my committee. Their valuable suggestions before and after my defense helped improve my thesis enormously.

I would like to thank my lab mates – Dainis Boumber, Kinjal Dhar Gupta, Fransisco Hernandez, Son Hoang, Valerio Roberto, and Bangsheng Sui. I had a fantastic time with them and I am looking forward to a reunion in the future.

My special thanks goes to my family, and especially to my mother for all the sacrifice she made for me. I would also like to thank all my friends for their constant encouragement and valuable suggestions. I am fortunate to have all them in my life.

In memory of my Father, Yongfu Guan, and my Friend, Lun Guo.

Pattern Recognition of Space Plasma Regimes

An Abstract of a Thesis

Presented to

the Faculty of the Department of Computer Science

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

By

Pengfei Guan

May 2013

Abstract

Space plasma is made of electrically charged gases or fluids in space that are made up of free electrons and ions. They are studied extensively not only to analyze the dynamic processes of stellar bodies but also to understand various phenomena including particle acceleration, wave-particle interaction, applied science of space weather, and its impact on human technology. The identification of primary particles of plasma is of utmost importance for these kinds of research. There is considerable amount of data available; however, deriving a formula or methods for manual plasma regime identification is extremely time consuming, and can be highly unreliable and lack robustness. An automatic process of classifying these primary particles is of high demand. Currently, existing techniques that use machine learning algorithms have difficulty in distinguishing perceptible boundaries and regions as good as the human eye. In contrast, we propose a classification method to identify plasma particles automatically given a highly diversified time series data, based on energy and pitch angle. We came up with this algorithm after exploiting various learning techniques on the entire available data. Experiments are reported on datasets obtained from the Fast Auroral SnapshoT (FAST) explorer, which is the second mission in NASA's Small Explorer Satellite Program (SMEX).

Contents

List of Figures

List of Tables

Chapter 1 INTRODUCTION

1.1 Background

1.1.1 Machine Learning

Machine Learning, a branch of Artificial Intelligence, aims to build computer systems that can adapt and learn from experience. A machine learning system is normally trained on datasets to learn "relevant knowledge", and then such knowledge is used to complete new tasks by testing on new datasets. These train and testing steps are indispensable to formulate machine learning systems.

There are several reasons that allow machine learning to become important in helping people solve problems. Sometimes, the input/output pairs do not indicate a concise relationship that can be discovered via an implicit suitable relationship with efficient and effective algorithms running on the datasets. The important relationship and correlations are probably hidden in large amount of data and they can be extracted effectively by machine learning methods. Some tasks only works well in specific environments. Certain characteristics of the working environment, however, are unknown to the designer at the beginning. Machine learning methods can adjust and improve current designs as new information arrives. Moreover, machines capture more knowledge as needed than human can. In fact, although we can hardly enumerate all the possible reasons here, machine learning shines by its incomparable advantages.

1.1.2 Space Plasma Regimes

In Space Science, space plasma is an electrically charged gas or fluid made up of free electrons and ions. It is often dubbed "the fourth state of matter", and may be the most

common state of matter in the universe that carries an incredible wealth of information in different areas. One key area of space plasma physics is the study of space weather ([01]), defined as the study of natural process in space that can affect Earth, the near-Earth environment, satellites, and space travel. One example of such processes is geomagnetic storms caused by solar coronal events.

Experimental plasma physics relies on the simultaneous measurement of multiple parameters inside plasma. Example parameters include plasma density and temperature, electromagnetic field power, particle velocities, and more. For a review of space plasma measurement approaches see ([02][03][04]).

The Fast Auroral SnapshoT (FAST) was launched in 1996 into a near-polar, highly elliptical orbit (350 km * 4,200 km) to study physical processes in Earth's auroral zones. Its explorer passed through the nightside auroral oval in the 22 to 24 MLT rang on orbits 1740-1779. We analyze the data collected from these orbits for our statistical survey.

Figure 1-1 shows a pass of the FAST through the northern auroral zone of Earth's ionosphere, there is a comprehensive set of particles detected the space explorer. Three primary particles were classified from a coarse, by-hand work demonstrated as Downward currents (green bar), Upward currents (purple bar), and Alfvénic currents (red bar) in the figure. The key physics and characteristics of the three auroral regions are displayed in **Figure 1-2**.

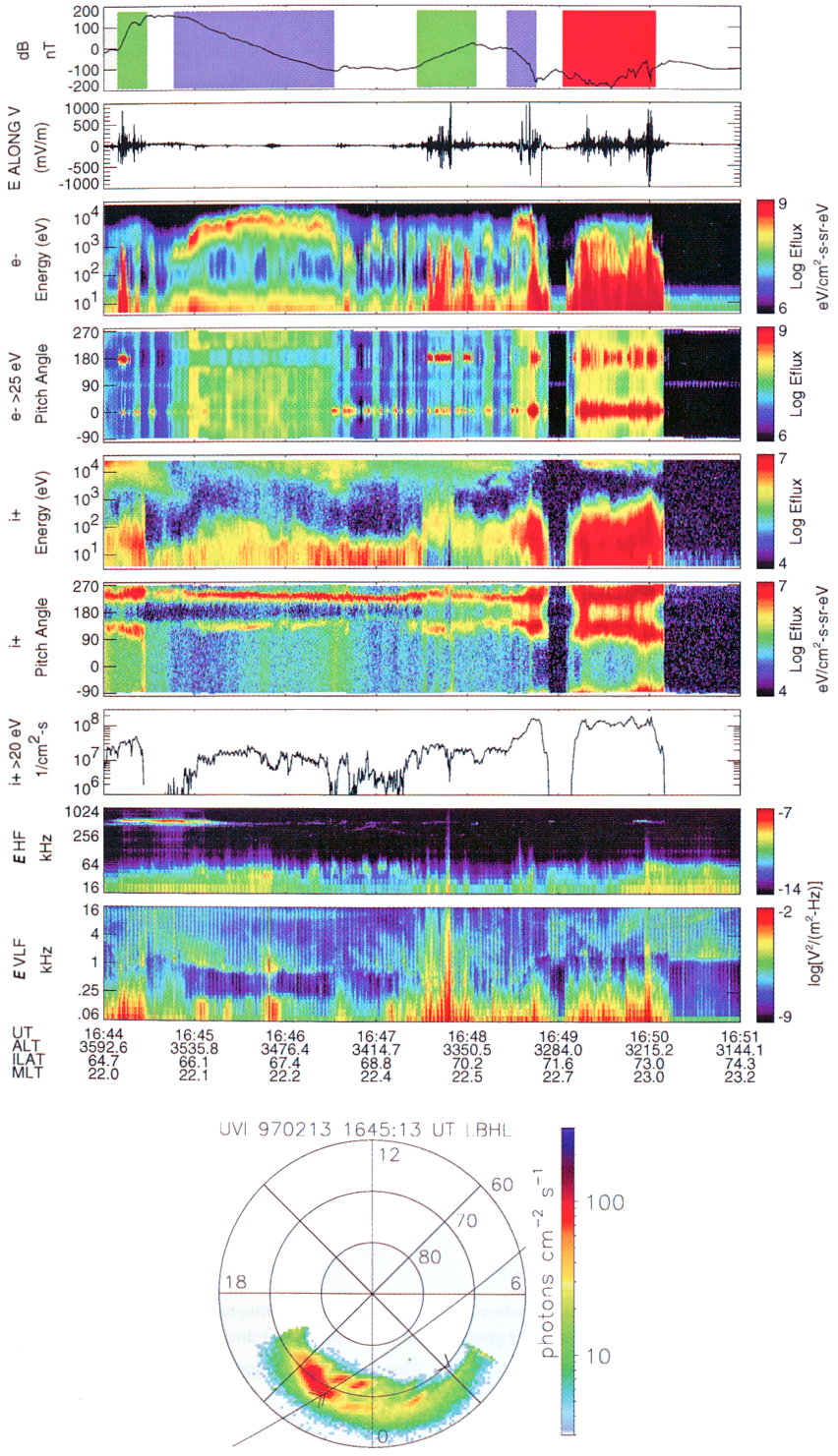

Figure 1-1: FAST satellite pass through all three primary auroral regions of downward (green), upward (blue) and Alfvénic currents (red). The FAST orbit is plotted into a global UV auroral image taken from POLAR (bottom). From [05].

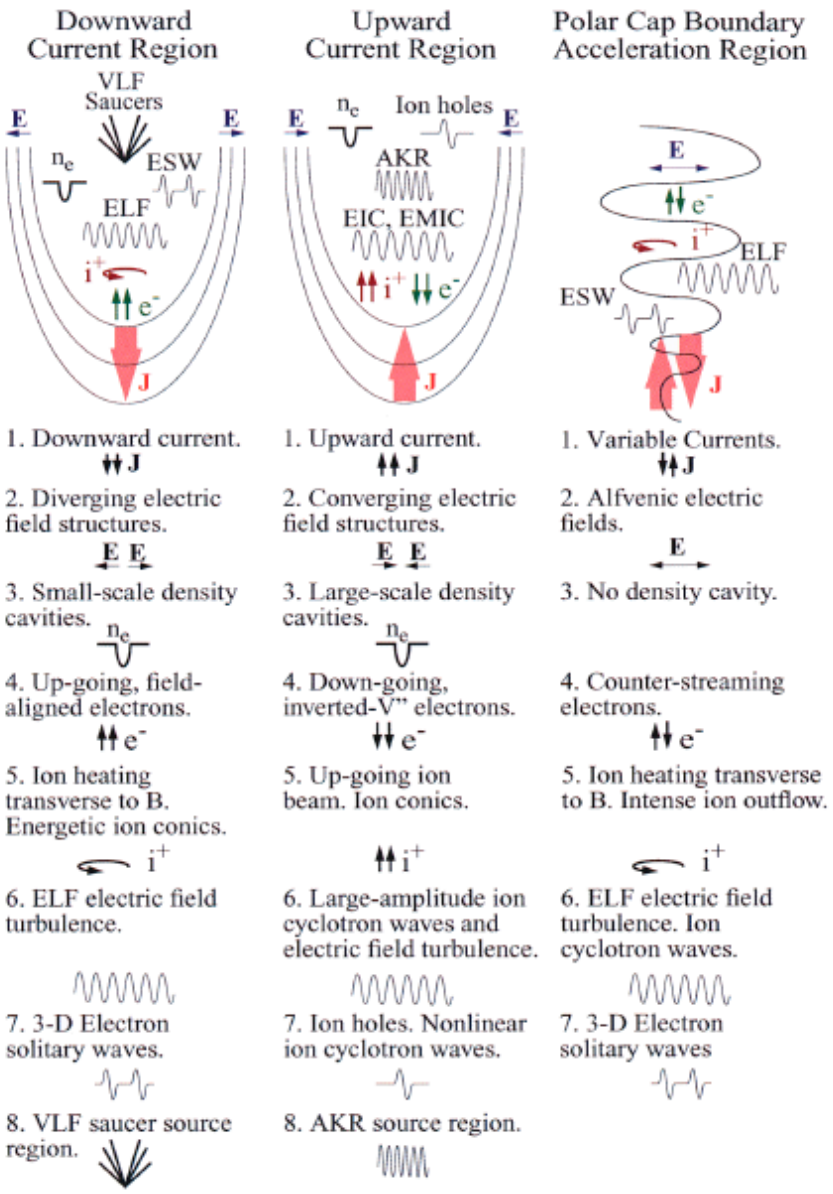

Figure 1-2: Summary diagram that describes the key physics and characteristics of the three auroral regions, ordering auroral observations according to electric current. This diagram contains both particle and waves/fields phenomena. From [06].

1.2 Objective

The main goal of this project is to develop a machine learning system for the automatic detection of particles in space science using a set of pattern recognition tools, including machine learning, data mining, and image processing. The system can focus on measurements in variable datasets, which are relevant to space weather research (e.g., such as predicting geomagnetic storm caused particle precipitation and ion outflow events), and on fundamental plasma physics.

Chapter 2 Related Work

There is only a few of studies on the automated classification and identification of space plasma data. Ball [07] displayed an example of a modern survey for which the methods of astroinformatics are ideally suited. Strangeway, et al. [08] found the evidence of very low plasma density by observing VLF waves in the auroral zone. And Newell, et al. [09] proposed a methodology to map the ionosphere to the magnetosphere.

Several standard computational approaches were involved. The main focus points are statistical studies of events to extract average behavior to understand the individual response of the system. In either case, event selection is mostly done by hand (visual inspection).

2.1 Statistical Analysis

The author in [10] investigated the behavior of 10 different characterizations of the magnetosphere and analyzed the magnetospheric state variables from different technical perspectives and introduced the ensemble learning approach to work on the datasets.

In order to find the comparisons of electrons and ions, the author in [11] developed a statistical model of auroral ion precipitation. By using this model, the auroral precipitating ions can be displayed in a well-ordered pattern in magnetic local time and geomagnetic latitude.

Badman, et al. [12] analyzed a sample of 22 Hubble Space Telescope images of Saturn's southern auroral oval to statistically determine the average location and width of the aurora, and their variability.

2.2 Bayesian Methods

In [13] the author described the reconstruction algorithm based on Bayesian method and developed a concrete algorithm for the Generalized-Auroral Computed Tomography.

Lointier, et al. [14] proposed a Bayesian classifier for studying the Solar Wind-Magnetosphere-Ionosphere system by identifying and tracking the projection of magnetospheric regions on the high-latitude ionosphere.

2.3 Correlation Analysis

The author in [15][16] proved the correlation of solar wind with oxygen content of ion and plasmapause position separately; [17] discussed the importance of O^+ ions across the polar cap as the event studies.

It is very difficult to develop techniques that can distinguish discernible boundaries (and regions) as well as, or better than human eye. Some approaches may produce satisfactory results on certain kinds of data; however, no general technique has been proposed. For instance, Newell, et al. [18] developed an algorithm using Defense Meteorological Satellite Program (DMSP) data, covering the period from 1984 to about 1990, which is highly uniform with rare data gaps. This algorithm was made for specific and consistent identifications only. In contrast, our proposed work plans to develop tools that can automatically identify and classify plasma environments of diverse data types; machine learning techniques are fully exploited to establish an efficient collaborative framework.

Chapter 3 Methodology

3.1 Feature Extraction

We plan to extract the relevant features from the spectrograms that can produce a function mapping our data sets as input to desired, labeled classes as output. **Figure 3-1** shows an example of two spectrograms taken from an auroral pass captured by the FAST satellite and POLAR spacecraft. Time is on X-axis, angle Y-axis, and energy value of the detected particle Z-axis (different color refers to different energy values). For instance, point (x, y, z) represents the particle detected by the sensor at the time x with an entering angle y and there are particles of number z detected.

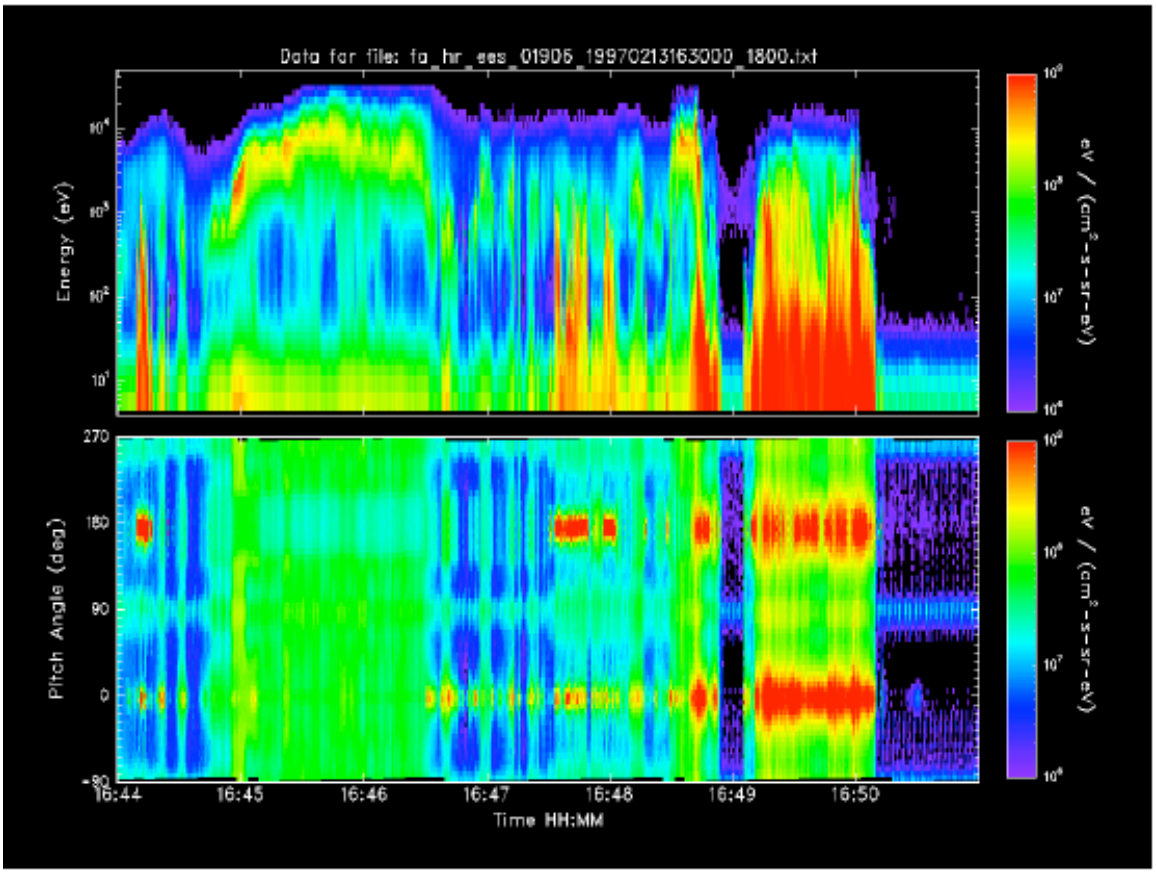

Figure 3-1: An example of electron spectrograms versus energy and pitch angle. Different colors refer to different frequencies.

3.1.1 Bin Packages

To reduce the size of features, we take advantage of bin packages. For every instance of time, we have W angles in total with a corresponding energy value e on each. We divide W into L bins of equal size W/L. In each bin, we add up all the energy values corresponding to the angles belonging to that bin to get one value B. By doing this, our feature's dimensionalities have been decreased by W/L times without losing much relevant information. Repeat the process along the entire time instance T. The equation (1) follows below:

$$\{B_j^t = \sum_{i=1}^{\lceil W/L \rceil} e_{i+\lceil W/L \rceil * (j-1)} \mid 1 \le j \le \lceil W/L \rceil, 1 \le t \le T\} \qquad (1)$$

3.1.2 Discrete Wavelet Transform

Wavelet Transform (WT) as a technique particularly useful for analyzing signals, and was mainly used to attack the problem related to frequency and time resolution properties. Compared to Fourier Transform, WT contains additional special properties of the wavelets, which show up at the resolution in time at higher analysis frequencies of the basis function that we can extract relevant information in a more effective way. Considering that our dataset is discretely sampled, we take Discrete Wavelet Transform (DWT) as our analysis techniques.

DWT is computed by successive lowpass and highpass filtering of the discrete time-domain signal, providing high time resolution and low frequency resolution for high frequencies and high frequency resolution and low time resolution for low frequencies. Thus, DWT is able to provide a compact representation of a signal in time and frequency with efficient computation [19]. There are different types of mother wavelets. Because

the Haar wavelet can give us the best resolution power over time/space domain, we chose it as our mother wavelet and performed decomposition at level 3 of signal. The DWT is defined by the following equation:

$$W(j,k) = \sum_{j}\sum_{k} x(k)2^{-j/2}\varphi(2^{-j}n - k) \qquad (2)$$

where φ (t) is a time function with finite energy and fast decay called the mother wavelet. Also, **Figure 3-2** displays how the signal is decoded by the DWT, where signal is X[n], low pass filter is G0, and the high pass filter is denoted by H0. The high pass filter passes filter produces detail information, while the low pass filter associated with scaling function produces coarse approximations.

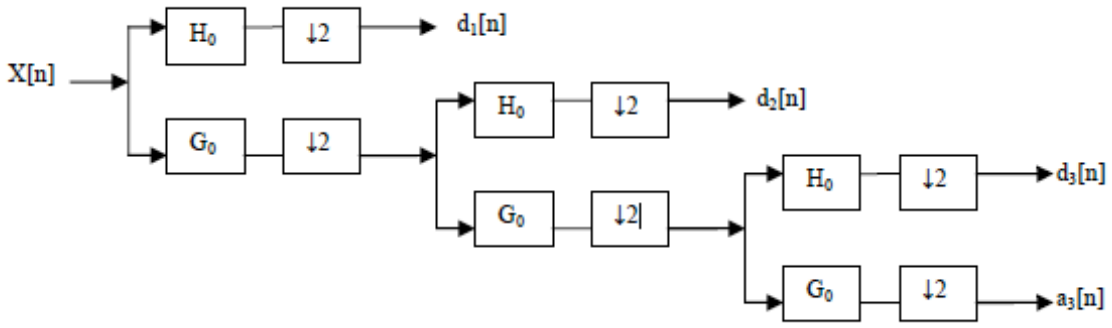

Figure 3-2: Three-level wavelet decomposition tree.

I. We take a window of size S time stamps, where S must be equal to 2^n (n is a positive integer). Put together all the j^{th} bin of each time stamp covered by the window where time t is the center, we obtain the vector V_j^t (3). After that, we calculate the DWT of V_j^t to get vector $V_j^{t'}$ (4) introduced below, and achieve V_t'' by merging all the $V_j^{t'}$ together (5). Thus, V_t'', as DWT-time, is the feature vector that represents the instance of time t. Move the window by one time stamp where

time (t+1) becomes the center of the window, and repeat the process as above till the end of time sequence.

$$\{V_j^t = (B_j^{(T_{t-(S-1)/2})}, B_j^{(T_{t-(S-1)/2+1})}, ..., B_j^{(T_{t+(S-1)/2})}) \mid 1 \le j \le L, S \le t \le T - S + 1\} \quad (3)$$

$$\{V_j^{t'} = DWT(V_j^t) \mid 1 \le j \le L, S \le t \le T - S + 1\} \quad (4)$$

$$\{V_t'' = (V_1^{t'}, V_2^{t'}, ..., V_L^{t'}) \mid S \le t \le T - S + 1\} \quad (5)$$

II. In addition, we also study the DWT on energy stamps. We calculate the DWT on all the bins at each time point. Say there are L bins, where L must be equal to 2^n (n is an integer). At time point t, we get the features vector V_t' of DWT-energy described as (6), (7).

$$\{V_t = (B_t^1, B_t^2, ..., B_t^L) \mid 1 \le t \le T\} \quad (6)$$

$$\{V_t' = DWT(V_t) \mid 1 \le t \le T\} \quad (7)$$

III. For each instance, we get the final vector of features V_t''' by merging V_t'' in (5) and V_t' in (7) together stated in (8). S is the size of time stamp window described in I.

$$\{V_t''' = (V_t'', V_t') \mid S \le t \le T - S + 1\} \quad (8)$$

3.1.3 Features Ranking

After 3.1.2, for each instance, the size of final vector of features is (S * L + L). From the perspectives of Information Gain, Information Gain Ratio, Chi-squared, and Relief, we choose the top i features, which rank the highest on these four factors.

I. Information Gain

Information gain is the change of the entropy from a prior state to present one. The equation (9) is displayed as below:

$$IG(T,a) = H(T) - H(T|a) \qquad (9)$$

where H denotes the entropy. Because we want to find the most relevant features, and thus the one with high information gain should be preferred to others.

II. Information Gain Ratio

Information gain ratio can be used to improve on a limitation of information gain. It is just the ratio between the information gain and the intrinsic value (10).

$$IGR(Ex,a) = IG/IV \qquad (10)$$

III. Chi-squared

It is one of the most widely used probability distribution in inferential statistics. If X_1, …, Z_k are independent, standard normal random variables, then the sum of their squares, $Q = \sum_{i=1}^{k} X_i^2$, is distributed according to the chi-squared distribution with k degrees of freedom. It tells us that if the dataset follows certain kind of probability distribution.

IV. Relief

Relief is a feature selection method ([20]) based on attribute estimation. It assigns a grade of relevance to each feature, and those features with a higher value than a given threshold will be preferred. The general algorithm of relief is followed in (11) and (12):

$$W[f] = W[f] - diff(f,E_1,H) + \sum_{C \neq class(E_1)} P(C) * diff(f,E_1,M(C)) \qquad (11)$$

$$diff(f,E_1,E_2) = \begin{cases} 0 & if\ value(f,E_1) = value(f,E_2) \\ 1 & otherwise \end{cases} \left| \frac{|value(f,E_1) - value(f,E_2)|}{\max(f) - \min(f)} \right| \qquad (12)$$

Relief-F finds one nearest neighbor of E1 from every class. For these neighbors, Relief evaluates the relevance of every feature f ∈ F accumulating it into W[f]. H represents the nearest neighbor from the same class, while M(C) of class C stands for from different class. Ultimately, W[f] is divided by m to get the average evaluation. Relief is able to capture feature interactions; this is important when the relevance of a feature is hidden in the interaction with other features.

3.1.4 Principal Component Analysis

Principal component analysis (PCA) is a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrected variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible ([21]). PCA can help us reducing the number of dimensionalities of our feature set without loss much of information. We extract the features from the previous i features in 3.1.3 by taking the advantage of PCA. Taking the first j numbers ($j < i$), which occupies the most of the relevant information. Consequently, our new feature sets is the first j component.

3.2 Classifier Selection

3.2.1 Support Vector Machines

Support Vector Machines (SVM) preprocesses the data by representing all examples in a sufficient higher dimensional space where the classes can be separated by a hyperplane. We try to find the separating hyperplane with the “largest” margin, which is the distance between the hyperplane and the closet example to it. These examples are called support vectors. The equation bellows uses a discriminant function:

$$g(x) = \sum_{i=1}^{N} a_i K(x_i, x) \qquad (10)$$

where {ai} is a set of real parameters, index i runs along the number of training examples, and K is a kernel of degree one in polynomial.

3.2.2 Neural Networks

Neural networks simulate the workings of the brain. A network is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. These elements are divided into layers with weighted directional connections. Inputs are passed from a base layers through the network one layer at a time. Each neuron produces an output based on the weighted inputs it receives and a processing function (a sigmoid function often). After having gone through all the layers, the weights are adjusted according to the comparison between the final output and the target output. Here is a simple equation:

$$g_k(x) = f(\sum_j w_{kj} f(\sum_i w_{ji} a_i + w_{j0}) + w_{k0}) \qquad (11)$$

where x is the input parameter vector, f(.) is a nonlinear (i.e., sigmoid) function, and ai is a component of vector x. Index i runs along the components of vector x while index j goes with a number of intermediate functions. K refers to the k^{th} output neuron.

3.2.3 Naïve Bayes

The Naïve Bayes classifier, derived from Bayes Theorem, is a parametric technique. We use it to identify the maximum posterior probability of a class given the input vector x, P(Yi|x). This can be represented mathematically as:

$$P(Y = y_i \mid X = x_k) = \frac{P(X = x_k \mid Y = y_i) P(Y = y_i)}{\sum_j P(X = x_j \mid Y = y_j) P(Y = y_j)} \qquad (12)$$

3.2.4 Boosting – Decision Tree

The Decision Tree classifier is a non-parametric technique, approximating a target concept using a tree representation, where each internal node corresponds to an attribute, and every terminal node corresponds to a class. This technique recursively divides the feature space using boundaries orthogonal to the feature axes based on the value of a selected attribute.

Boosting is a popular ensemble classifier for reducing variance and bias components of error. It combines those "weaker" learners iteratively with respect to a distribution, and formulating a strong learner finally. Each learner tries to correct the mistakes made the previous learner, and the final prediction is decided by a weighted voting scheme. We plan to use Decision Tree classifier as the base algorithm with Boosting.

3.2.5 Random Forest

Random Forest is an ensemble classifier using many decision tree models described in 3.2.4. It can be used for classification or regression. Random Forest runtimes are fast, and they are able to deal with unbalanced and missing data. Random Forest weaknesses are that when used for regression they cannot predict beyond the range in the training data, and that they may overfitting data sets that is particularly noisy.

3.3 Data Modeling

3.3.1 Cross-validation

Cross-validation is a technique that helps a learning algorithm to validate its own trained model by choosing a part of its training set as validation set. In an n-fold cross-validation, the learning algorithm first divides the training set into n subsets. It uses (n-1) subsets together as the training set, and the remaining one set as the validation set. Once

the model is trained on the training set, it uses the validation set to assess its accuracy. The learning algorithm repeats the same process n number of times, each time taking a different subset among the n subsets as the validation set and uses the rest as the training set. After it has completed n runs, the final accuracy estimation is obtained by averaging over all the models it created in each of its run. We have used a 10 fold cross-validation method where n=10.

3.3.2 Weka

In order to achieve good training models, we test the datasets with multiple learning algorithms introduced in 3.2 using the tool Weka, which is a collection of machine learning algorithms for data mining tasks. From the model report produced by Weka, we normally take two factors into consideration including classification performance, root mean squared error to find the best-fit algorithm working with the updated features. The algorithms ([22][23][24]) will then be used to generate a predictive model. Therefore, the best training model is the combination of the new features and the suitable efficient algorithms.

Chapter 4 Experimental Results

4.1 Datasets

There are two images **Figure 4-1** and **Figure 4-2** displayed as below, both of them taken by FAST and POLAR spacecraft at the same time interval. **Figure 4-1** shows the EANG and EEV of the dataset, while **Figure 4-2** shows the IANG and IEV dataset. We assigned the Blank Area as Background Class 0; both of them contain three classes, Downward currents (green bar) Class 1, Upward currents (purple bar) Class 2, and Alfvénic currents (red bar) Class 3. Each sub-image contains 1326 time instances.

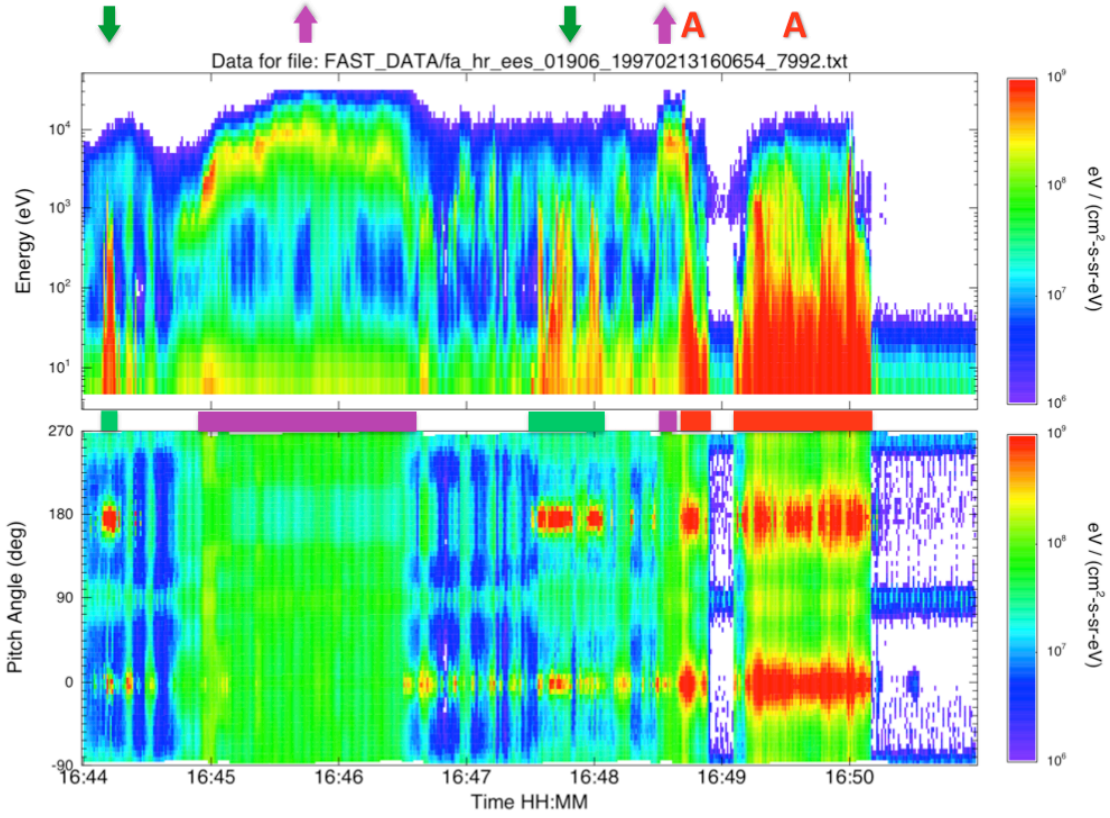

Figure 4-1: The two spectrograms (EANG/EEV) of Figure 3-1 with a by-hand and coarse labeling work. It shows the dataset taken from approximately 16:44 to 16:50.

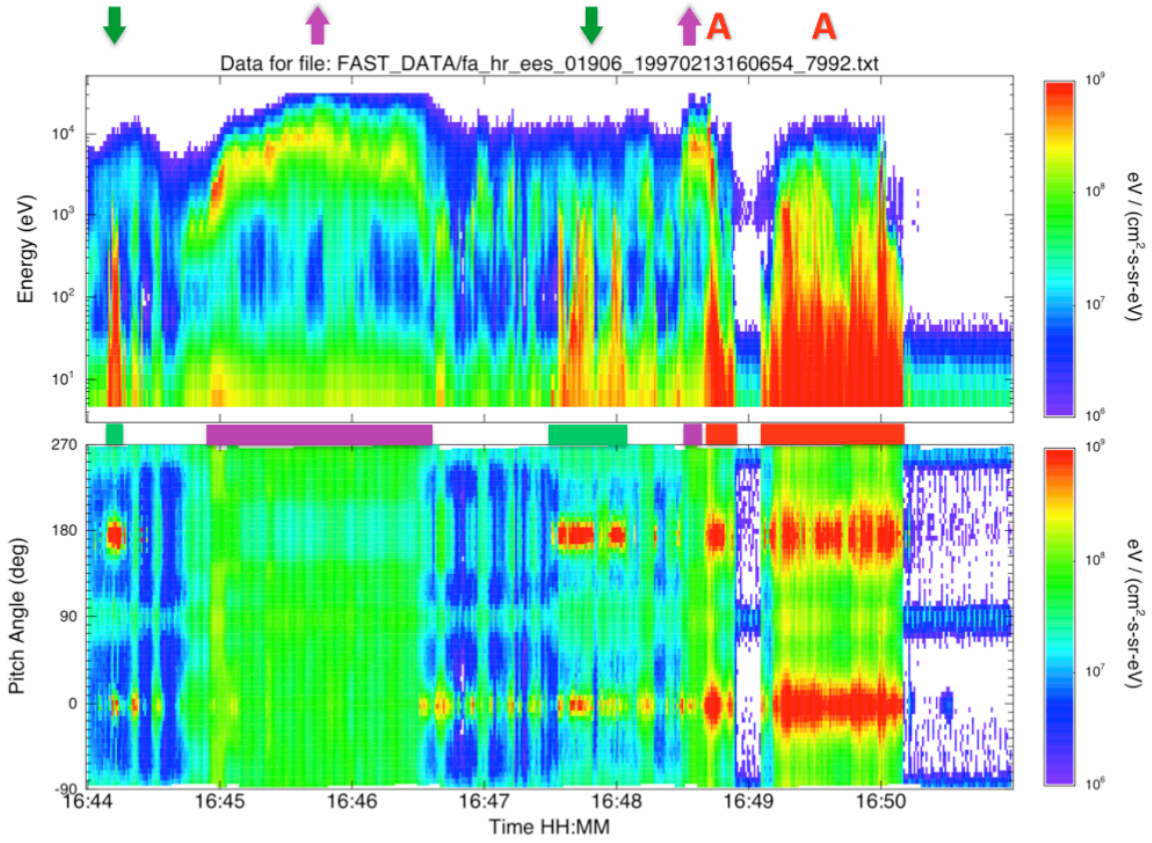

Figure 4-2: The two spectrograms (IANG/IEV) of Figure 3-1 with a by-hand and coarse labeling work. It shows the dataset taken from approximately 16:44 to 16:50.

4.2 Experiments

We mainly use two factors to decide whether a model is good or not. Accuracy is used to display the proportion of correctly classified instances while root mean squared error (RMSE) demonstrates the stability of the model. From all the classifiers we considered in 3.2, we decided to focus on Boosting Decision Tree (BDT) and Random Forest (RF), because these two classifiers gave us the best accuracy when compared to other algorithms working on our datasets. All the testing results are based on 10-fold cross-validation.

4.2.1 Phase 1

As for the Bin Packages, we tried to make two different sizes of bin sets: 4, 8. The classification results are shown in **Table 4-1**, **Table 4-2**, **Table 4-3**, and **Table 4-4**. From the result tables below, we finally decide to formulate 8 bins for each instance.

File 1-EANG	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	81.429%	0.2880	84.118%	0.2735
RF	81.681%	0.2542	83.193%	0.2351

Table 4-1: results of using 4 and 8 bins on File1-EANG.

File 1-EEV	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	87.647%	0.2384	93.193%	0.1806
RF	88.656%	0.2015	92.185%	0.1696

Table 4-2: results of using 4 and 8 bins on File1-EEV.

File 1-IANG	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	81.261%	0.2882	90.084%	0.2114
RF	81.513%	0.2607	89.160%	0.1952

Table 4-3: results of using 4 and 8 bins on File1-IANG.

File 1-IEV	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	90.336%	0.2117	90.840%	0.2028
RF	90.672%	0.1856	90.168%	0.1859

Table 4-4: results of using 4 and 8 bins on File1-IEV.

4.2.2 Phase 2

We use 8,16, and 32 respectively as the size of time stamps window for DWT and compared the performance under two classifiers shown in **Table 4-5**, **Table 4-6**, **Table 4-7**, and **Table 4-8**. Since models with 32-time stamps window have better performance, we selected 32 as our window size for DWT on time stamps.

File 1-EANG	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	92.984%	0.1798	95.234%	0.1458	97.066%	0.1173
RF	88.081%	0.2144	90.894%	0.1905	93.615%	0.1672

Table 4-5: results of using 8, 16, and 32 as window size of DWT on File1-EANG.

File 1-EEV	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	96.365%	0.1287	97.787%	0.0954	98.792%	0.0719
RF	90.279%	0.1934	93.872%	0.1678	96.031%	0.1504

Table 4-6: results of using 8, 16, and 32 as window size of DWT on File1-EEV.

File 1-IANG	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	95.013%	0.1500	97.362%	0.1132	98.188%	0.0920
RF	91.631%	0.1870	95.234%	0.1657	96.290%	0.1442

Table 4-7: results of using 8, 16, and 32 as window size of DWT on File1-IANG.

File 1-IEV	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	96.957%	0.1200	99.234%	0.0634	98.878%	0.0722
RF	92.984%	0.1893	97.277%	0.1515	96.894%	0.1420

Table 4-8: results of using 8, 16, and 32 as window size of DWT on File1-IEV.

We merge DWT on time stamps with window size 32 and DWT on energy with window size 8. The results displayed in **Table 4-9**, **Table 4-10**, **Table 4-11**, and **Table 4-12**.

File 1-EANG	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	97.066%	0.1137
RF	92.752%	0.1699

Table 4-9: results of merging DWT on time and on energy on File1-EANG.

File 1-EEV	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	98.878%	0.0737
RF	95.772%	0.1455

Table 4-10: results of merging DWT on time and on energy on File1-EEV.

File 1-IANG	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	98.533%	0.0819
RF	96.290%	0.1458

Table 4-11: results of merging DWT on time and on energy on File1-IANG.

File 1-IEV	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	99.051%	0.0665
RF	97.325%	0.1384

Table 4-12: results of merging DWT on time and on energy on File1-IEV.

4.2.3 Phase 3

There are 264 features for each instance (256 features coming from DWT on time stamps and 8 features getting from DWT on energy). We use feature ranking described in 3.1.3. by selecting the top 20 features from the perspectives of Information Gain, Gain Ratio, Chi-squared, and Relief. The features with high score in these four factors are definitely what we are looking for, because they contain relevant information. In **Figure 4-3**, we found the top 40 features from the total 264 features. We selected Top 35 features according to its occurrences in all tests. In time-axis, feature 1 to 32 are all coming from the results of DWT on time stamps while 33 is the outcome of DWT on energy. In the grid, the number represents the times this specific feature appeared during the feature selections. **Table 4-13**, **Table 4-14**, **Table 4-15**, and **Table 4-16** shows the results after feature ranking.

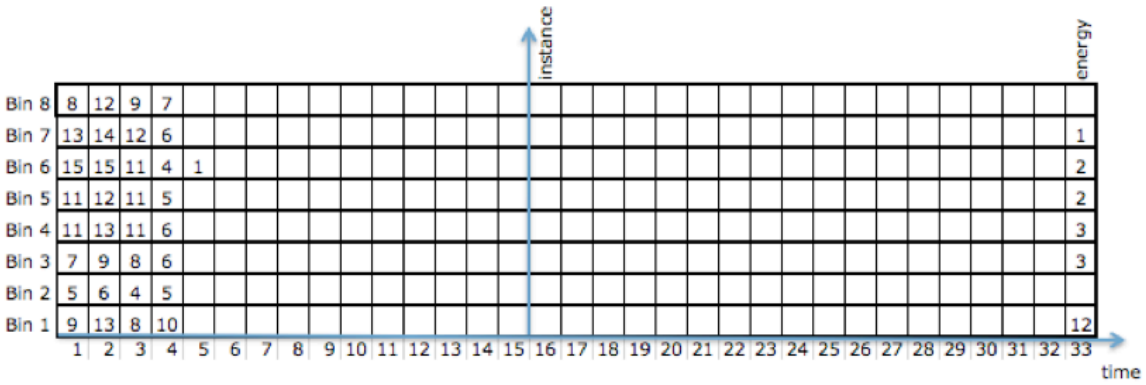

Figure 4-3: There are 264 features for each time instance displayed above.

File 1-EANG	Top 35	
	Accuracy	Root mean squared error
BDT	98.620%	0.0795
RF	98.792%	0.0884

Table 4-13: results of using features ranking on File1-EANG.

File 1-EEV	Top 35	
	Accuracy	Root mean squared error
BDT	98.792%	0.0776
RF	99.310%	0.0704

Table 4-14: results of using features ranking on File1-EEV.

File 1-IANG	Top 35	
	Accuracy	Root mean squared error
BDT	98.620%	0.0813
RF	99.137%	0.0749

Table 4-15: results of using features ranking on File1-IANG.

File 1-IEV	Top 35	
	Accuracy	Root mean squared error
BDT	99.051%	0.0671
RF	99.224%	0.0667

Table 4-16: results of using features ranking on File1-IEV.

4.2.4 Phase 4

We use Principal Component Analysis to continue with our goal of feature reduction. After following the steps stated in 3.1.4., we finally got 9 features out of 35. The results have been displayed in **Table 4-17**, **Table 4-18**, **Table 4-19**, and **Table 4-20**.

File 1-EANG	PCA-9features	
	Accuracy	Root mean squared error
BDT	97.757%	0.1027
RF	97.239%	0.1138

Table 4-17: results of using PCA on File1-EANG.

File 1-EEV	PCA-9features	
	Accuracy	Root mean squared error
BDT	97.498%	0.1058
RF	96.549%	0.1170

Table 4-18: results of using PCA on File1-EEV.

File 1-IANG	PCA-9features	
	Accuracy	Root mean squared error
BDT	98.533%	0.0827
RF	98.706%	0.0818

Table 4-19: results of using PCA on File1-IANG.

File 1-IEV	PCA-9features	
	Accuracy	Root mean squared error
BDT	98.447%	0.0856
RF	98.965%	0.0811

Table 4-20: results of using PCA on File1-IEV.

4.3 Overall Description

We formulated 8 bins by adding up every 8 connecting values from the 64 energy values corresponding to the angles at each time point. Then we generated 264 wavelet coefficients using a window of size 32 on time stamps and a window of size 8 on angle stamps. We kept the best 35 coefficients that rank the highest from the perspectives of Information Gain, Rain Ratio, Chi-squared, and Relief. After that, we did PCA on these 35 features and kept the best 10 components as the final feature set.

4.4 Experiments on Different Datasets

There are three other different datasets displayed here for testing our methodology. All of them will be tested using the same way as above. We just displayed the results of each phase only.

4.4.1 Datasets II

Figure 4-4 and **Figure 4-5** displayed the dataset II as the same format as above, all the result sets were shown in **Table 4-21**, **Table 4-22**, **Table 4-23**, **Table 4-24**, and **Table 4-25**.

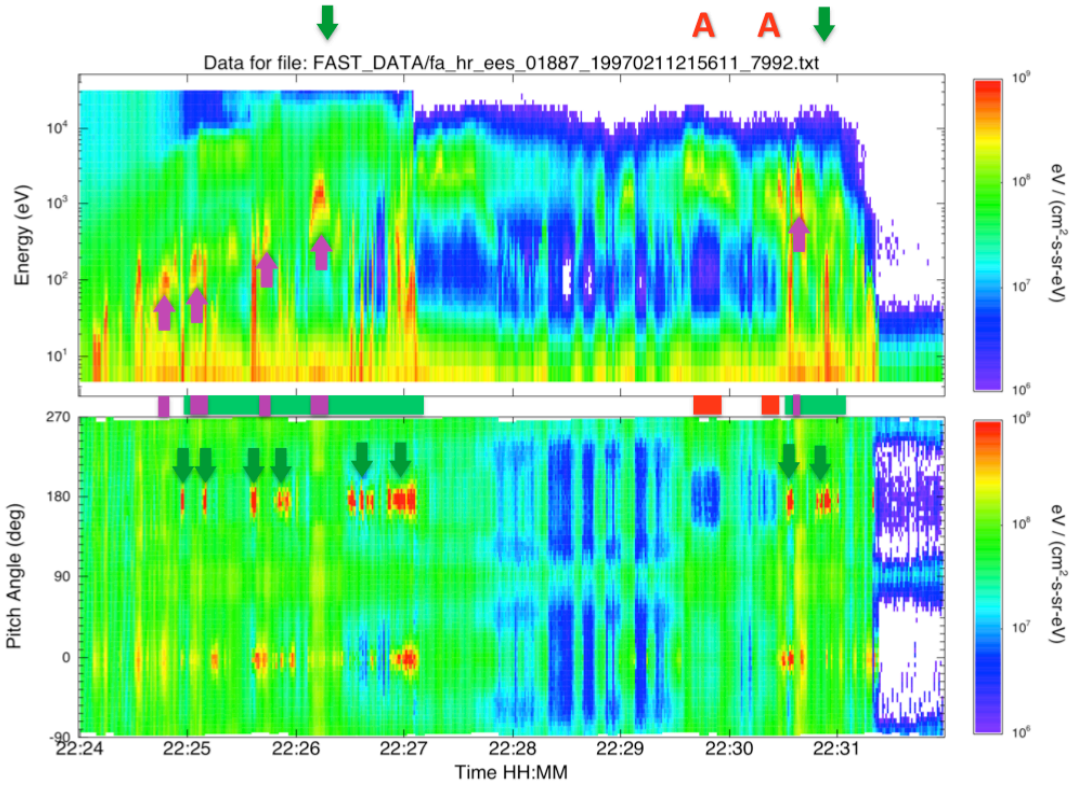

Figure 4-4: The two spectrograms (EANG/EEV) of Dataset II. It shows the dataset taken from approximately 22:24 to 22:32.

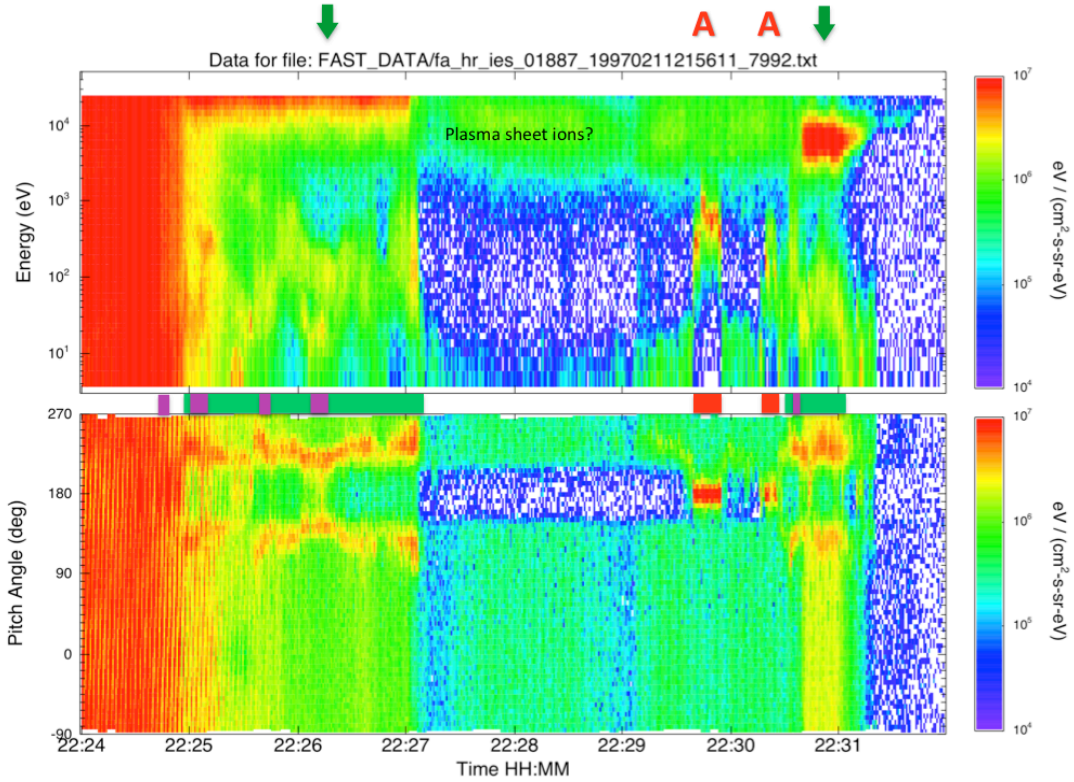

Figure 4-5: The two spectrograms (IANG/IEV) of Dataset II. It shows the dataset taken from approximately 22:24 to 22:32.

File 2-EANG	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	83.179%	0.2737	90.040%	0.2122
RF	84.433%	0.2398	89.314%	0.1967

File 2-EEV	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	93.470%	0.1722	96.768%	0.1249
RF	93.272%	0.1582	96.438%	0.1226

File 2-IANG	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	91.887%	0.1977	93.008%	0.1834
RF	91.557%	0.1776	92.216%	0.1673

File 2-IEV	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	95.251%	0.1512	95.646%	0.1423
RF	94.525%	0.1446	95.185%	0.1324

Table 4-21: Bin Package result sets of Data II.

File 2-EANG	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	93.572%	0.1705	96.402%	0.1275	96.902%	0.1181
RF	91.518%	0.1956	93.138%	0.1843	94.680%	0.1745

File 2-EEV	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	97.482%	0.1078	98.801%	0.0743	98.451%	0.0866
RF	93.903%	0.1732	95.670%	0.1556	96.633%	0.1502

File 2-IANG	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	97.018%	0.1190	97.468%	0.1067	96.431%	0.1332
RF	94.632%	0.1448	95.936%	0.1383	97.845%	0.0999

File 2-IEV	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	97.614%	0.1075	98.201%	0.0933	97.913%	0.0993
RF	95.030%	0.1489	96.935%	0.1299	96.835%	0.1277

Table 4-22: DWT on Time result sets of Data II.

File 2-EANG	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	96.835%	0.1176
RF	94.950%	0.1675

File 2-EEV	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	98.721%	0.0786
RF	96.700%	0.1410

File 2-IANG	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	97.913%	0.1021
RF	96.768%	0.1303

File 2-IEV	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	98.115%	0.0905
RF	97.374%	0.1230

Table 4-23: DWT on Time and on Energy result sets of Data II.

File 2-EANG	Top 35	
	Accuracy	Root mean squared error
BDT	98.317%	0.0887
RF	98.182%	0.1029

File 2-EEV	Top 35	
	Accuracy	Root mean squared error
BDT	98.788%	0.0756
RF	98.586%	0.0760

File 2-IANG	Top 35	
	Accuracy	Root mean squared error
BDT	98.317%	0.0909
RF	98.451%	0.0769

File 2-IEV	Top 35	
	Accuracy	Root mean squared error
BDT	98.855%	0.0753
RF	98.855%	0.0697

Table 4-24: Features Ranking result sets of Data II.

File 2-EANG	PCA-9features	
	Accuracy	Root mean squared error
BDT	97.306%	0.1129
RF	96.700%	0.1258

File 2-EEV	PCA-9features	
	Accuracy	Root mean squared error
BDT	98.586%	0.0830
RF	98.586%	0.0921

File 2-IANG	PCA-9features	
	Accuracy	Root mean squared error
BDT	98.653%	0.0810
RF	98.586%	0.0742

File 2-IEV	PCA-9features	
	Accuracy	Root mean squared error
BDT	98.653%	0.0812
RF	98.721%	0.0694

Table 4-25: PCA result sets of Data II.

4.4.2 Datasets III

Figure 4-6 and **Figure 4-7** displayed the dataset III as the same format as above, all the result sets were shown in **Table 4-26**, **Table 4-27**, **Table 4-28**, **Table 4-29**, and **Table 4-30**.

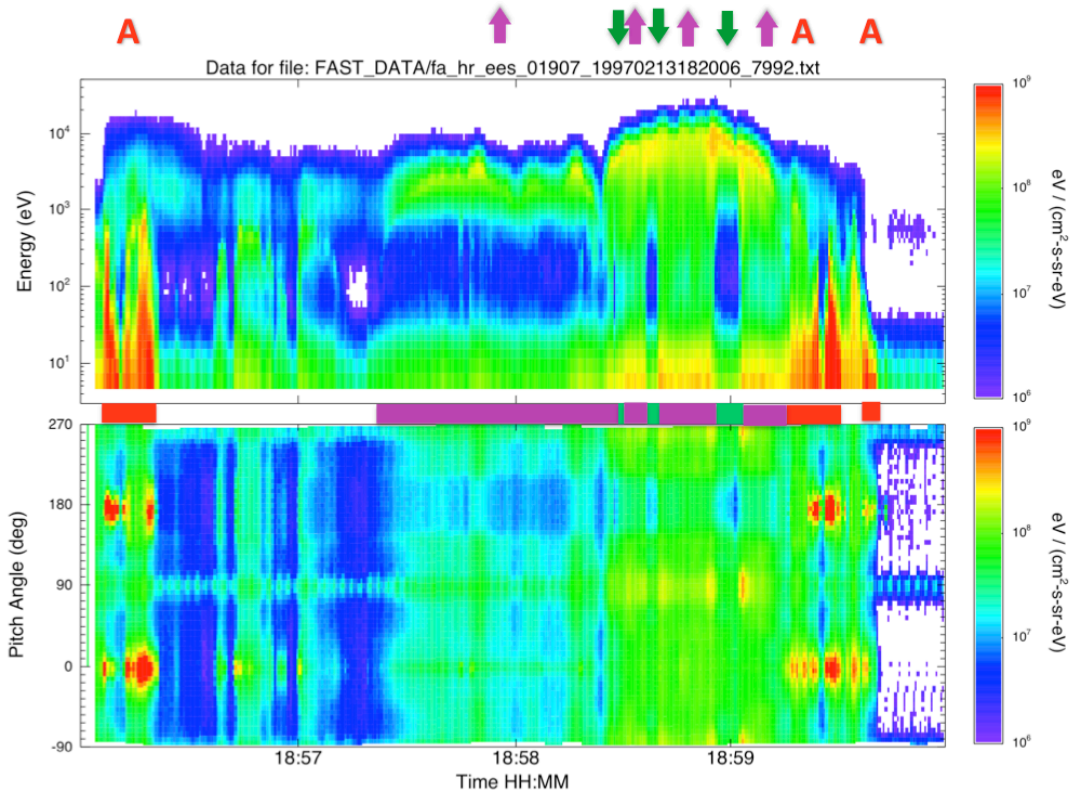

Figure 4-6: The two spectrograms (EANG/EEV) of Dataset III. It shows the dataset taken from approximately 18:56 to 19:00.

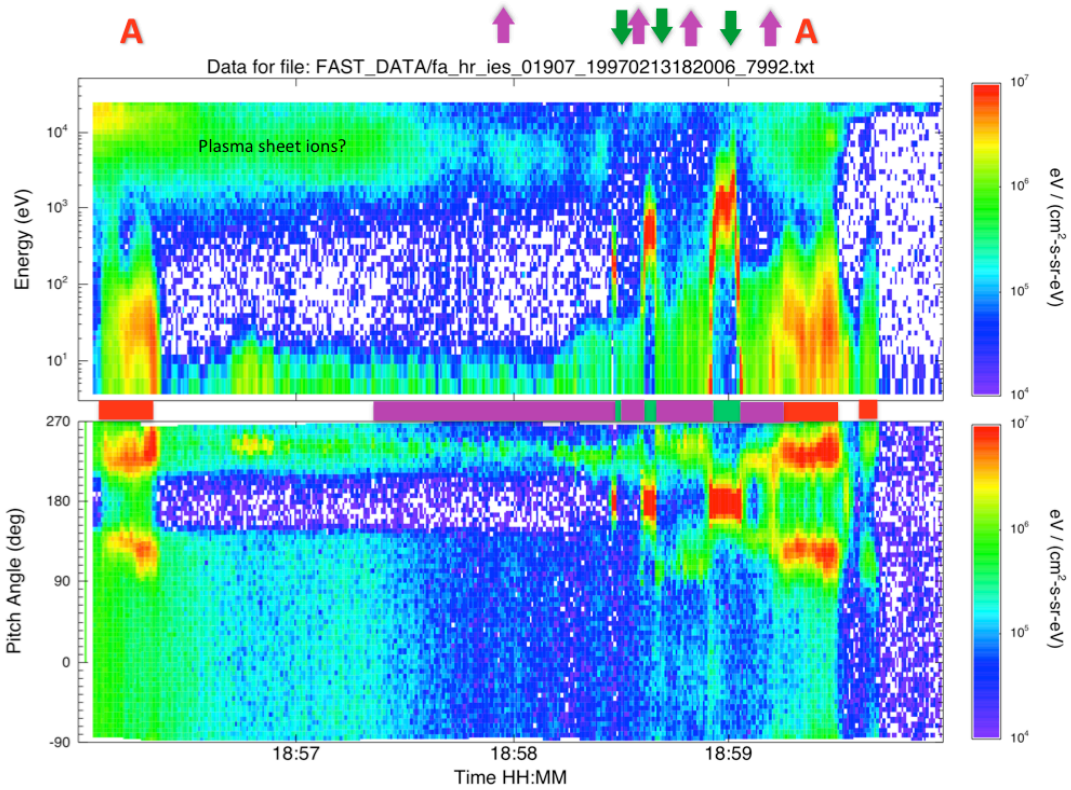

Figure 4-7: The two spectrograms (IANG/IEV) of Dataset III. It shows the dataset taken from approximately 18:56 to 19:00.

File 3-EANG	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	78.192%	0.3213	80.851%	0.2988
RF	76.330%	0.2928	81.649%	0.2626

File 3-EEV	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	90.692%	0.2103	89.362%	0.2213
RF	89.628%	0.2006	89.096%	0.1965

File 3-IANG	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	86.436%	0.2553	87.766%	0.2456
RF	85.638%	0.2203	87.234%	0.2092

File 3-IEV	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	85.638%	0.2637	88.564%	0.2329
RF	85.106%	0.2284	90.692%	0.2022

Table 4-26: Bin Package result sets of Data III.

File 3-EANG	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	92.141%	0.1900	91.967%	0.1960	93.044%	0.1732
RF	91.328%	0.1934	89.751%	0.1898	92.754%	0.1823

File 3-EEV	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	96.206%	0.1380	94.737%	0.1630	95.942%	0.1435
RF	92.954%	0.1708	91.690%	0.1755	93.913%	0.1610

File 3-IANG	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	93.496%	0.1762	93.906%	0.1685	95.362%	0.1429
RF	90.786%	0.1999	91.690%	0.1854	92.754%	0.1632

File 3-IEV	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	95.122%	0.1492	93.075%	0.1823	95.073%	0.1488
RF	92.141%	0.1850	92.798%	0.1766	95.073%	0.1646

Table 4-27: DWT on Time result sets of Data III.

File 3-EANG	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	92.174%	0.1823
RF	94.203%	0.1762

File 3-EEV	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	95.652%	0.1372
RF	93.913%	0.1598

File 3-IANG	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	96.812%	0.1241
RF	93.913%	0.1577

File 3-IEV	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	95.362%	0.1498
RF	94.203%	0.1614

Table 4-28: DWT on Time and on Energy result sets of Data III.

File 3-EANG	Top 35	
	Accuracy	Root mean squared error
BDT	92.754%	0.1746
RF	93.623%	0.1598

File 3-EEV	Top 35	
	Accuracy	Root mean squared error
BDT	95.362%	0.1495
RF	96.232%	0.1230

File 3-IANG	Top 35	
	Accuracy	Root mean squared error
BDT	95.942%	0.1400
RF	95.942%	0.1165

File 3-IEV	Top 35	
	Accuracy	Root mean squared error
BDT	95.652%	0.1484
RF	95.073%	0.1274

Table 4-29: Features Ranking result sets of Data III.

File 3-EANG	PCA-9features	
	Accuracy	Root mean squared error
BDT	93.044%	0.1822
RF	93.044%	0.1546

File 3-EEV	PCA-9features	
	Accuracy	Root mean squared error
BDT	95.942%	0.1409
RF	95.362%	0.1351

File 3-IANG	PCA-9features	
	Accuracy	Root mean squared error
BDT	95.942%	0.1395
RF	95.942%	0.1270

File 3-IEV	PCA-9features	
	Accuracy	Root mean squared error
BDT	95.362%	0.1468
RF	95.652%	0.1210

Table 4-30: PCA result sets of Data III.

4.4.3 Datasets IV

Figure 4-8 and **Figure 4-9** displayed the dataset IV as the same format as above, all the result sets were shown in **Table 4-31**, **Table 4-32Table 4-33**, **Table 4-34**, and **Table 4-35**.

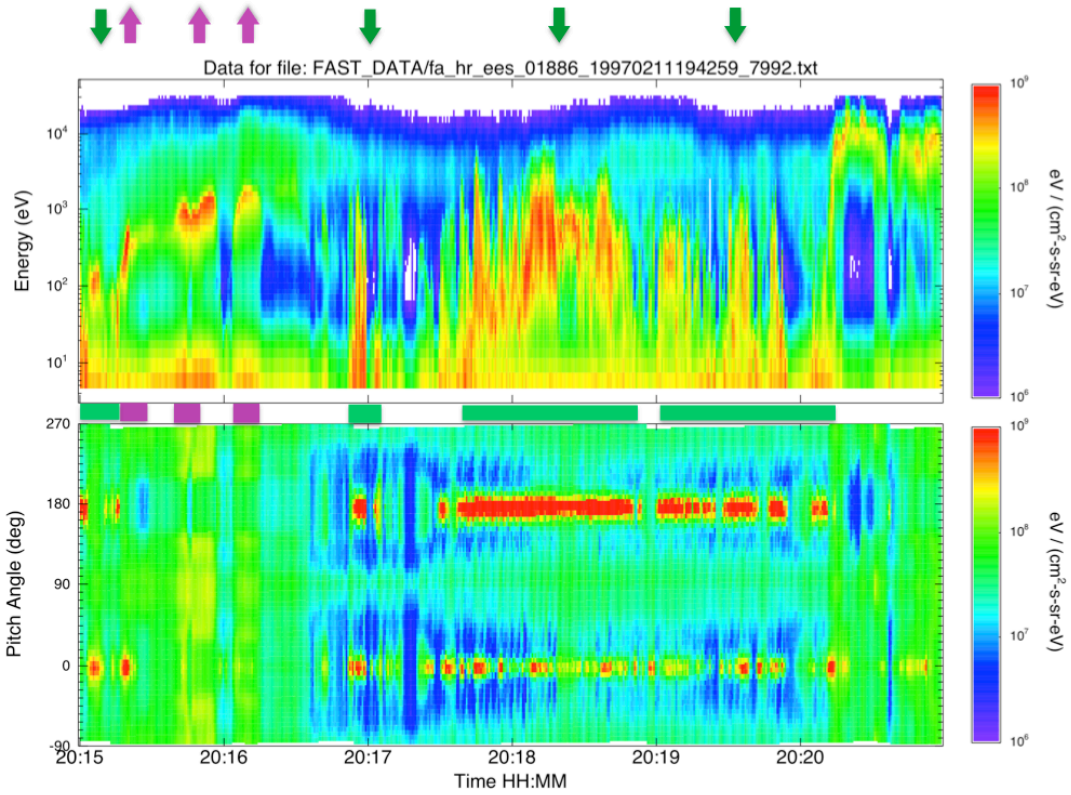

Figure 4-8: The two spectrograms (EANG/EEV) of Dataset IV. It shows the dataset taken from approximately 20:15 to 20:21.

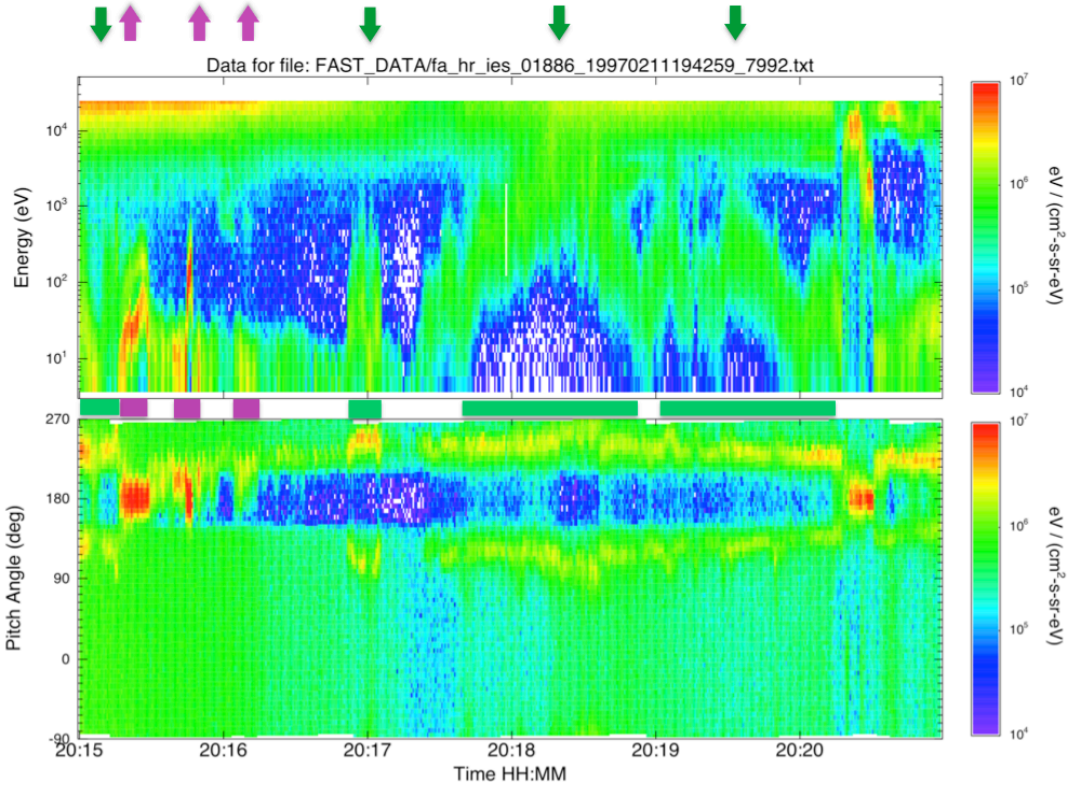

Figure 4-9: The two spectrograms (IANG/IEV) of Dataset IV. It shows the dataset taken from approximately 20:15 to 20:21.

File 4-EANG	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	78.697%	0.2993	84.331%	0.2636
RF	80.810%	0.2729	84.331%	0.2408

File 4-EEV	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	88.028%	0.2368	91.725%	0.1992
RF	88.380%	0.2087	91.725%	0.1823

File 4-IANG	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	79.754%	0.2949	85.211%	0.2641
RF	79.930%	0.2626	85.739%	0.2173

File 4-IEV	4 bins		8 bins	
	Accuracy	RMSE	Accuracy	RMSE
BDT	89.085%	0.2272	93.310%	0.1803
RF	88.732%	0.2053	93.134%	0.1670

Table 4-31: Bin Package result sets of Data IV.

File 4-EANG	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	90.909%	0.2075	92.586%	0.1886	94.972%	0.1567
RF	83.601%	0.2406	88.788%	0.2178	91.434%	0.1987

File 4-EEV	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	95.187%	0.1506	96.383%	0.1305	96.648%	0.1241
RF	90.909%	0.2006	92.767%	0.1826	94.227%	0.1739

File 4-IANG	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	96.078%	0.1391	96.022%	0.1405	96.834%	0.1209
RF	93.939%	0.1802	93.671%	0.1720	95.531%	0.1621

File 4-IEV	DWT-8onTime		DWT-16onTime		DWT-32onTime	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
BDT	95.187%	0.1544	95.479%	0.1404	96.462%	0.1304
RF	91.979%	0.1882	94.213%	0.1762	93.296%	0.1839

Table 4-32: DWT on Time result sets of Data IV.

File 4-EANG	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	92.924%	0.1829
RF	90.130%	0.1952

File 4-EEV	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	96.648%	0.1229
RF	90.875%	0.1955

File 4-IANG	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	96.276%	0.1317
RF	95.531%	0.1668

File 4-IEV	DWT-32onTime+8onEnergy	
	Accuracy	RMSE
BDT	96.834%	0.1237
RF	94.600%	0.1800

Table 4-33: DWT on Time and on Energy result sets of Data IV.

File 4-EANG	Top 35	
	Accuracy	Root mean squared error
BDT	96.648%	0.1267
RF	96.462%	0.1245

File 4-EEV	Top 35	
	Accuracy	Root mean squared error
BDT	97.765%	0.1024
RF	97.021%	0.1173

File 4-IANG	Top 35	
	Accuracy	Root mean squared error
BDT	96.834%	0.1242
RF	97.393%	0.1081

File 4-IEV	Top 35	
	Accuracy	Root mean squared error
BDT	97.952%	0.0999
RF	97.021%	0.1117

Table 4-34: Features Ranking result sets of Data IV.

File 4-EANG	PCA-9features	
	Accuracy	Root mean squared error
BDT	96.276%	0.1342
RF	96.648%	0.1160

File 4-EEV	PCA-9features	
	Accuracy	Root mean squared error
BDT	96.648%	0.1277
RF	95.531%	0.1306

File 4-IANG	PCA-9features	
	Accuracy	Root mean squared error
BDT	96.648%	0.1279
RF	96.462%	0.1266

File 4-IEV	PCA-9features	
	Accuracy	Root mean squared error
BDT	95.903%	0.1378
RF	96.089%	0.1235

Table 4-35: PCA result sets of Data IV.

Chapter 5 Conclusions and Discussion

5.1 Conclusions

We introduced a set of techniques to build a machine learning system for Plasma particles data obtained from FAST satellite and PLOAR spacecraft. The high performance achieved using our model suggests that our methodology is found to be effective in performing the required tasks.

In Chapter 4 we optimized the parameters in every phase to achieve better classification performance. For instance, we chose the best size for bin package; considered the different sizes of the window for Discrete Wavelet Transform; evaluated the whole set of features using four determining factors; and finally reduced the dimensionalities of features dramatically through Principal Component Analysis. Moreover, we selected the best suitable classifiers working with our feature set and took advantage of 10-fold cross-validation to check the stability and portability of our model.

5.2 Discussion

5.2.1 Feature Analysis

Looking at all the result tables we displayed in 4.2, we learned that Discrete Wavelet Transform has the biggest impact on classification performance. Without losing much relevant information, feature ranking reduced the feature dimensionality from 264 to 35 and Principal Component Analysis decreased it to 9 ultimately. See **Figure 5-1**.

In **Figure 4-3** we can easily found that the top features stay very close to each other. They were concentrated on the first four window stamps in each time instance and the first bin on energy stamp.

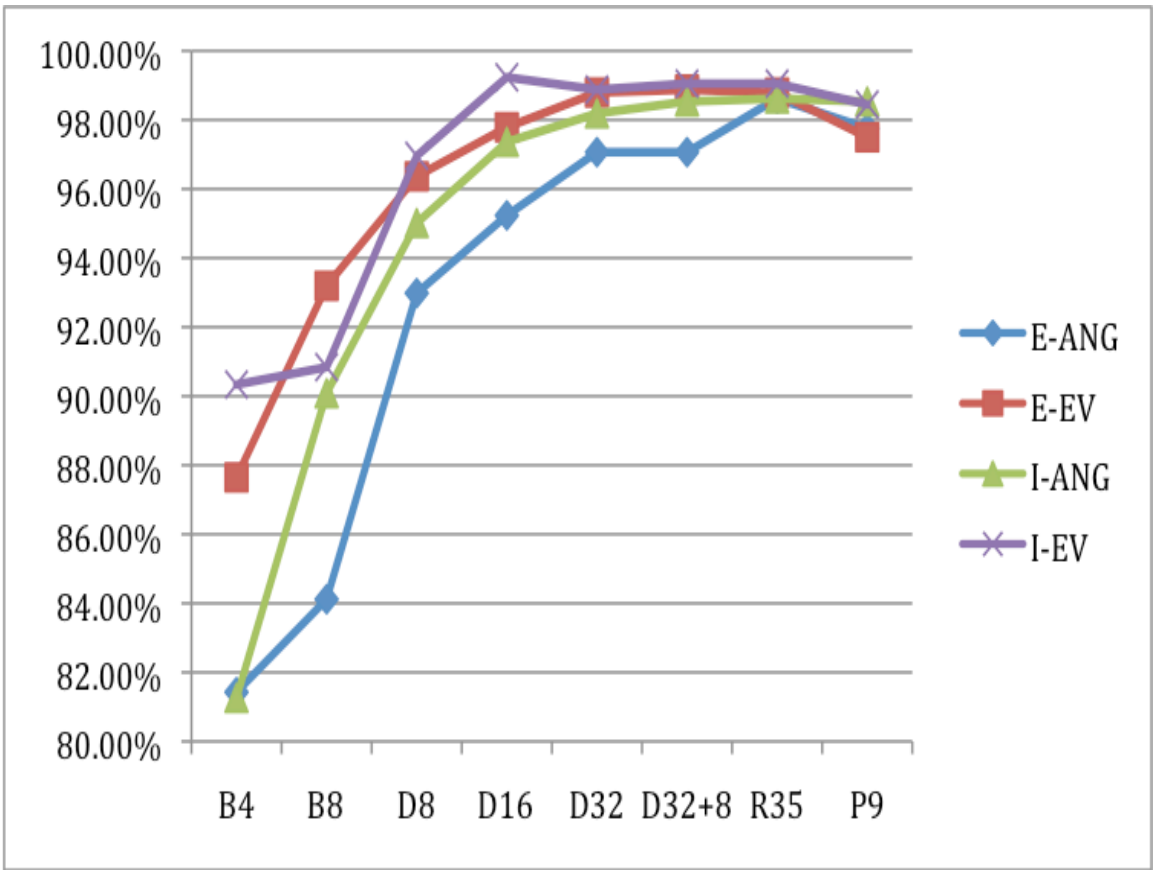

Figure 5-1: Accuracy results in each phase on dataset I.

5.2.2 Window Slides

We tried to use "window slide" to improve performance. In Window Slide, we improve the classification performance by correcting the prediction work done by the model. We find the predictions on test data and take a window of size K predictions (K must be an odd number) from the beginning. Within this window, If previous $(\frac{k-1}{2})$ predictions and following $(\frac{k-1}{2})$ predictions are all the same, say class a, then we change the centered prediction $(\frac{k+1}{2})$ to class a. Otherwise, move the window by one prediction and repeat the same process till the end.

$$x = \begin{cases} a & \text{if previous } \left(\frac{k-1}{2}\right) \text{ numbers and following } \left(\frac{k-1}{2}\right) \text{ numbers are all class a} \\ x & \text{otherwise} \end{cases} \quad (13)$$

5.2.3 Data Analysis

In order to know the portability of our model, we took four different datasets taken at different time spans. All of them were EEV part: **Figure 5-2** (File 1), **Figure 5-3** (File 2/Data II), **Figure 5-4** (File 3/Data III), and **Figure 5-5** (File 4/Data IV) displayed as below. We plot the top two features according to the algorithm discussed in 3.1.3 altogether the four datasets. In **Figure 5-7**, most of the points were concentrated in a way that makes it hard to differentiate. We took the logarithmic of our datasets and the new distribution of our classes was shown in **Figure 5-7**. We can understand the figure well according to **Table 5-1: Explanation for Figure 5-7**

Class 0	Black	File 1	.
Class 1	Blue	File 2	*
Class 2	Green	File 3	+
Class 3	Red	File 4	^

Table 5-1: Explanation for Figure 5-7

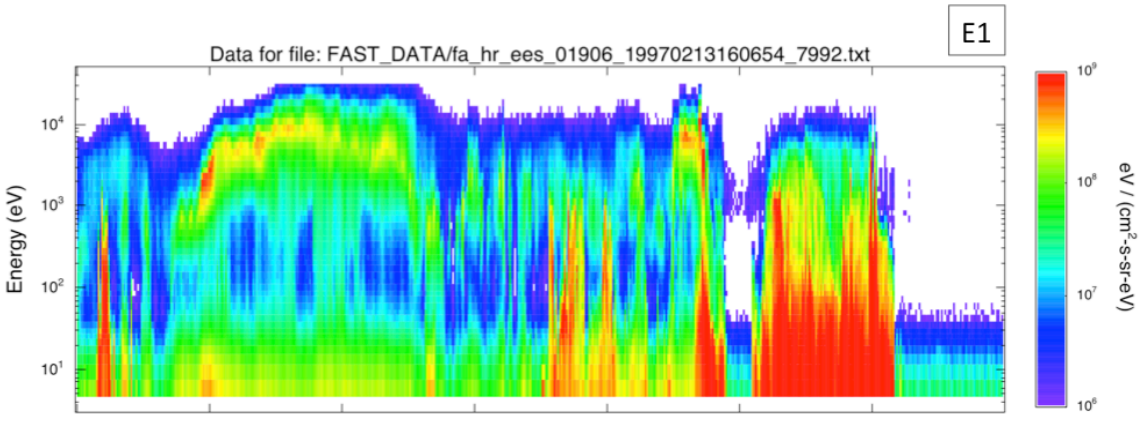

Figure 5-2: One spectrograms of EEV, the dataset taken from approximately 16:44 to 16:50.

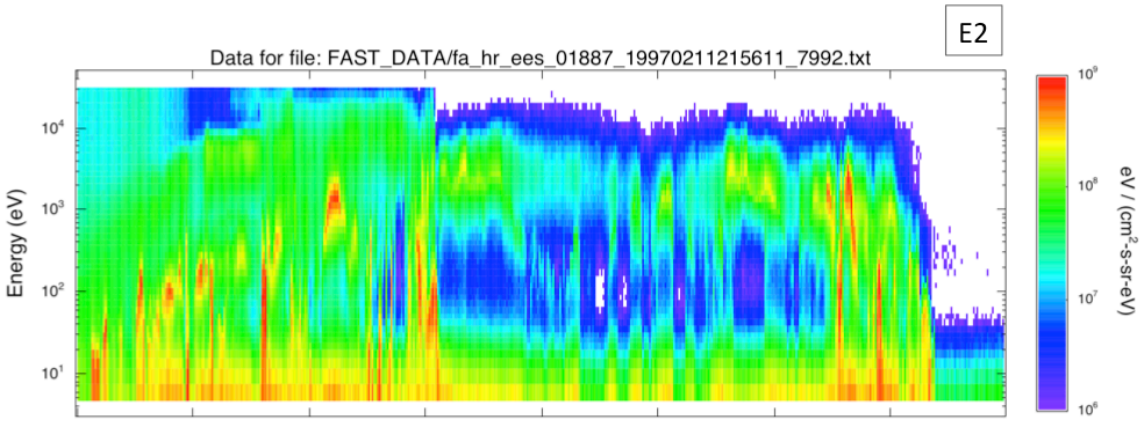

Figure 5-3: One spectrograms of EEV, the dataset taken from approximately 22:24 to 22:31.

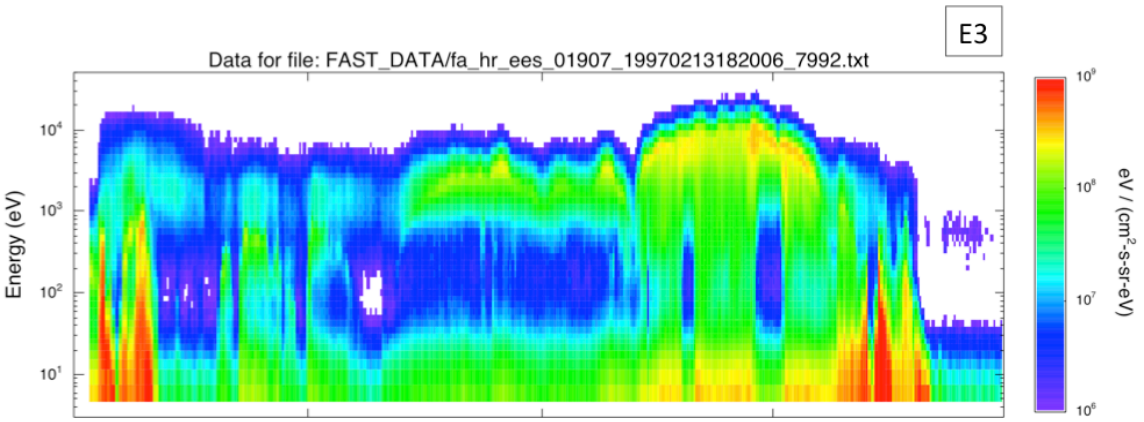

Figure 5-4: One spectrograms of EEV, the dataset taken from approximately 18:56 to 18:90.

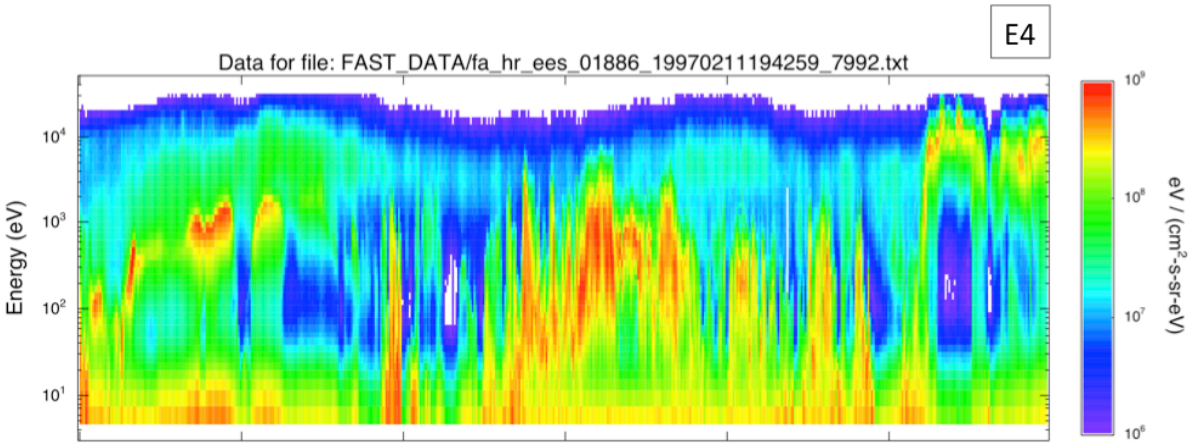

Figure 5-5: One spectrograms of EEV, the dataset taken from approximately 20:15 to 20:21.

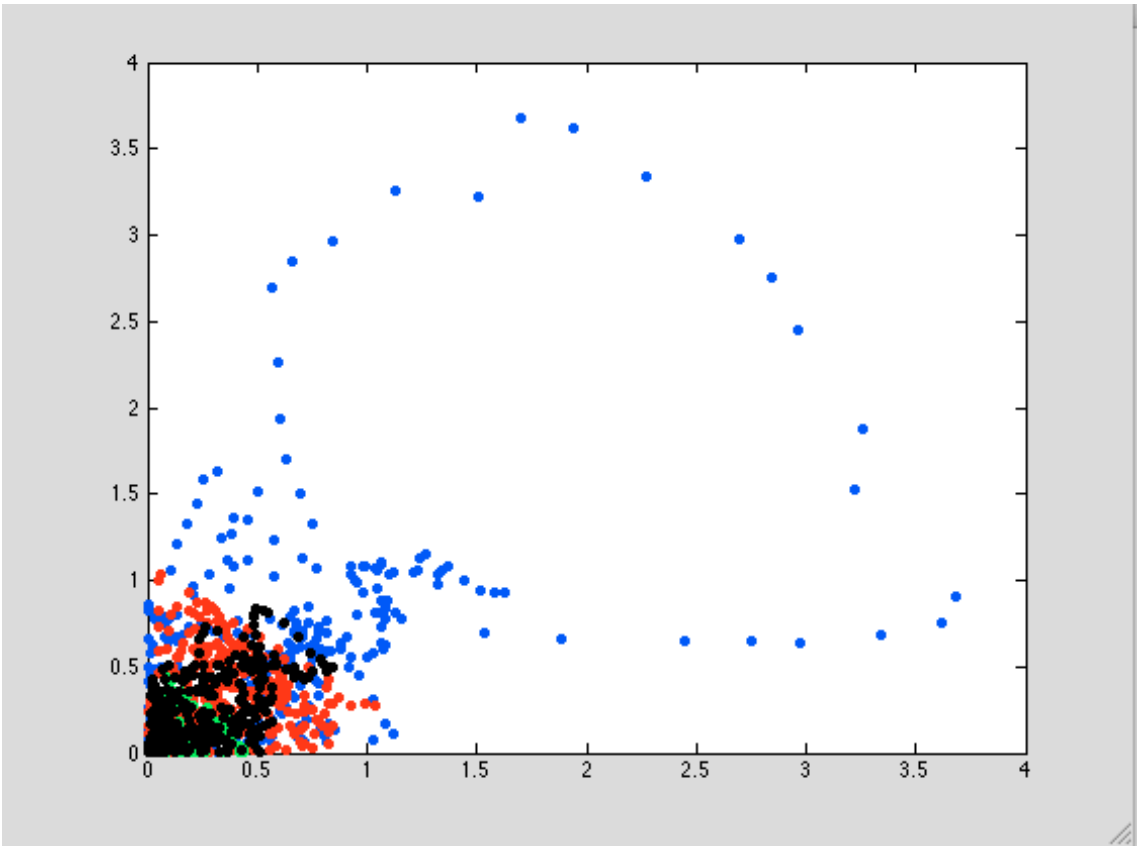

Figure 5-6: 4 files plotting together with different colors respectively, Data / (10^10)

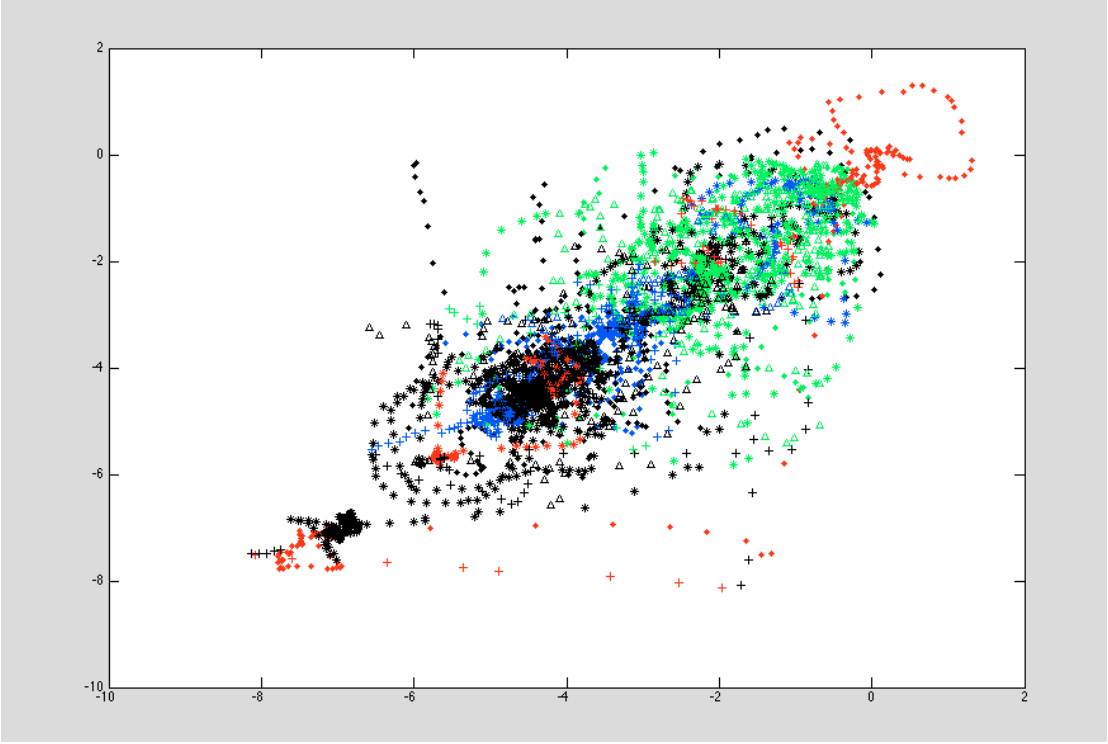

Figure 5-7: 4 files plotting together with different colors respectively, Log (Data / (10^10))

I. Mixture of Gaussian

We calculated the mixture of Gaussian for all four datasets. The results were shown in **Table 5-2**, **Table 5-3**, **Table 5-4**, and **Table 5-5**.

Components	Mixing proportion	Mean (X)	Mean (Y)	
1	0.19566	1.0793	1.0736	* 1.0e+08
2	0.128486	3.4093	3.1506	* 1.0e+08
3	0.223071	7.3234	7.5107	* 1.0e+09
4	0.11028	7.5314	7.5866	* 1.0e+06
5	0.018895	2.3786	0.158	* 1.0e+09
6	0.248296	2.0094	2.021	* 1.0e+08
7	0.042827	4.0061	5.0114	* 1.0e+08
8	0.032485	5.7744	4.9677	* 1.0e+08

Table 5-2: Results of Mixture of Gaussian for File1-EEV.

Components	Mixing proportion	Mean (X)	Mean (Y)	
1	0.080984	5.2321	2.7163	* 1.0e+09
2	0.083659	1.5335	1.4951	* 1.0e+07
3	0.125265	8.0829	7.6548	* 1.0e+08
4	0.031501	1.1185	0.9596	* 1.0e+09
5	0.406477	1.2342	1.2494	* 1.0e+08
6	0.030361	5.4432	4.939	* 1.0e+08
7	0.057933	2.8755	6.1286	* 1.0e+09
8	0.040333	1.0385	1.1461	* 1.0e+09
9	0.143486	1.5262	1.6452	* 1.0e+09

Table 5-3: Results of Mixture of Gaussian for File2-EEV.

Components	Mixing proportion	Mean (X)	Mean (Y)	
1	0.281194	2.641	3.0802	* 1.0e+08
2	0.101155	1.4965	2.0986	* 1.0e+09
3	0.394765	6.5673	6.5746	* 1.0e+07
4	0.162018	5.6487	6.0989	* 1.0e+08
5	0.060868	3.6045	1.2566	* 1.0e+09

Table 5-4: Results of Mixture of Gaussian for File3-EEV.

Components	Mixing proportion	Mean (X)	Mean (Y)	
1	0.093405	1.0831	3.3716	* 1.0e+09
2	0.078207	4.7677	2.008	* 1.0e+09
3	0.035256	6.6485	9.3194	* 1.0e+08
4	0.066716	1.2004	0.6615	* 1.0e+09
5	0.075635	2.9241	1.2115	* 1.0e+09
6	0.185994	1.2015	1.2474	* 1.0e+08
7	0.058208	0.4728	1.5449	* 1.0e+09
8	0.07723	5.4632	4.9798	* 1.0e+09
9	0.102708	3.705	5.0415	* 1.0e+09
10	0.045525	1.7328	0.4274	* 1.0e+09
11	0.181115	3.3996	3.3235	* 1.0e+08

Table 5-5: Results of Mixture of Gaussian for File4-EEV.

II. Parzen Windows

We use the Parzen Windows techniques to check the possibilities of each instance in File3_EEV and File4_EEV belong to File1_EEV and File2_EEV. Since our data's range is huge, we only could get comparatively clear results by setting the Parzen window's size h to a big number.

5.2.4 Code

Our programming code implements the Discrete Wavelet Transform (3.1.2), Principal Component Analysis (3.1.4), Window Slides (5.2.2), Mixture of Gaussians, and Parzen Window (5.2.3). All of them are available upon request.

Chapter 6 Future Work

6.1 Testing

All the experiments presented here demonstrate that our methodology succeeds in doing an automated approach to classification. However, in order to obtain a reliable model, we need to test the model on other datasets also. All the techniques we displayed in 5.2 definitely will help us understanding the dataset well and lead to good research direction.

Moreover, the automated tools will be anticipated to work on tens of thousands of spectrograms with only a few of training datasets.

6.2 Transfer Learning

Due to the nature of the domain data, we plan to use transfer learning to expedite the analysis of thousands of spectrograms covering broad regions of Earth's upper atmosphere. Transfer Learning ([25][26][27][28][29][30][31]) will help to exploit knowledge gathered from previous experience, to expedite the model generation process. In that way, the target task can be different from the source task.

We plan to selectively add examples from previous tasks that both increase classification performance, and correspond to cases where two or more data models disagree with the class label. The two-mode approach acts as a strong filter in selecting only informative instances during model generation ([32][33][34]).

6.3 Mixture of Experts

Also, we are going to use Mixture of Experts to attack the problems caused from having thousands of spectrograms with different inputs. In the mixture of expert model,

different learners cover the different input regions, and there is a “soft” switching between learners ([35][36]) that enhances the “good” experts and diminish the “bad” ones to work on different input region.

Mixture learning procedure consists of two tasks. The first one is to learn the parameters of individual expert networks; the second is to learn the parameters of the gating network, which is used to decide where to make a split. Based on the probability we partition the space, different experts will be assigned to different partitions to improve predictions.

References

[01] P. T. Newell and R. J. M. Greenwald, R. A., "The Role of The Ionosphere In Aurora and Space Weather. " Reviews Geophysics, vol. 39, no. 2, pp. 137-149, 2001.

[02] R. F. Pfaff, J. E. Borovsky, D. T. Young, "Measurement Techniques in Space Plasma: Fields." AGU Monograph 103, 1998.

[03] R. F. Pfaff, J. E. Borovsky, D. T. Young, "Measurement Techniques in Space Plasma: Particles." AGU Monograph 103, 1998.

[04] M. Wüest, D. S. Evans, R. von Steiger, "Calibration of Particle Instruments in Space Physics." Published for International Space Science Institute by ESA Communications, Noordwijik, The Netherlands, 2007.

[05] G. Paschmann, S. Haaland, and R. Treumann, Eds., "Auroral Plasma Physics." Kluwer Academic Publisher, 2003, vol. 15.

[06] C. W. Carlson, R. F. Pfaff, J. G. Watzin, "The Fast Auroral SnapshoT (FAST) mission." Geophys. Res. Lett., DOI:10.1029/98GL01592, 2008.

[07] N. M. Ball, "Utilizing Astroinformatics to Maximize the Science Return of the Next Generation Virgo Cluster Survey." arXiu: 1110.568v1, astro-ph.CO., 2011.

[08] R. J. Strangeway, L. Kepdo, R. C. Elphic, and et al., "Fast Observations of Vlf Waves in the Auroral Zone: Evidence of Very Low Plasma Densities." Geophysical Research Letters, vol. 25, no. 12, pp. 2065-2068, 1998.

[09] P. T. Newell, C.-I. Meng, "Mapping The Dayside Ionosphere To The Magnetosphere According to Particle Precipitation Characteristics." Geophysical Research Letters, vol. 19, no. 6, pp. 609-612, 1992.

[10] P. T. Newell, T. Sotirelis, K. Liou, and F. J. Rich, "Pairs of Solar Wind-magnetosphere Coupling Functions: Combining a Merging Term with a Viscous Term Works Best." Journal of Geophysical Research: Space Physics, vol. 113, no. A4, no. A04218, pp. 1-10, 2008.

[11] D. A. Hardy, M. S. Gussenhoven, and D. Brautigam, "A Statistical Model of Auroral Ion Precipitation." Journal of Geophysical Research: Space Physics, vol. 94, no. A1, pp. 370-392, 1989.

[12] S. V. Badman, S. W. H. Cowley, J.-C. Gérard, and D. Grodent, "A Statistical Analysis of the Location and Width of Saturn's Southern Auroras." Annales Geophysicae, vol. 24, no. 12, pp. 3533-3545, 2006.

[13] Y. –M. Tanaka, T. Aso, B. Gustavsson, K. Tanabe, Y. Ogawa, A. Kadokura, H. Miyaoka, T. Sergienko, U. Brändström, I. Sandahl, "Feasibility Study on Generalized-aurora Computed Tomography." Annales Geophysicae, vol. 29, no. 3, pp. 551-562, 2011.

[14] G. Lointier, T. Dudok de Wit, C. Hanuise, X. Vallières, and J.-P. Villain, "A Statistical Approach for Identifying the Ionospheric Footprint of Magnetospheric Boundaries from Super DARN Observation." Annales Geophysicae, vol. 26, no. 2, pp. 305-314, 2008.

[15] H. A. Elliott, R. H. Comfort, P. D. Craven, M. O. Chandler, and T. E. Moore, "Solar Wind influence on the Oxygen content of Ion Outflow in the High-altitude Polar Cap during Solar Minimum Conditions." Journal of Geophysical Research: Space Physics, vol. 106, no. A4, pp. 6067-6084, 2001.

[16] B. A. Larsen, D. M. Klumpar, G. Gurgiolo, “Correlation between Plasmapause Position and Solar Wind Parameters.” Journal of Atmospheric and Solar-Terrestrial Physics, vol. 69, no. 3, pp. 334-340, 2007.

[17] H. A. Elliott, J.-M. Jahn, C. J. Pollock, T. E. Moore, J. L. Horwitz, “O^+ Transport across the Polar Cap.” Journal of Atmospheric and Solar-Terrestrial Physics, vol. 69, no. 13, pp. 1541-1555, 2007.

[18] P. T. Newell, S. Wing, C.-I. Meng, and V. Sigillito, “The Auroral Oval Position, Structure, and Intensity of Precipitation From 1984 Onward: An Automated On-Line Data Base.” Journal of Geophysics Research, vol. 96, no. A4, pp. 5877-5882, 1991.

[19] George Tzanetakis, Georg Essl, and Perry Cook, “Audio Analysis using the Discrete Wavelet Transform.” In. Proc. WSES Int. Conf. Acoustics and Music: Theory and Applications (AMTA 2001), Skiathos, Greece, 2001.

[20] Kenji Kira and Larry A. Rendell, “A Practical Approach to Feature Selection.” In Proceedings of the ninth international workshop on Machine learning, pp. 249-256, Morgan Kaufmann Publisher Inc., 1992.

[21] Lindsay I Smith, “A tutorial on Principal Components Analysis”, http://www.ce.yildiz.edu.tr/personal/songul/file/1097/principal_components.pdf (April 25, 2013), February 26, 2002.

[22] Bishop, C. M., “Pattern Recognition and Machine Learning.” Springer, 2006.

[23] R. O. Duda, P. E. Hart, D. G. Stork, “Pattern Classification.” Second Edition, Wiley-Interscience, 2001.

[24] T. Hastie, R. Tibshirani, J. Friedman, “The Elements of Statistical Learning: Data Mining, Inference, and Prediction.” Springer, 2001.

[25] S. Bickel, C. Sawade, T. Scheffer, “Transfer Learning by Distribution Matching for Targeted Advertising.” In Twenty-Second Annual Conference on Neural Information Processing Systems, pp. 145-152, 2008.

[26] W. Dai, Q. Yang, G. R. Xue, Y. Yu, “Boosting for Transfer Learning.” In Proceedings of the 24th International Conference on Machine Learning, pp. 193-200, 2007.

[27] W. Dai, Y. Chen, G. R. Xue, Q. Yang, Y. Yu, “Translated Learning: Transfer Learning across Different Feature Space.” In Twenty-Second Annual Conference on Neural Information Processing System, pp. 353-360, 2008.

[28] W. Dai, O. Jin, G. R. Xue, Q. Yang, Y. Yu, “EigenTransfer: A Unified Framework for Transfer Learning.” In Proceedings of the 16th International Conference on Machine Learning, pp. 193-200, 2009.

[29] S. J. Pan, Q. Yang, “A Survey on Transfer Learning.” IEEE Trans. Knowledge and Data Eng., vol. 22, no. 10, pp. 1345-1359, 2010.

[30] R. Raina, A. Battle, H. Lee, B. Packer, A. Y. Ng.. “Self-taught Learning: Transfer Learning from Unlabeled Data.” In Proceedings of the 24th International Conference on Machine Learning, pp. 759-766, 2007.

[31] P. Zhao, S. C. H. Hoi, “OTL: A Framework of Online Transfer Learning.” In Proceedings of the 27th International Conference on Machine Learning, pp. 1231-1238, 2010.

[32] M. F. Balcan, A. Blum, K. Yang, “Co-Training and Expansion: Towards Bridging Theory and Practice.” In Proceedings of the Eighteenth Annual Conference on Neural Information Processing Systems, pp. 89-96, 2005.

[33] Y. Baram, R. El-Yaniv, K. Luz, “Online Choice of Active Learning Algorithms.” In Proceedings of the Twentieth International Conference on Machine Learning, vol. 5, pp. 255-291, 2004.

[34] B. Bryan, J. Schneider, “Actively Learning Level-Sets of Composite Functions.” In Proceedings of the Twenty-fifth International Conference on Machine Learning, pp. 80-87, 2008.

[35] R. A. Jacobs, M. I. Jordan, G. E. Hinton, “Adaptive Mixtures of Local Experts.” Neural Computation, vol. 3, No. 1, pp. 79-87, 1991.

[36] S. E Yuksel, J. N. Wilson, P. D. Gader, “Twenty Years of Mixture of Experts.” IEEE Transactions on Neural Networks and Learning Systems, vol. 23, pp. 1177-1193, 2012.