Predicting hourly GOES-08 electron flux from solar wind data using neural networks

P. Wintoft

June 2, 2000

Contents

1 Introduction

Here we examine the prediction of GOES-08 electron fluxes from OMNI solar wind data. All data are 1 hour averages. The electron flux are measured at the energy levels >0.6 MeV and >2 MeV. The model will only use the solar wind data for its inputs. This means that for any given time the model will be able to predict the electron flux for any local time.

2 Data

The OMNI data fields selected are: total magnetic field B, B_y- and B_z-magnetic field components in GSM coordinates, particle density n, and velocity V. The OMNI data are one hour averages where each time stamp indicates the beginning of the interval, thus time 0 is the interval [0,1], time 1 the interval [1,2] and so on.

The GOES-08 database currently used extends over the years 1995 to 1999 and contain 5 minute average electron fluxes at the >0.6 MeV and >2 MeV electron levels. The data are averaged into one hour averages with the time stamps also indicating the beginning of an interval.

3 Analysis

3.1 The OMNI data

Over the time 1995 to 1999 the best data coverage is for the period June 1995 to July 1996. During that period there are about 600 data points with 48 hours contiguous data for each month. The maximum number of points per month is about $24 \times 30 = 720$.

3.2 The GOES-08 electron data

3.3 Correlation between OMNI and GOES data

We calculate the linear correlation between two variables for different time shifts. The linear correlation between two variables x and y is

$$r = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2}\sqrt{\sum_i (y_i - \bar{y})^2}}, \quad (1)$$

where $\bar{x}$ is the mean of the x_i's and $\bar{y}$ is the mean of the y_i's.

Calculating the linear correlation between two sets of variables gives an overall feel their relations. The x_i's are one of the OMNI solar wind parameter at time $t_i + \tau$ and the y_i's are one of the GOES electron energies at time t_i. The time shift τ is used to study the delayed response of the GOES data with respect to the OMNI data. The correlation coefficient r becomes a function of τ which is varies over the interval -100 hours to +100 hours. The result is shown in Figure 1.

We see that the instantaneous linear correlation between a solar wind parameter $x(t+\tau)$ and the GOES electron flux $y(t)$ is strongest for the velocity. The correlation peaks at $\tau \approx -25$ hours for the > 0.6 MeV electron flux (upper left panel of Figure 1) and at $\tau \approx -50$ hours for the > 2 MeV electron flux (upper right panel). The density is the second most important and is anticorrelated with a peak at $\tau \approx -10$ hours for the > 0.6 MeV electrons and $\tau \approx -20$ hours. The magnetic field show a very weak correlation to the electron flux.

We can also see from Figure 1 that the correlations are stronger at times when the GOES satellite is in the local noon sector than in the local midnight sector. Typically the noon values are a factor of ?? higher than the midnight values.

To conclude we see that there is clear relationship between the instantaneous solar wind velocity and density with the GOES electron flux. This relation is stronger at local noon than at local midnight. The instantaneous linear correlation between the solar wind magnetic field and the electron flux is weak or non-existant.

4 The model

The time scales on which the magnetosphere responds to the solar wind changes from minutes to days. The shortest time scale that we use here is one hour. To model a dynamic system we need both the current values and past values. The standard approach for a feed forward neural network (FFNN) is to construct a time delay line of the input data. For the prediction of the electron flux the solar wind input data should extend over several days. The problem that occurs is that the data contain data gaps which dramatically reduces the number of available training examples when the time delay is extended over several days. To overcome this problem we approach the solution in two steps: 1) first we model the long-term response (days) of the electron flux using daily average solar wind data, 2) then we introduce the short-term response using hourly solar wind data with a shorter time delay line.

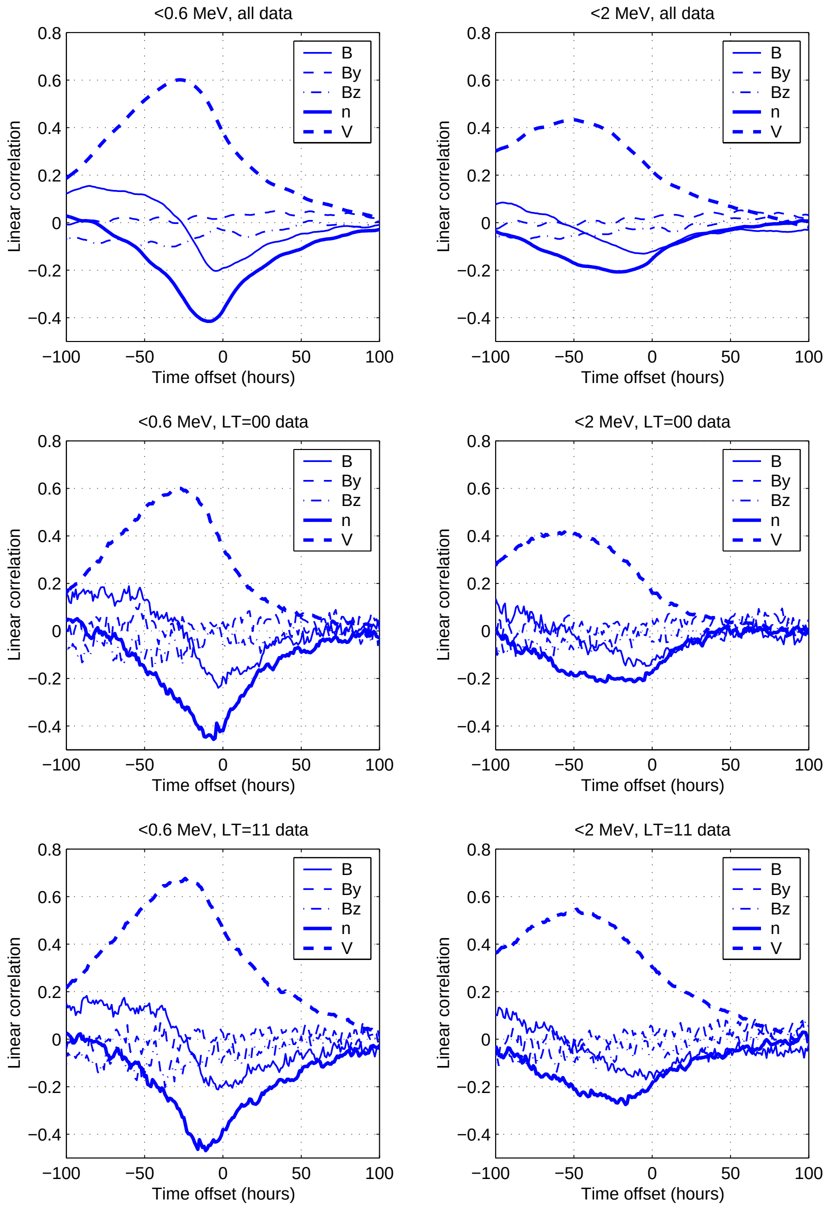

Figure 1: The figure show the linear correlation between the OMNI solar wind data and the GOES electron data for time offsets τ from -100 hours to +100 hours. The top left panel is the >0.6 MeV electron data for all local time sectors, while the bottom left is data for only local time 11. The top right panel is the >2 MeV electron data for all local time sectors, while the bottom right is data for only local time 11.

4.1 The daily variation

First we try to find the optimal network for the prediction of the daily variation. On time scales of days only the density and velocity show a large degree of variation. The B_z component has structures that typically only survive for up to 10 hours, so a daily average will effectively remove the variation and the average will be around zero. The B_y variation is also on time scales of hours. However, the daily average B_y will be some positive or negative value depending on the IMF polarity, and not close to zeros as the B_z component. To cunclude, we see that the daily averages of the density and the velocity are of importance, while the daily average magnetic field is not.

The inputs to the network are the running daily average density and velocity, and the local time of the GOES satellite. The average of one parameter is

$$\langle x \rangle(t) = \frac{1}{24} \sum_{s=-23}^{0} x(t+s), \tag{2}$$

where t and s are in hours and x can be subsistuted for n or V. The local time is coded into a 24 element vector $\mathbf{l} = (l_0, l_1, \ldots, l_{23})$ so that

$$l_i = \begin{cases} 1 & i \leq h < i+1 \\ 0 & \text{otherwise} \end{cases}, \quad 0 \leq i \leq 23, \tag{3}$$

where h is the local time. The coding will thus generate a vector with 23 zeros and a one at the row corresponding to the same local time. To capture the time evolution we create a time delay line of the density and velocity. The input vector thus becomes

$$\mathbf{x}(t; t_1, t_2) = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \langle n \rangle(t+t_1) \\ \vdots \\ \langle n \rangle(t+t_2) \\ \langle V \rangle(t+t_1) \\ \vdots \\ \langle V \rangle(t+t_2) \\ l_0 \\ \vdots \\ l_{23} \end{bmatrix}, \tag{4}$$

where t_1 and t_2 marks the start and end in hours, respectively, of the time delay window. E.g. with $t_1 = -7 \cdot 24$ and $t_2 = -2 \cdot 24$ means that the time window extends over 6 days with the last used input is 2 days back, thus producing 2 day ahead predictions.

The output of the model is the GOES-08 $>$ 0.6 Mev or $>$ 2 MeV electron flux. The mapping from input $\mathbf{x}$ to output y is

$$y(t) = F(\mathbf{x}(t; t_1, t_2)) + r(t). \tag{5}$$

The optimization of the neural network F consists of finding the number of hidden neurons, the weight values, and the input time window t_1 and t_2 so that the residual $r(t)$ is minimized. The network has the following form

$$F(\mathbf{x}) = \sum_{i=1}^{m} v_i \tanh(\sum_{j=1}^{n} w_{ij} x_j + b_i) + a, \tag{6}$$

where v_i and w_{ij} are the weights, and a and b_i are the biases. The network has m hidden neurons and n inputs. Thus, the network has a linear output transfer function and a tanh hidden transfer function.

When the optimal network, using daily solar wind density and speed as inputs, has been determined we can use this information to proceed with the hourly input data.

4.2 The hourly variation

We expect that also the hourly variation of the solar wind will influence the evolution of the magnetospheric electron flux. From the previous section we have removed the slow variation of the electron flux that can be associated with the daily average solar wind density and velocity.

The second network is trained to learn the residual from the first network (Equation 5)

$$r(t) = G(\mathbf{z}(t)) + e(t) \tag{7}$$

where $e(t)$ is the final error. The input is

$$\mathbf{z}(t) = \begin{bmatrix} z_1 \\ \vdots \\ z_q \end{bmatrix} = \begin{bmatrix} n(t+s_1) \\ \vdots \\ n(t+s_2) \\ V(t+s_1) \\ \vdots \\ V(t+s_2) \\ B_z(t+s_1) \\ \vdots \\ B_z(t+s_2) \\ l_0 \\ \vdots \\ l_{23} \end{bmatrix}, \tag{8}$$

where now the solar wind parameters are hourly averages, and s_1 and s_2 mark the start and end times, respectively, of the input window. The network G has the same form as network F in Equation 6, thus

$$G(\mathbf{z}) = \sum_{i=1}^{p} v_i \tanh(\sum_{j=1}^{q} w_{ij} z_j + b_i) + a. \quad (9)$$

The weights v and w are of course different from those in Equation 6. The network has p hidden neurons and q input neurons. The final output of the two networks is

$$y(t) = F(\mathbf{x}(t)) + G(\mathbf{z}(t)) + e(t). \quad (10)$$

5 Preprocessing

5.1 Data gaps

5.2 Training, validation, and test sets

5.3 Normalization

The solar wind data typically have a range of values that varies over a magnitude or less. Typical ranges for the density are 0 to 50 cm^{-2}, velocity 250 to 800 kms^{-2}, B_z component ± 10 nT. The data are normalized to lie in the range ± 0.8 to fit the range of the tanh function.

The electron flux, on the other hand, show a large degree of variation from 10^{-1} to 10^5 $\mathrm{cm}^{-2}\mathrm{s}^{-1}\mathrm{sr}^{-1}$ for the > 0.6 MeV electrons and 10^{-1} to 10^4 $\mathrm{cm}^{-2}\mathrm{s}^{-1}\mathrm{sr}^{-1}$ for the > 2 MeV electrons. Transforming the > 2 MeV electrons from the range $[0, 10^4]$ to the range $[-0.8, +0.8]$ will result in a network that can predict the high values but not the low. E.g., assume that the network has an error of 5%, then the error will be 500 and the low values will be totally neglected. To solve this we may instead take the logarithm of the e-flux. Both the low and high values will be modelled, however, the high values will be slightly worse modelled as compared to the linear transform. A third approach could be to use a semilogarithmic scaling. Our approach here will be the use of the logarithmic transformation.

The normalization of the input and output data are summarized in Table 1.

Table 1: The normalization of the input and output data. The table shows for each parameter whether linear or logarithmic transformation is used. The Low value is scaled to -0.8 and the High value to +0.8.

Parameter	Transformation	Low	High
n	linear	0	40
V	linear	250	800
B_z	linear	-10	10
> 0.6 MeV	logarithmic	?	?
> 2 MeV	logarithmic	10^{-1}	10^{5}

6 Results

First, we find optimal time delay window that is needed using daily average solar wind density and velocity. Several different networks are trained by varying the start and end times of the time delay window, and varying the number of hidden neurons. Then, after the optimal daily model has been found, the hourly data are introduced. As for the daily models, the optimal input window and optimal number of hidden neurons are determined. The two models are then combined to produce the final predictions.

6.1 The > 0.6 MeV electron flux model

6.2 The > 2 MeV electron flux model

We explore the parameter space by letting t_1 and t_2 be varied from -10 days to 0 days. We also let the number hidden neurons m attain the values 3, 5, and 10. For each combination of t_1, t_2, and m a network is trained on the data in the training set. The performance of the network is continuosly monitored using the validation set, and when the validation set error starts increasing the training is stopped. The validation RMS errors and correlations are shown in Figure 2 and Figure 3. Each dot represents the validation error or correlation calculated from each network. The upper panels show the results as a function of t_1 and the lower panels as a function of t_2. The network that gives the minimum error and maximum correlation is marked with a circle. For each t_1 in the figures the range of dots represents different combinations of t_2 and m.

The optimal network (circle) has $t_1 = -8$ days, $t_2 = 0$ days, and $m = 10$. However, for a real-time application this model would only produce nowcasting as $t_2 = 0$. If we instead choose the network with $t_2 = -1$ day we are able to forecast one day ahead. From the figures we also see that

Table 2: Summary of the results for the optimal network (NET 1) and the second optimal network (NET 2). The root-mean-square error (RMSE) and the linear correlation (CORR) are calculated for the training, validation, and test sets. The RMSE has units of electrons$\text{cm}^{-2}\text{s}^{-1}\text{sr}^{-1}$.

Network	t_1	t_2	m	RMSE	CORR	Data set
NET 1	-8	0	10	$1.09 \cdot 10^3$	0.843	Training
				$1.32 \cdot 10^3$	0.789	Validation
				$1.24 \cdot 10^3$	0.806	Test
NET 2	-9	-1	5	$1.09 \cdot 10^3$	0.842	Training
				$1.33 \cdot 10^3$	0.786	Validation
				$1.31 \cdot 10^3$	0.786	Test

the networks at $t_2 = -1$ and $t_2 = 0$ gives similar results. The root-mean-square errors (RMSE) and linear correlations (CORR) for the two networks are summarized in Table 2.

We thus use the NET 2 model and now introduce the hourly solar wind plasma and magnetic field data.

7 Discussion

8 Conclusion

References

[1] Blake, J.B, D.N. Baker, N. Turner, K.W. Ogilvie, and R.P. Lepping, Correlation of changes in the outer-zone relativistic-electron population with upstream solar wind and magnetic field measurements, *Geophys. Res. Lett.*, *24*, 927–929, 1997.

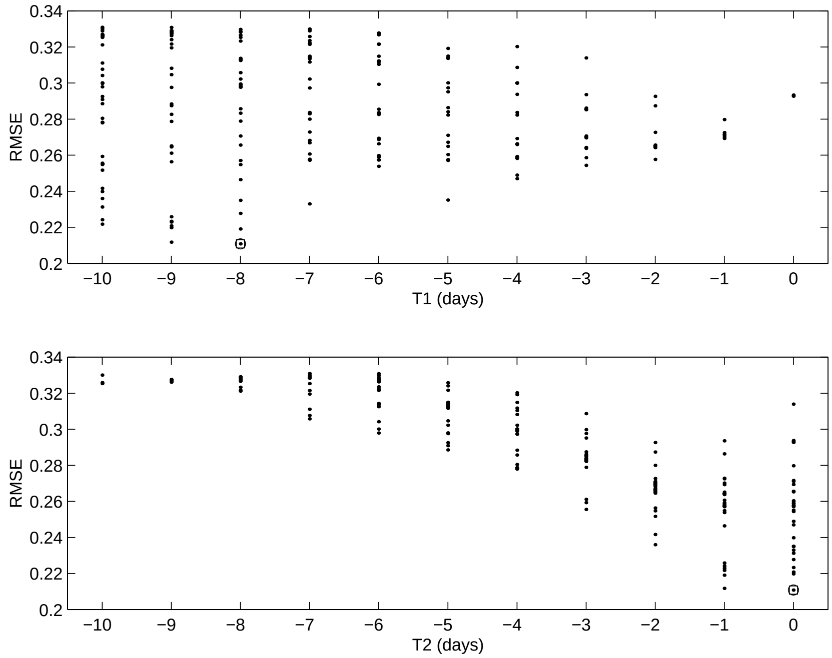

Figure 2:

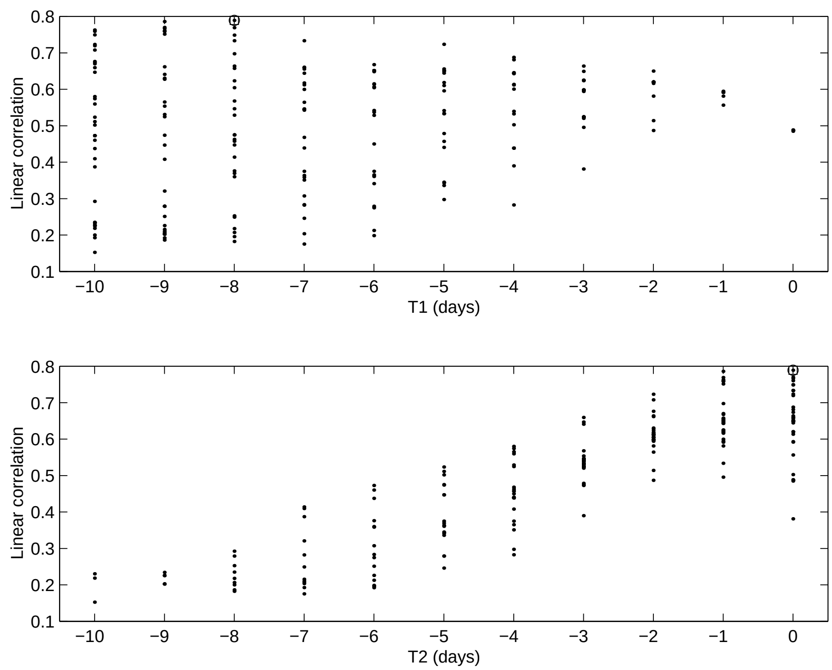

Figure 3: