

visualisation_est_1

March 29, 2020

1 Visualisation of estimations in Group A

Done on LDPRF 2098.

1.0.1 Import necessary libraries

```
[1]: import numpy as np
import pandas as pd
import copy
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers

# import codebase
import thermalModel_main as tmm
import thermalModel_groupB as tm_gb

import importlib
importlib.reload(tmm)
importlib.reload(tm_gb)
```

Using TensorFlow backend.

```
[1]: <module 'thermalModel_groupB' from
'C:\\Users\\user\\Anaconda3\\lib\\thermalModel_groupB.py'>
```

1.1 ANN Ah Model

1.1.1 Data loading and cleaning

```
[2]: df = tm_gb.load_csv(filename = 'LDPRF_2098.csv',
                        features_list = ['runtime_s', 'AhCha', 'AhDch', 'Amb', 'Temp'],
                        mode = 0)
```

```
C:\\Users\\user\\Anaconda3\\lib\\thermalModel_groupB.py:47: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame
```

See the caveats in the documentation: <http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy>

```
df['second'][set_index[index]:set_index[index+1]] =  
df['second'][set_index[index]:set_index[index+1]] + second_increment[index]  
C:\Users\user\Anaconda3\lib\thermalModel_groupB.py:49: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame
```

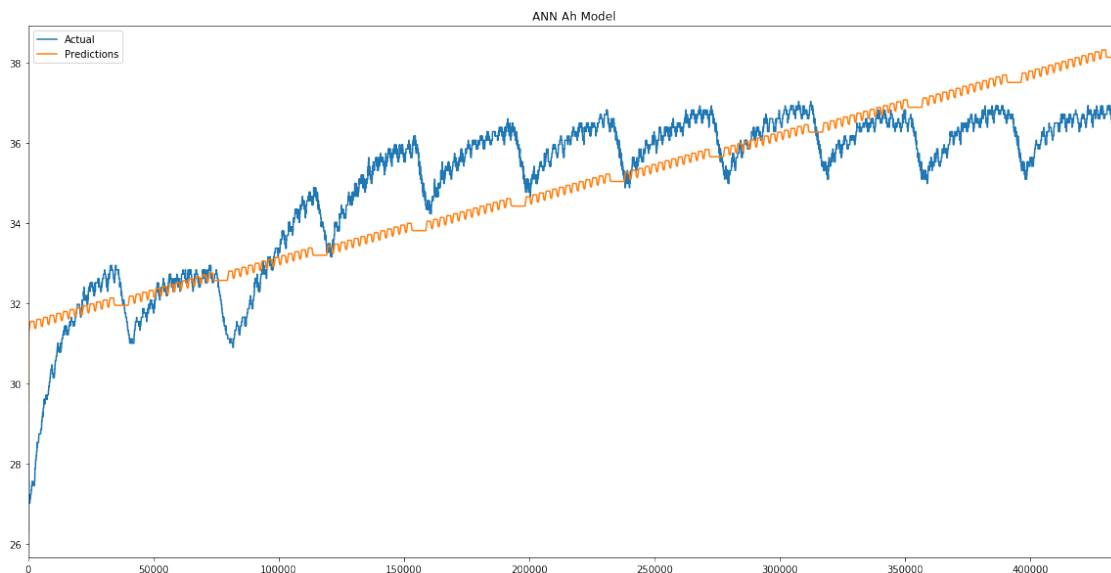
See the caveats in the documentation: <http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy>

```
df['second'][set_index[index]:] = df['second'][set_index[index]:] +  
second_increment[index]  
C:\Users\user\Anaconda3\lib\thermalModel_groupB.py:56: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame
```

See the caveats in the documentation: <http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy>

```
df['second'][set_index[index]:] = df['second'][set_index[index]:] +  
seconds_summation[index]
```

```
[3]: df = df.drop(columns=['runtime_s'])  
  
ML_model = tf.keras.models.load_model('ANN_0.9_Ah_model_1' + '.h5')  
_ = tmm.predictions_plot(model_name = "ANN Ah Model",  
                           dataframe_entry = df,  
                           model = ML_model,  
                           window_size = 1,  
                           plot_errors = True)
```

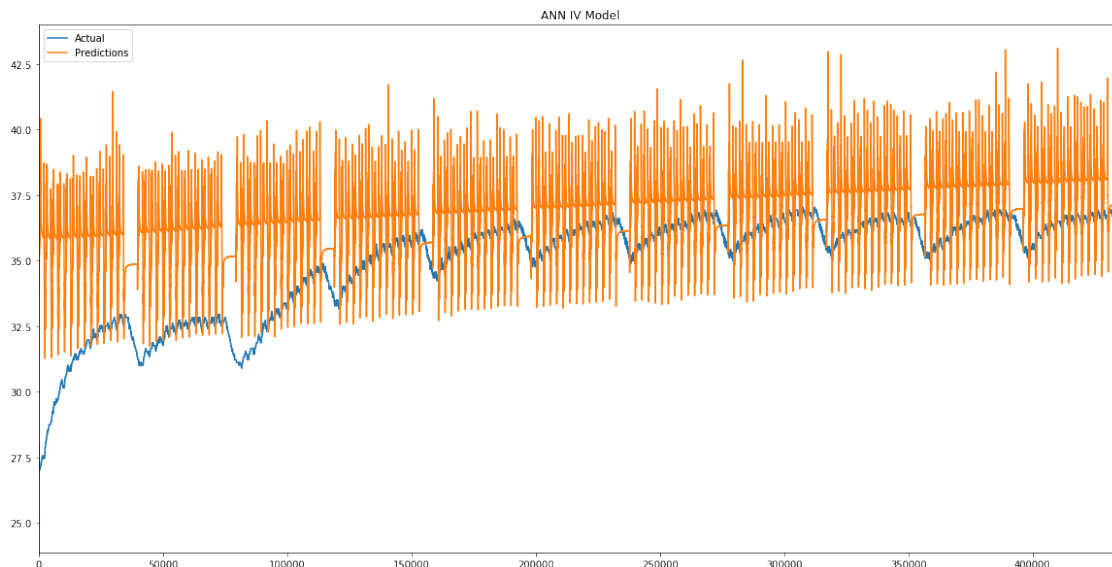


1.2 ANN IV Model

1.2.1 Data loading and cleaning

```
[4]: df = tm_gb.load_csv(filename = 'LDPRF_2098.csv',  
                        features_list =  
                        →['runtime_s', 'Current', 'Voltage', 'Amb', 'Temp'],  
                        mode = 1)
```

```
[5]: df = df.drop(columns=['runtime_s'])  
  
ML_model = tf.keras.models.load_model('ANN_0.9_IV_model_1' + '.h5')  
_ = tmm.predictions_plot(model_name = "ANN IV Model",  
                        dataframe_entry = df,  
                        model = ML_model,  
                        window_size = 1,  
                        plot_errors = True)
```



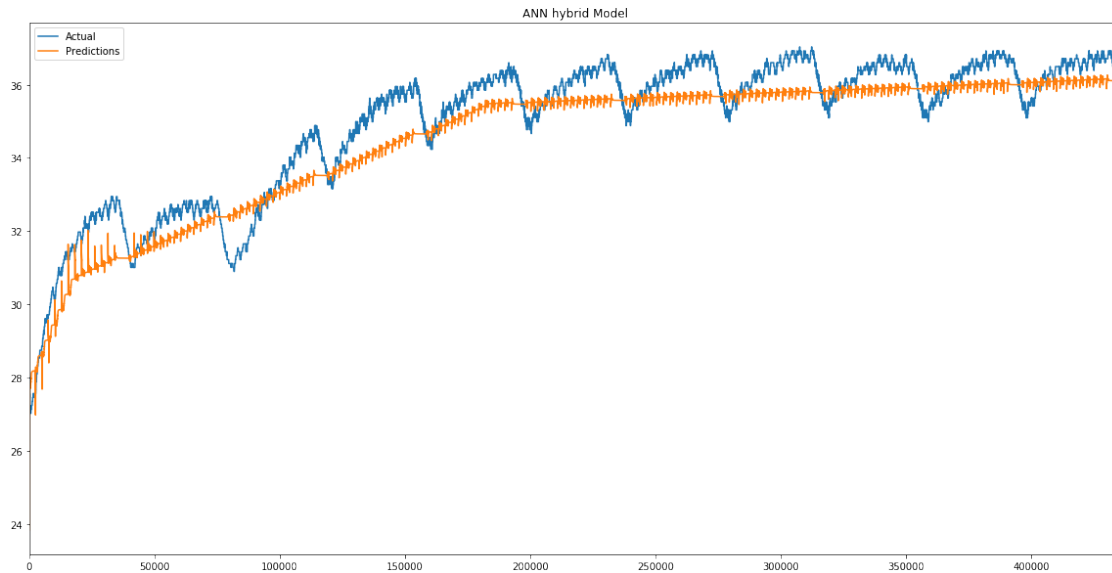
1.3 ANN Hybrid Model

1.3.1 Data loading and cleaning

```
[6]: df = tm_gb.load_csv(filename = 'LDPRF_2098.csv',  
                        features_list =  
                        →['runtime_s', 'Current', 'Voltage', 'AhCha', 'AhDch', 'Amb', 'Temp'],  
                        mode = 2)
```

```
[7]: df = df.drop(columns=['runtime_s'])  
  
ML_model = tf.keras.models.load_model('ANN_0.9_hybrid_model_1' + '.h5')
```

```
_ = tmm.predictions_plot(model_name = "ANN hybrid Model",
                        dataframe_entry = df,
                        model = ML_model,
                        window_size = 1,
                        plot_errors = True)
```



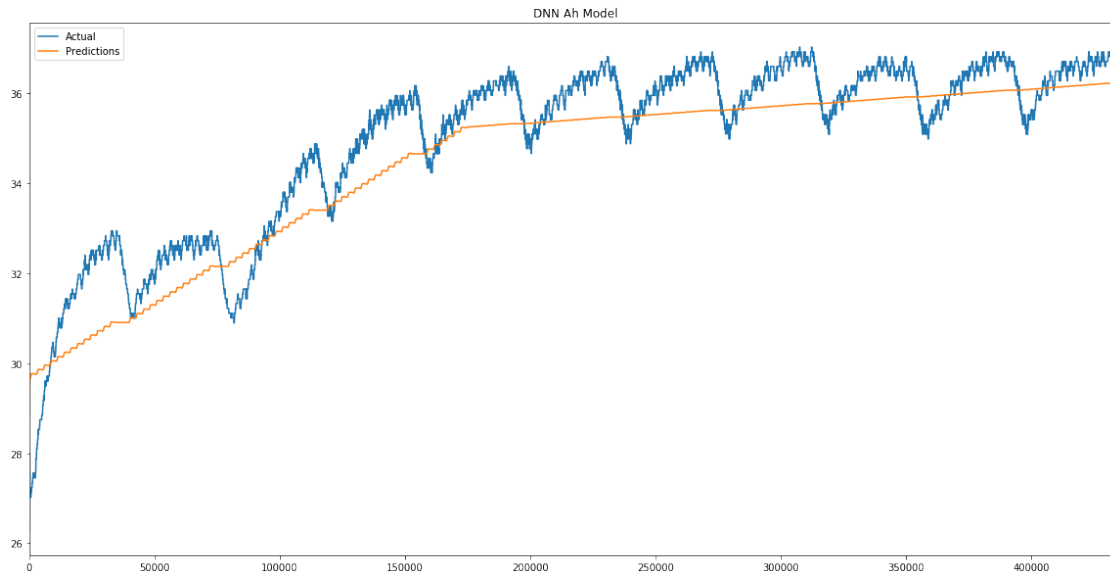
1.4 DNN Ah Model

1.4.1 Data loading and cleaning

```
[8]: df = tm_gb.load_csv(filename = 'LDPRF_2098.csv',
                        features_list = ['runtime_s', 'AhCha', 'AhDch', 'Amb', 'Temp'],
                        mode = 0)
```

```
[9]: df = df.drop(columns=['runtime_s'])
```

```
ML_model = tf.keras.models.load_model('DNN_0.9_Ah_model_1' + '.h5')
_ = tmm.predictions_plot(model_name = "DNN Ah Model",
                        dataframe_entry = df,
                        model = ML_model,
                        window_size = 1,
                        plot_errors = True)
```



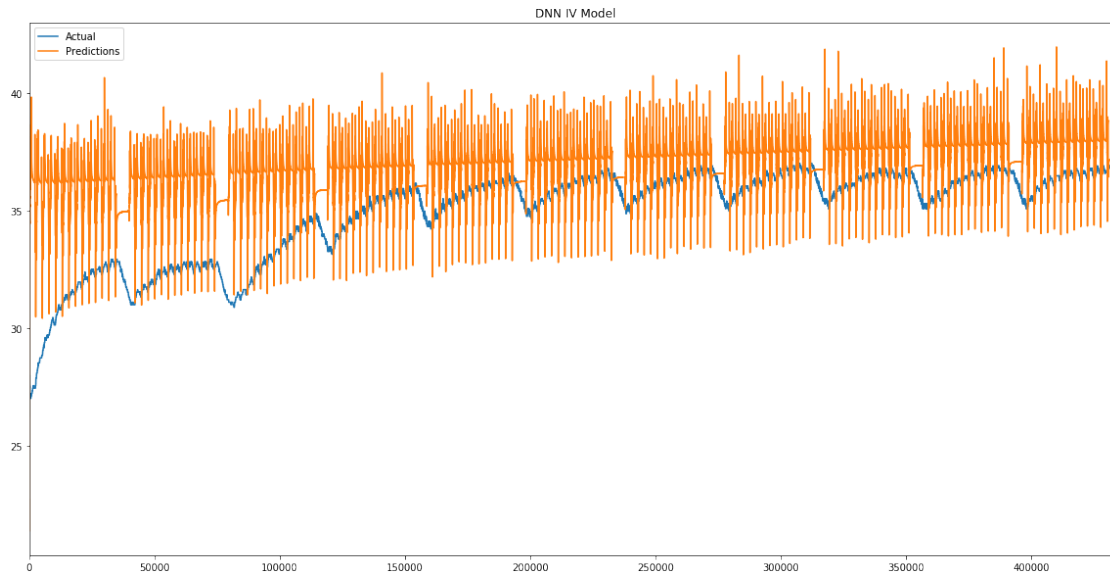
1.5 DNN IV Model

1.5.1 Data loading and cleaning

```
[10]: df = tm_gb.load_csv(filename = 'LDPRF_2098.csv',
                        features_list = □
                        → ['runtime_s', 'Current', 'Voltage', 'Amb', 'Temp'],
                        mode = 1)
```

```
[11]: df = df.drop(columns=['runtime_s'])

ML_model = tf.keras.models.load_model('DNN_0.9_IV_model_1' + '.h5')
_ = tmm.predictions_plot(model_name = "DNN IV Model",
                        dataframe_entry = df,
                        model = ML_model,
                        window_size = 1,
                        plot_errors = True)
```



1.6 DNN Hybrid Model

1.6.1 Data loading and cleaning

```
[12]: df = tm_gb.load_csv(filename = 'LDPRF_2098.csv',
                           features_list = [
                               → ['runtime_s', 'Current', 'Voltage', 'AhCha', 'AhDch', 'Amb', 'Temp'],
                                   mode = 2)
```

```
[13]: df = df.drop(columns=['runtime_s'])

ML_model = tf.keras.models.load_model('DNN_0.9_hybrid_model_1' + '.h5')
_ = tmm.predictions_plot(model_name = "DNN hybrid Model",
                           dataframe_entry = df,
                           model = ML_model,
                           window_size = 1,
                           plot_errors = True)
```

