

# **RAMALAN JUMLAH KANDUNGAN ELEKTRON GPS MENGGUNAKAN MESIN VEKTOR SOKONGAN**

HARSHINI A/P JAYASANGAR  
PROF. MADYA DR. ZALINDA BINTI OHMAN

Fakulti Teknologi dan Sains Maklumat, Universiti Kebangsaan Malaysia

## **ABSTRAK**

Ramalan Cuaca Angkasa menggabungkan penyelidikan sains ruang dengan pengembangan aplikasi praktikal untuk kepentingan aktiviti manusia yang dipengaruhi oleh persekitaran ruang angkasa. Ionosfera adalah bahagian atmosfera atas yang mempengaruhi penyebaran gelombang radio. Jumlah Kandungan Elektron (TEC), didefinisikan sebagai jumlah elektron sepanjang jalur isyarat antara satelit GPS dan penerimanya iaitu, kandungan ionisasi ionosfera. Data TEC yang dikumpul daripada GPS adalah untuk membina model empirikal. Signal GPS dipengaruhi oleh ionosfera semasa perjalanan daripada satelit GPS ke penerimanya di darat. Dengan kemajuan dalam algoritma peramalan TEC, kesan cuaca ruang ionosfera pada komunikasi radio dan sistem navigasi dapat dikenal pasti terlebih dahulu untuk memberi amaran dan mengurangkan kesannya. Kajian ini dijalankan untuk membangunkan model peramalan nilai TEC dengan menggunakan algoritma teknik perlombongan data. Teknik perlombongan data yang digunakan adalah teknik pengelasan. Data-data dari 2010 hingga 2012 telah dikumpulkan oleh Pusat Sains Angkasa Universiti Kebangsaan Malaysia dan digunakan dalam kajian ini. Data yang dikumpul adalah berbentuk siri masa yang berdimensi tinggi dan berjumlah besar. Ketinggian dimensi siri masa merendahkan efisiensi perlombongan data. Empat fasa yang terlibat dalam kajian ini ialah fasa pengumpulan data, fasa pemprosesan data, fasa pembinaan model dan fasa penilaian corak keputusan kajian. Bagi mengatasi ketinggian dimensi siri masa, pendekatan pengurangan data Piecewise Aggregate Approximation (PAA) dan Symbolic Aggregate Approximation (SAX) digunakan untuk prapemprosesan data. Perlombongan data dijalankan dengan menggunakan teknik pengelasan iaitu Mesin Sokongan Vektor. Ukuran prestasi bagi model peramalan yang digunakan dalam kajian ini dinilai untuk mendapatkan hasil ramalan yang sesuai.

## **1 PENGENALAN**

Ramalan Cuaca Angkasa adalah bidang yang agak baru yang menggabungkan ruang penyelidikan sains dengan pengembangan aplikasi praktikal untuk kepentingan aktiviti manusia yang dipengaruhi oleh persekitaran ruang angkasa. Ionosfera adalah bahagian atas atmosfera yang dapat mempengaruhi penyebaran gelombang radio. Jumlah Kandungan Elektron (TEC), yang ditakrifkan sebagai jumlah elektron sepanjang jalur isyarat di antara satelit GPS dan penerima, menerangkan kandungan ionisasi ionosfera.

Data TEC yang berasal dari sistem kedudukan global (GPS) telah dikumpulkan untuk membina model empirikal. Isyarat GPS kebanyakannya dipengaruhi oleh ionosfera semasa perjalanan dari satelit GPS ke penerimanya di darat.

Dengan kemajuan dalam algoritma peramalan TEC berdasarkan pemerhatian GPS, kesan cuaca ruang ionosfera pada komunikasi radio dan sistem navigasi dapat dikenal pasti terlebih dahulu untuk memberi amaran dan mengurangkan kesannya. Dalam kajian ini model Mesin Vektor Sokongan digunakan untuk meramalkan nilai TEC.

## **2 PENYATAAN MASALAH**

Perkhidmatan GPS dipengaruhi oleh cuaca ruang ionosfera. GPS kini merupakan salah satu sistem penentuan kedudukan satelit global yang paling popular kerana mempunyai isyarat dan prestasi global. Ini adalah sistem radio navigasi berasaskan satelit yang digunakan untuk mengesahkan kedudukan dan waktu di angkasa dan di Bumi. Isyarat komunikasi radio GPS terpengaruh kerana ionosfera Bumi yang selalu dipengaruhi oleh

sinaran matahari dan medan geomagnetik. Pemantauan kesan ionosfera pada sistem satelit GPS telah mendapat perhatian penting untuk melindungi hubungan komunikasi radio semasa fenomena cuaca ruang angkasa yang buruk. Pengubahan ionosfera dapat dijelaskan secara kuantitatif dengan nilai TEC yang diukur oleh GPS. Pelbagai proses geo-fizikal dan algoritma matematik menggunakan kaedah saintifik berdasarkan pemerhatian penderiaan jauh. Walau bagaimanapun, algoritma pemodelan dan ramalan ancaman ionosfera untuk perkhidmatan GPS masih mencabar untuk menyokong pergantungan pada aplikasi teknologi moden.

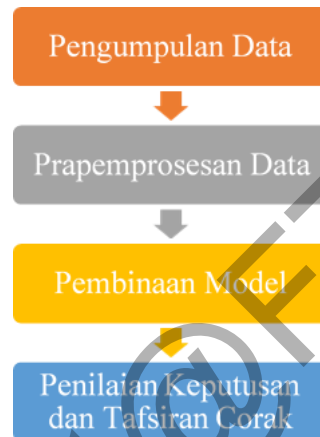
## **3 OBJEKTIF KAJIAN**

Berikut merupakan objektif yang ingin dicapai melalui projek ini:

- a. Untuk meramalkan Kandungan Total Elektron (TEC) menggunakan salah satu kaedah pembelajaran mesin.

## 4 METOD KAJIAN

Metodologi kajian perlu dirangka untuk memastikan kajian ini berjalan dengan lancar dan memenuhi keperluan spesifikasi kajian. Rajah merujuk kepada fasa-fasa perlombongan data yang dijalankan iaitu fasa pengumpulan data, fasa prapemprosesan data, fasa pembinaan model dan fasa penilaian keputusan dan tafsiran corak kajian.



Rajah 1.1 Fasa perlombongan data

### 4.1 Fasa Perancangan

#### a) Pengumpulan Data

Sebilangan besar kandungan Elektron dan data siri masa dikumpulkan masing-masing dari tahun 2010 hingga 2012 dalam format excel. Data yang digunakan adalah siri masa yang bermula dari tahun 2010 hingga 2012.

### 4.2 Fasa Prapemprosesan Data

#### a) Pembersihan Data

Pembersihan data dijalankan di fasa ini untuk meminimakan data yang tidak konsisten, mengisi data yang hilang, mannganti data yang bertindih atau membuang data yang tidak relevan untuk meningkatkan kualiti data. Prapemprosesan data harus dilakukan supaya boleh meningkatkan kualiti dan kuputusan data yang tepat.

b) Transformasi Data

Siri masa pengumpulan data akan menjalani pengurangan dimensi untuk menghasilkan siri masa yang lebih tepat dan untuk perlombongan data.

#### **4.3 Fasa Pembinaan Model**

Pendekatan perlombongan data dijalankan di fasa ini dengan menggunakan teknik algoritma pada data. Teknik pembelajaran mesin yang digunakan adalah Mesin Sokongan Vektor (SVM). Selepas mengaplikasikan algoritma tersebut, satu model akan dibina daripada kumpulan data dan algoritma tersebut.

#### **4.4 Fasa Pentafsiran Corak dan Keputusan**

Model pembangunan perlombongan data akan ditafsirkan atau dipersembahkan menggunakan teknik visual untuk dikaji.

### **5 HASIL KAJIAN**

Pemprosesan data yang merupakan teknik perlombongan data yang melibatkan mengubah data mentah menjadi format yang dapat difahami. Bab ini juga menunjukkan representasi pengurangan data melalui PAA, representasi dan melalui SAX dan pengelasan rentetan perkataan SAX.

Bab ini juga akan membincangkan mengenai proses pembangunan, implementasi dan pengujian dalam kajian ini. Dalam kajian ini, teknik algoritma SVM digunakan sebagai model peramalan nilai TEC selepas pengurangan dimensi terhadap siri masa TEC dijalankan.

Selain itu juga, bab ini bertujuan untuk menentukan corak nilai TEC menggunakan model SVM seperti yang telah dirancang pada fasa sebelum ini. Perancangan yang dilaksanakan dapat membantu dalam memenuhi matlamat dan objektif yang ditetapkan pada peringkat awal.

## 5.1 Perbincangan Pemrosesan Data

Pemrosesan data menyediakan data mentah untuk pemrosesan selanjutnya. Terdapat beberapa fasa dalam pemrosesan data iaitu, pembersihan data, integrasi data, transformasi data, pengurangan data, diskretasi data dan persampelan data. Dalam pemrosesan data ini, terdapat beberapa langkah. Langkah pertama ialah pengimporan libraries. Seterusnya, pengimporan set data. Langkah ketiga ialah mengendalikan nilai hilang dalam set data yang diimportkan. Langkah keempat ialah memisahkan kumpulan data menjadi Set Latihan dan Ujian. Langkah yang terakhir ialah penskalaan ciri.

### Langkah 1: Pengimporan Libraries

```
[ ] import pandas
import numpy
import matplotlib.pyplot as plt
```

Rajah 4.1 Pengimporan Libraries.

- NumPy adalah pakej asas untuk pengkomputeran saintifik dengan Python. Ia mengandungi antara lain, objek array N-dimensi yang kuat, fungsi (penyiaran) yang canggih, alat untuk mengintegrasikan kod C / C ++ dan FORTRAN, Algebra linear, transformasi Fourier, dan keupayaan nombor rawak.
- Pandas adalah untuk manipulasi dan analisis data. Pandas adalah sumber terbuka, perpustakaan berlesen BSD yang menyediakan struktur data berprestasi tinggi, mudah digunakan dan alat analisis data untuk bahasa pengaturcaraan Python.
- Matplotlib dapat digunakan dalam skrip Python, shell Python dan IPython, notebook Jupyter, pelayan aplikasi web, dan empat alat alat antara muka pengguna grafik. Seaborn adalah perpustakaan visualisasi data Python berdasarkan matplotlib.

### Langkah 2: Pengimporan Set Data

```
▶ from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

Rajah 4.2 Pengimportan data dalam Google Colab.

```

inputData = list()

for filename in filenames:
    filepath = r'/content/drive/MyDrive/Colab Notebooks/2012/{0}.std'.format ( filename )
    data = pandas.read_csv ( filepath, None, ' ', None, ['Time', 'Satellite', 'TEC'], engine='python' )
    datas = data.groupby ( 'Time' ) [ 'TEC' ].mean()
    #times = data.index.to_list()
    #TECs = data.to_list()

    impFit = list()# used for train our data for modelling
    impTest = list() #used for testing the model for prediction

    for timeIndex in range ( 60, 86401, 60 ):
        try:
            value = [ timeIndex, datas [ timeIndex ] ]
            impFit.append ( value )
            impTest.append ( value )
        except KeyError as e:
            impTest.append ( [ timeIndex, numpy.nan ] )

    imp.fit ( impFit )
    datas = imp.transform ( impTest )

```

Rajah 4.3 Pengimportan Set data

Dengan menggunakan Pandas, mengimport set data dan fail yang telah digunakan di sini adalah fail .std. Selepas pengimportan set data, menggunakan fungsi head(). Fungsi ini mengembalikan baris n pertama untuk objek berdasarkan kedudukan. Ia digunakan untuk menguji set data dengan cepat jika objek mempunyai jenis data yang tepat di dalamnya (Secara default mengembalikan 5 baris). Set data menunjukkan lima baris yang pertama yang mengandungi Stesen dan tarikh (UKMBxxx-yyyy-mm-dd.std), Masa (dalam saat) dan nilai TEC (dalam TECU).

### Langkah 3: Mengendalikan Nilai Hilang dalam Set Data

Konsep nilai yang hilang penting untuk difahami agar berjaya menguruskan data. Sekiranya nilai yang hilang tidak ditangani dengan betul, maka dia akhirnya dapat membuat kesimpulan yang tidak tepat mengenai data tersebut. Oleh kerana pengendalian yang tidak betul, hasil yang diperoleh akan berbeza dengan yang terdapat nilai yang hilang.

```

Create imputer to predict missing data values

[ ] from sklearn.experimental import enable_iterative_imputer
    from sklearn.impute import IterativeImputer

    imp = IterativeImputer ( max_iter=10, random_state=0 )

```

Rajah 4.4 Menangani nilai yang hilang.

Untuk menangani data yang hilang, pengiraan min, median atau mod ciri boleh digunakan. Ini adalah pendekatan yang dapat menambahkan perbezaan pada set data. Tetapi kehilangan data dapat ditolak dengan kaedah ini yang memberikan hasil yang lebih baik dibandingkan dengan penghapusan baris dan lajur. Nilai min telah digunakan untuk menggantikan nilai yang hilai dalam atribut Masa (dalam saat) iaitu 43213.205761.

#### Langkah 4: Memisahkan Kumpulan Data menjadi Set latihan dan Ujian

Ini merupakan model algoritma dari yang diimport untuk membuat ramalan. Secara amnya set data dibahagikan menjadi nisbah 80:20, 80 persen data dilatih dan 20 persen data untuk diuji. Namun, Pemisahan ini dapat bervariasi sesuai dengan bentuk dan ukuran kumpulan data.

#### Langkah 5: Penskalaan Ciri

Penskalaan Ciri adalah kaedah untuk membatasi julat pemboleh ubah sehingga dapat dibandingkan dengan alasan umum.

```
[ ] features = list() #Xtrain
target = list() #Ytrain

for rowIndex in range ( len ( inputData ) ):
    featureTime = 0
    featureMonth = months [ rowIndex ]

    row = inputData [ rowIndex ]

    for index in range ( len ( row ) ):
        featureTime += 30
        features.append ( [ featureTime, featureMonth ] )
        target.append ( inputData [ rowIndex ] [ index ] )

features = numpy.array ( features ) # using two features, which are the time of a day(calcul
'''the reason why im using time n month in my feature variable is because the graph for
the predicted TEC values can be different at different months.'''
target = numpy.array ( target ) #the target is the TEC SAX value from (0-6) (with 7 level)
```

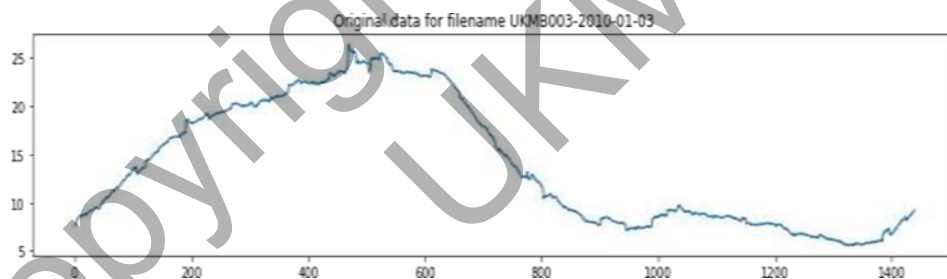
Rajah 4.5 Penskalaan ciri.

Rajah 4.6 menunjukkan penggunaan penskalaan ciri, model mesin tidak berdasarkan jarak Euclidean, masih perlu melakukan penskalaan ciri kerana algoritma akan berkumpul lebih cepat dan efisien.

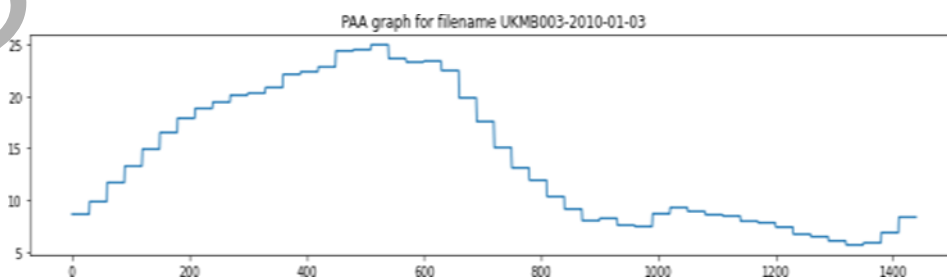
## 5.2 Perbincangan Representasi Data Melalui PAA

Nilai TEC pada setiap stesen pencerapan direkod pada selang 60 saat dan dihantar kepada GISTM. Terdapat lebih kurang 6400 set data yang dikumpulkan oleh alat pencerapan setiap hari dan bilangan set data mencapai 800000 set data dalam masa setahun (andaian data sempurna). Oleh itu, pendekatan Piecewise Aggregate Approximation (PAA) digunakan untuk memampatkan ketinggian dimensi data. PAA mampu mengekalkan data atau bentuk siri masa selepas proses pengurangan melalui lower bounding.

Eksperimen dijalankan terhadap data antara tahun 2010 hingga 2012 yang dikumpulkan dari beberapa stesen satelit. Graf siri masa nilai TEC akan dibandingkan dengan graf PAA untuk memastikan ketepatan pengurangan dimensi. Pemilihan bilangan segmen atau titik data penting dalam memberi representasi graf yang lebih lanjut. Dalam konteks ini, 48 titik data dipilih untuk menghasilkan graf PAA. Representasi PAA stesen satelit ditunjukkan secara min harian iaitu daripada tahun 2010 hingga tahun 2012. Rajah 4.6 hingga Rajah 4.14 menunjukkan perbandingan siri masa asal dengan PAA bagi ketiga-tiga tahun tersebut.

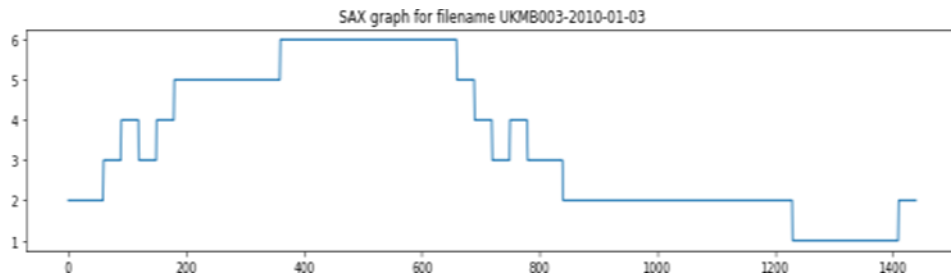


Rajah 4.6 Graf siri masa asal min harian bagi tahun 2010.

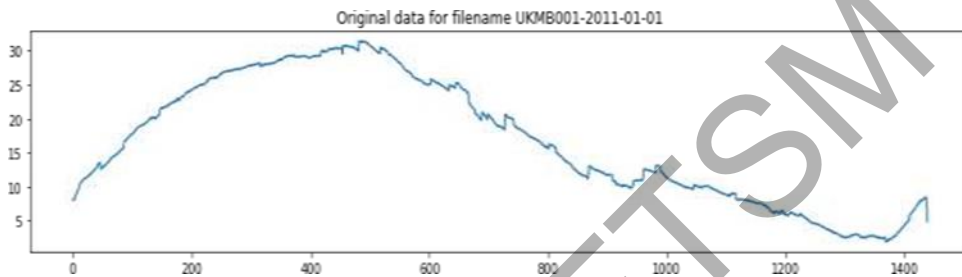


Rajah 4.7 Graf siri masa PAA min harian bagi tahun 2010.

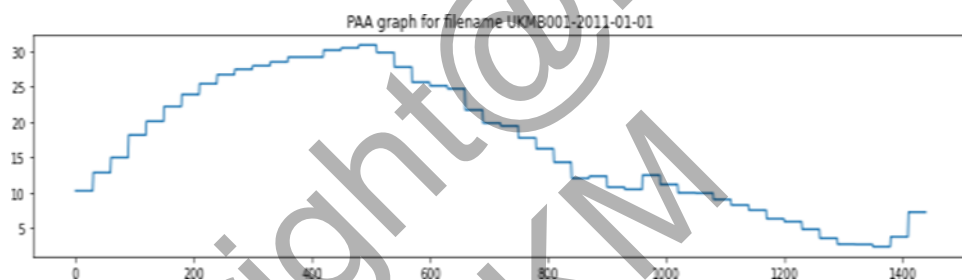




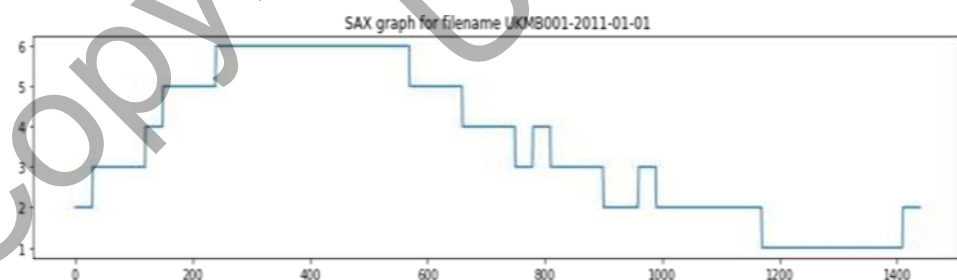
Rajah 4.8 Graf siri masa SAX min harian bagi tahun 2010



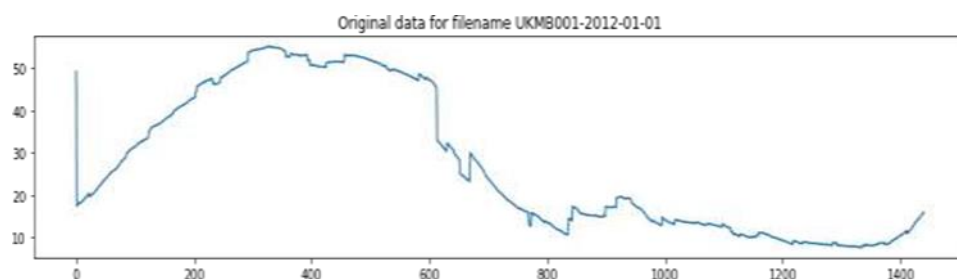
Rajah 4.9 Graf siri masa asal min harian bagi tahun 2011



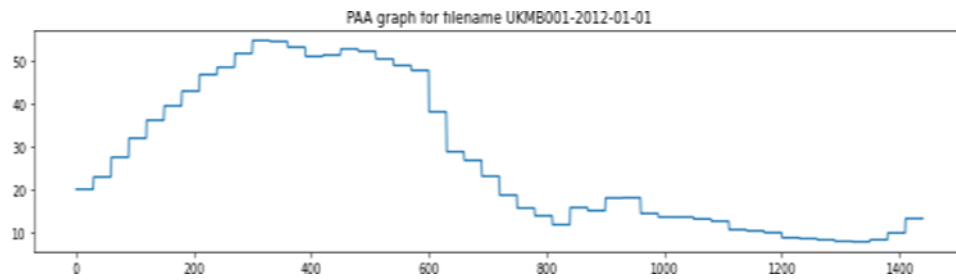
Rajah 4.10 Graf siri masa PAA min harian bagi tahun 2011



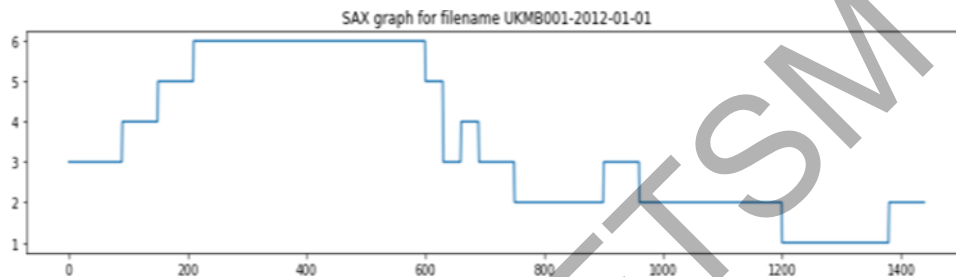
Rajah 4.11 Graf siri masa SAX min harian bagi tahun 2011



Rajah 4.12 Graf siri masa asal min harian bagi tahun 2012



Rajah 4.13 Graf siri masa PAA min harian bagi tahun 2012



Rajah 4.14 Graf siri masa SAX min harian bagi tahun 2012

Berdasarkan pemerhatian daripada perbandingan graf ketiga-tiga tahun, pengurangan dimensi siri masa melalui pendekatan PAA adalah berjaya mengekalkan bentuk siri masa asal. Pengurangan dimensi data harian sebanyak 7000 hingga 10000 data kepada 48 set data (pengurangan sebanyak 145% hingga 208%) tanpa menjejaskan integriti data. Pengurangan data melalui transformasi SAX memudahkan siri masa diproses dengan masa yang lebih singkat dan meningkatkan efisien perlombongan data.

### 5.3 Perbincangan Perbandingan Model-model Menggunakan Teknik SVM.

Tiga jenis model digunakan iaitu, linear, polynomial dan rbf untuk menilai ketepatan peramalan dalam kajian ini. Ukuran prestasi akan dinilai melalui Root Mean Squared Error (RMSE).

#### 5.3.1 Pemodelan Menggunakan Teknik SVM

Dalam algoritma SVM, setiap item data diplotkan sebagai titik dalam ruang n-dimensi (di mana n adalah jumlah ciri yang dimiliki) dengan nilai setiap ciri menjadi nilai koordinat tertentu. Kemudian, klasifikasi dilakukan dengan mencari hipersatah yang membezakan

kelas-kelas dengan lebih baik dan efisien.

### **Membina ciri dan susunan sasaran yang digunakan untuk melatih pengelasan SVM.**

Untuk pemodelan dan klasifikasi model SVM telah digunakan. Pemboleh ubah *features* dan *target* (*array*) digunakan untuk melatih set data menggunakan pengelasan SVM. *features* merupakan min harian yang diambil untuk setiap 30 minit (30,60,90,120,...,1440). Manakala, *target* merupakan nilai TEC SAX yang diperolehi daripada hasil nilai SAX sebelumnya, iaitu simbol nilai 1 hingga 6.

```
[ ] features = list() #Xtrain
target = list() #Ytrain

for rowIndex in range ( len ( inputData ) ):
    featureTime = 0
    featureMonth = months [ rowIndex ]

    row = inputData [ rowIndex ]

    for index in range ( len ( row ) ):
        featureTime += 30
        features.append ( [ featureTime, featureMonth ] )
        target.append ( inputData [ rowIndex ] [ index ] )

features = numpy.array ( features ) # using two features, which are the time of a day(calcul
'''the reason why im using time n month in my feature variable is because the graph for
the predicted TEC values can be different at different months.'''
target = numpy.array ( target ) #the target is the TEC SAX value from (0-6) (with 7 level)
```

Rajah 4.15 Membina ciri dan susunan yang digunakan untuk melatih pengelasan SVM

### **Membina grid untuk menggambarkan hasil**

Untuk memplotkan data, perlu menentukan nilai x dan y. Dalam kes ini, untuk mendapatkan nilai x dan y, Langkah pertama adalah perlu mendapatkan nilai minima dan maksima. Kemudian menentukan langkah-langkah yang seterusnya:

- i. Kod ini merupakan kod yang digunakan untuk mencari nilai minima dan maksima untuk x dan y.

$$x\_min, x\_max = features [ :, 0 ].min() - 1, features [ :, 0 ].max() + 1$$

$$y\_min, y\_max = features [ :, 1 ].min() - 1, features [ :, 1 ].max() + 1$$

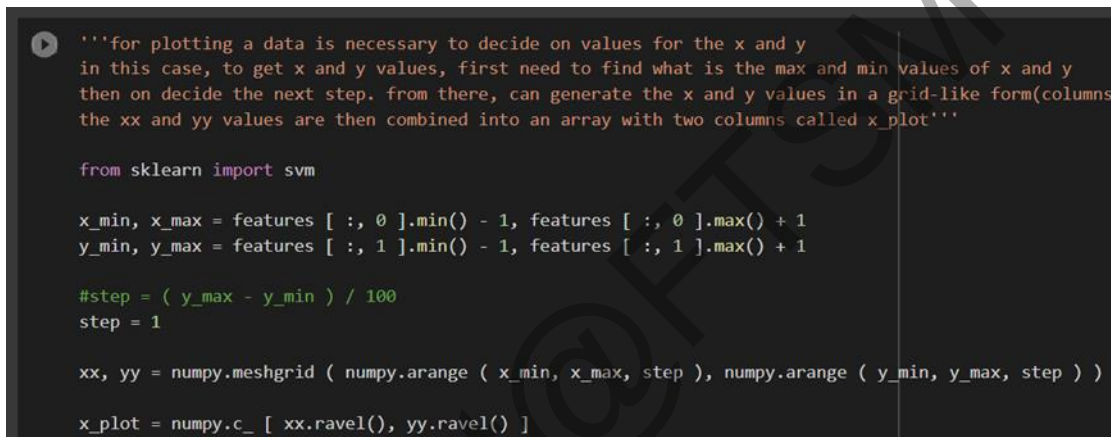
- ii. Seterusnya, kod sumber daripada pengaturcaraan Python adalah untuk

mendapatkan nilai x dan y dalam bentuk grid.

```
xx, yy = numpy.meshgrid( numpy.arange( x_min, x_max, step ), numpy.arange
( y_min, y_max, step ) )
```

- iii. Langkah seterusnya adalah kod Python ini bertujuan untuk menggabungkan nilai xx dan yy ke dalam array dengan dua lajur (column).

```
x_plot = numpy.c_[ xx.ravel(), yy.ravel() ]
```



```
'''for plotting a data is necessary to decide on values for the x and y
in this case, to get x and y values, first need to find what is the max and min values of x and y
then on decide the next step. from there, can generate the x and y values in a grid-like form(columns
the xx and yy values are then combined into an array with two columns called x_plot'''

from sklearn import svm

x_min, x_max = features [ :, 0 ].min() - 1, features [ :, 0 ].max() + 1
y_min, y_max = features [ :, 1 ].min() - 1, features [ :, 1 ].max() + 1

#step = ( y_max - y_min ) / 100
step = 1

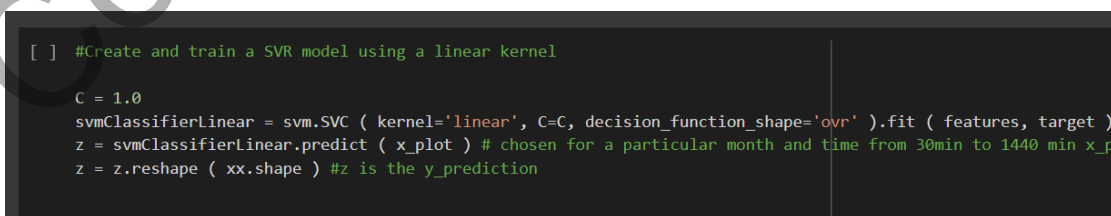
xx, yy = numpy.meshgrid( numpy.arange( x_min, x_max, step ), numpy.arange( y_min, y_max, step ) )

x_plot = numpy.c_[ xx.ravel(), yy.ravel() ]
```

Rajah 4.16 Membina grid untuk menggambarkan hasil.

### Melatih pengelasan SVM dan menggunakannya untuk melakukan ramalan

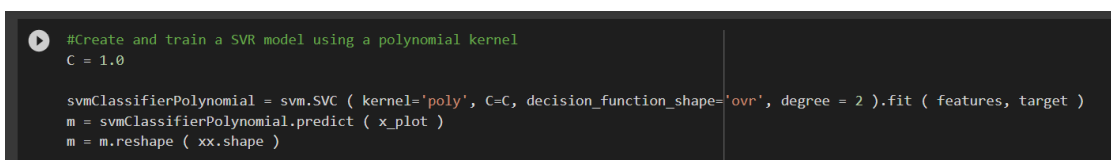
Kod pengelasan SVM dibuat seperti yang ditunjukkan dalam gambar di bawah bertujuan untuk melatih ciri (*features*) dan susunan sasaran (*target*). Dalam kajian ini, tiga jenis model digunakan iaitu, linear, polynomial dan rbf.



```
[ ] #Create and train a SVR model using a linear kernel

C = 1.0
svmClassifierLinear = svm.SVC ( kernel='linear', C=C, decision_function_shape='ovr' ).fit ( features, target )
z = svmClassifierLinear.predict ( x_plot ) # chosen for a particular month and time from 30min to 1440 min x_p
z = z.reshape ( xx.shape ) #z is the y_prediction
```

Rajah 4.17 Melatih pengelasan SVM menggunakan model linear.



```
#Create and train a SVR model using a polynomial kernel
C = 1.0

svmClassifierPolynomial = svm.SVC ( kernel='poly', C=C, decision_function_shape='ovr', degree = 2 ).fit ( features, target )
m = svmClassifierPolynomial.predict ( x_plot )
m = m.reshape ( xx.shape )
```

Rajah 4.18 Melatih pengelasan SVM menggunakan model polynomial.

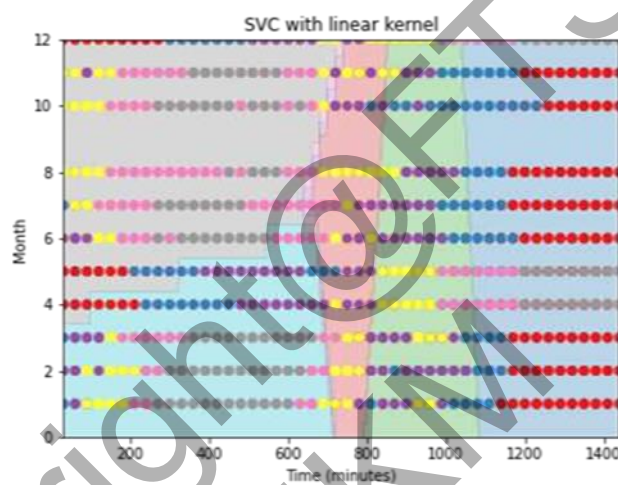
```
[ ] #Create and train a SVR model using a rbf kernel
C = 1.0

svmClassifierRBF = svm.SVC ( kernel='rbf', C=C, decision_function_shape='ovr', degree = 2 ).fit ( features, target )
n = svmClassifierRBF.predict ( x_plot )
n = n.reshape ( xx.shape )
```

Rajah 4.19 Melatih pengelasan SVM menggunakan model rbf.

**Menunjukkan SVM membahagikan nilai TEC SAX berdasarkan waktu (timestamp).**

Kod Python ini digunakan untuk visualisasi nilai TEC SAX, setiap warna ataupun regions menunjukkan kelas kelas yang dibahagikan menggunakan 7 titik data.



Rajah 4.20 SVM membahagikan nilai TEC SAX berdasarkan waktu (*timestamp*)

## 5.4 Proses Peramalan

Proses peramalan akan dijalankan setelah proses pemodelan berjaya dihasilkan bagi meramalkan hasil ramalan yang cekap. Dalam kajian ini teknik SVM digunakan untuk meramalkan data yang dikumpulkan tiga tahun yang berturutan, iaitu 2010, 2011 dan 2012. Proses peramalan ini harus meramalkan nilai TEC SAX untuk tahun yang berterusan daripada hasil yang diperolehi daripada proses pemodelan. Langkah ini amat penting untuk menghasilkan satu kajian yang dapat mencapai objektif kajian justeru menyelesaikan pernyataan masalah yang telah dikenal pasti.

### 5.4.1 Peramalan Menggunakan Teknik SVM

### Graf ramalan plot untuk tahun depan

Dalam proses peramalan ini, jangkaan nilai TEC diplotkan pada hari yang berterusan. Ini merupakan nilai jangkaan yang diramalkan oleh ketiga-tiga jenis model yang digunakan dalam pengelasan SVM.

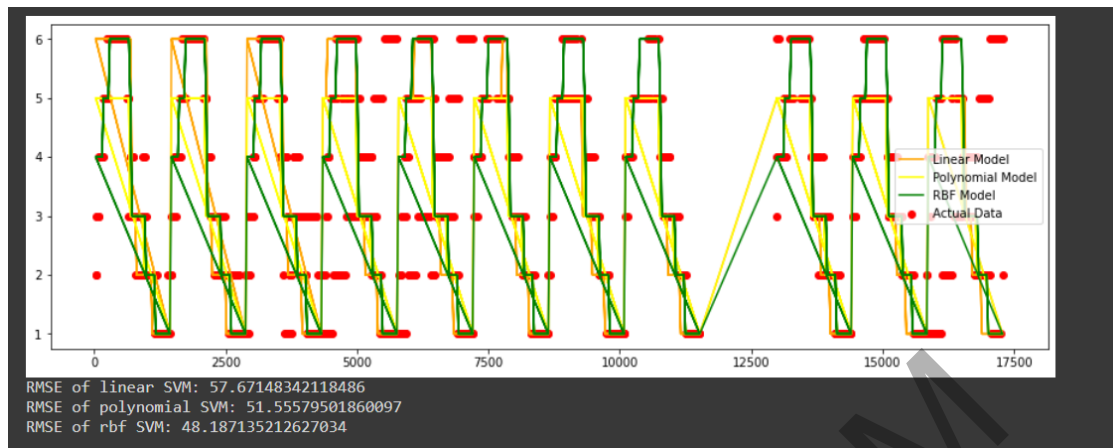


Rajah 4.21 Sumber kod dan output untuk peramalan menggunakan teknik SVM.

Seperti yang ditunjukkan dalam rajah di atas, simbol 6 (titik data) merupakan nilai jangkaan TEC yang tertinggi. Manakala, simbol 1 (titik data) merupakan nilai yang paling rendah tercatat dalam peramalan ini. Dengan ini, peramalan ini menunjukkan agar nilai TEC dijangkakan lebih tinggi pada waktu siang dan jangkaan nilai TEC menjadi semakin rendah pada waktu malam.

### Ukuran prestasi menggunakan *Root Mean Squared Error*.

Ukuran prestasi akan dinilai melalui Root Mean Squared Error (RMSE) untuk mendapatkan ramalan yang terbaik antara ketiga-tiga model tersebut.



Rajah 4.22 Ukuran prestasi dinilai Root Mean Squared Error.

Rajah 4.22 menunjukkan tiga jenis model digunakan untuk menilai ukuran prestasi menggunakan Root mean Squared Error (RMSE). Hasil daripada rajah di atas menunjukkan bahawa model linear mempunyai nilai RMSE 57.67% dan model polinomial memberikan nilai 51.55%. Manakala, nilai RMSE yang diperoleh dari model rbf adalah 48.18%. Semakin rendah nilai RMSE yang diperolehi, semakin baik model tersebut yang diberikan dapat "memuat" set data. Hal ini menunjukkan bahawa model rbf merupakan model yang terbaik dapat memuat set data dalam kajian ini.

## 6 KESIMPULAN

Secara keseluruhannya, bab ini telah merangkumi gambaran keseluruhan kajian ini. Kekangan projek dibincang untuk mengenalpasti kesusahan dalam menjalankan kajian. Perbincangan dijalankan terhadap cadangan masa hadapan yang boleh dilaksanakan untuk menyelesaikan masalah yang dihadapi dalam keberkesanan nilai TEC.

## 7 RUJUKAN

What Is Space Weather? | NASA Space Place – NASA Science for Kids, [Spaceplace.nasa.gov](https://spaceplace.nasa.gov),

- <https://spaceplace.nasa.gov/spaceweather/en/>  
 Space weather forecast: Big storms ahead, Science News for Students,  
<https://www.sciencenewsforstudents.org/article/space-weather-forecast-big-storms-ahead>
- Y. Reuveni, Utilizing ionospheric Total Electron Content (TEC) data for Solar flare predictions using Support Vector Machine (SVM), NASA/ADS,  
<https://ui.adsabs.harvard.edu/abs/2019AGUFMNH33C0930R/abstract>
- H.J. Singer, Space Weather Forecasting: A Grand Challenge,  
<https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/GM125p0023>
- R. Sridharan, A novel method based on GPS TEC to forecast L band scintillations over the equatorial region through a case study,  
<https://www.sciencedirect.com/science/article/abs/pii/S1364682612000405>
- Norsuzila Yaacob, GPS Total Electron Content (TEC) Prediction at Ionosphere Layer over the Equatorial Region, <https://www.intechopen.com/books/trends-in-telecommunications-technologies/gps-total-electron-content-tec-prediction-at-ionosphere-layer-over-the-equatorial-region>
- Using Machine Learning for Time Series Forecasting Project, CodeIT,  
<https://codeit.us/blog/machine-learning-time-series-forecasting>
- Norsuzila Yaacob, GPS Total Electron Content (TEC) Prediction at Ionosphere Layer over the Equatorial Region, <https://www.intechopen.com/books/trends-in-telecommunications-technologies/gps-total-electron-content-tec-prediction-at-ionosphere-layer-over-the-equatorial-region>
- Anshul Sharma, Time Series Data Representation and Dimensionality Reduction Techniques,  
[https://link.springer.com/chapter/10.1007/978-981-15-3357-0\\_18](https://link.springer.com/chapter/10.1007/978-981-15-3357-0_18)
- Tian Han, A pattern representation of stock time series based on DTW,  
<https://www.sciencedirect.com/science/article/abs/pii/S0378437120300157?via%3Dihub>
- An overview of time series forecasting models, Medium, <https://towardsdatascience.com/an-overview-of-time-series-forecasting-models-a2fa7a358fcb>
- Author Understanding code), SVM | Support Vector Machine Algorithm in Machine Learning, Analytics Vidhya, <https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>



CRISP-DM methodology leader in data mining and big data, Medium, <https://towardsdatascience.com/crisp-dm-methodology-leader-in-data-mining-and-big-data-467efd3d3781>

PAA and SAX features — tslearn 0.4.1 documentation, Tsllearn.readthedocs.io, [https://tslearn.readthedocs.io/en/stable/auto\\_examples/misc/plot\\_sax.html](https://tslearn.readthedocs.io/en/stable/auto_examples/misc/plot_sax.html)

Cs.ucr.edu, <https://www.cs.ucr.edu/~eamonn/SAX.pdf>

Ahmad Anis, Easy Guide To Data Preprocessing In Python – Kdnuggets, Kdnuggets, <https://www.kdnuggets.com/2020/07/easy-guide-data-preprocessing-python.html>

Common Time Series Data Analysis Methods and Forecasting Models in Python, Medium, <https://towardsdatascience.com/common-time-series-data-analysis-methods-and-forecasting-models-in-python-f0565b68a3d8>

Vignesh Krishnamoorthy, Symbolic Aggregate Approximation, Vigne.sh, <https://vigne.sh/posts/symbolic-aggregate-approximation/>

Pyts.approximation.PiecewiseAggregateApproximation — pyts 0.11.0 documentation, Pyts.readthedocs.io, <https://pyts.readthedocs.io/en/stable/generated/pyts.approximation.PiecewiseAggregateApproximation.html#pyts.approximation.PiecewiseAggregateApproximation>

Youtube.com, URL:<https://www.youtube.com/watch?v=MmC4b7gPY0Q>

Piecewise Aggregate Approximation — pyts 0.11.0 documentation, Pyts.readthedocs.io, [https://pyts.readthedocs.io/en/stable/auto\\_examples/approximation/plot\\_paa.html](https://pyts.readthedocs.io/en/stable/auto_examples/approximation/plot_paa.html)

Example: Calculate the piecewise aggregate approximation of time series (SPMF - Java), Philippe-fournier-viger.com, <https://www.philippe-fournier-viger.com/spmf/TimeSeriesPiecewiseAggregateApproximation.php>

Author Chonghui Guo, Article title: An Improved Piecewise Aggregate Approximation Based on Statistical Features for Time Series Mining [https://link.springer.com/chapter/10.1007/978-3-642-15280-1\\_23](https://link.springer.com/chapter/10.1007/978-3-642-15280-1_23)

Piecewise Aggregate Approximation (PAA) | SAX-VSM, Website Title: Jmotif.github.io, [https://jmotif.github.io/sax-vsm\\_site/morea/algorithm/PAA.html](https://jmotif.github.io/sax-vsm_site/morea/algorithm/PAA.html)

Machine Learning and RapidMiner Tutorials | RapidMiner Academy, Academy.rapidminer.com, <https://academy.rapidminer.com/learn/video/replace-missing-values-in-time-series>

Handle Missing Values in Time Series For Beginners, Kaggle.com, <https://www.kaggle.com/juejuewang/handle-missing-values-in-time-series-for-beginners>

Facebook Stock Prediction Using Python And SVR, Youtube.com

Klang vally rainfall forecasting model using time series data mining technique, Jatit.org, <http://www.jatit.org/volumes/Vol92No2/18Vol92No2.pdf>Nur'ain Mohamad. 2008. E-manual Buka—dan-Pasang Unit Air Cucle Machine 747 (Komponen Penghawa Dingin Kapal Terbang). Latihan Ilmiah, Fakulti Teknologi dan Sains Maklumat, Universiti Kebangsaan Malaysia.}