ICARUS

Diffraction cold open

PRELUDE

The words have been written in green are the name of jupyter notebooks and after that the hyperlink of their code cell in google colab have been presented with yellow color.

IMPORTING AND OPENING DATA

1. Importing and opening data by “uproot.open” and creating:

I. Pandas.DataFrame:

A. Nu:
https://colab.research.google.com/drive/1JmetWaTR_cePTNVKo8FZ9xc507UWViz4#scrollTo=6RVRYaNXQjkd&line=2&uniqifier=1

B. NewNu: (with regard to the changing hit_peakT value in respect of hit_tpc.

https://colab.research.google.com/drive/1_qKyDnQUmAWExNZ9nnEbkGymXeoaC2jC#scrollTo=eEJ-CvvqDCEx&line=1&uniqifier=1

PLOTTING HISTOGRAMS

1. newNu:

https://colab.research.google.com/drive/1_qKyDnQUmAWExNZ9nnEbkGymXeoaC2jC#scrollTo=9hsYluK2sJfC&line=2&uniqifier=1

2. mlf:

https://colab.research.google.com/drive/14MMx4_2ZxiBWYwKA9meGE5TU2_0BRRq8#scrollTo=vGJMC4zKAghh&line=2&uniqifier=1

SCATTER PLOTS OF SIGNALS

1.For

I. Neutrino events – tpc = 0 and plane = 2 for one file(100 collected events)

https://colab.research.google.com/drive/1X7zn6JSZ0SF2pmDBJGtkHTL3wTdQqVgW#scrollTo=nwH4UmzqcCMl&line=1&uniqifier=1

II. Neutrino events – tpc = 0 and plane = 2 – total

https://colab.research.google.com/drive/1X7zn6JSZ0SF2pmDBJGtkHTL3wTdQqVgW#scrollTo=nwH4UmzqcCMl&line=1&uniqifier=1

III. Cosmic events:

https://colab.research.google.com/drive/1X7zn6JSZ0SF2pmDBJGtkHTL3wTdQqVgW#scrollTo=YaKI0Lv7kmTw&line=1&uniqifier=1

CORRELATION MATRIX OF HIT-VARIABLES

1. ICA: Nuotrino Event

https://colab.research.google.com/drive/1OZEBh0Ds0nmy3CvDO3Hi9OP5WmV7zB5L#scrollTo=LJP2qUWBbV22&line=4&uniqifier=1

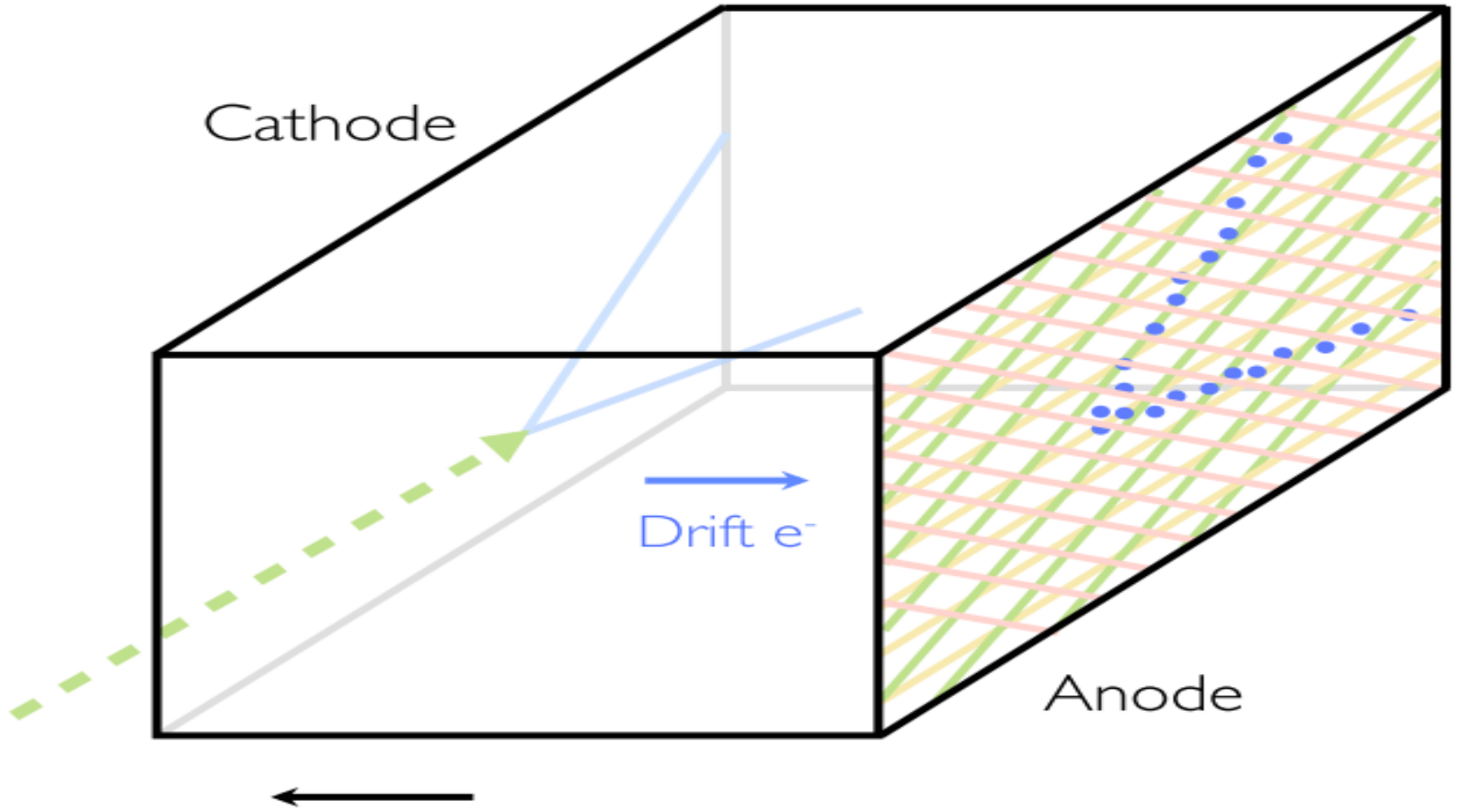
2. ICA: Cosmic:

https://colab.research.google.com/drive/1OZEBh0Ds0nmy3CvDO3Hi9OP5WmV7zB5L#scrollTo=LJP2qUWBbV22&line=4&uniqifier=1

3. Nu:

https://colab.research.google.com/drive/1JmetWaTR_cePTNVKo8FZ9xc507UWViz4#scrollTo=jPRVwm5Lvx_m&line=2&uniqifier=1

FORMING A MATRIX FOR SIGNALS WHICH COME TO PHOTO MULTIPLIERS



FORMING A MATRIX FOR SIGNALS WHICH COME TO PHOTO MULTIPLIERS

I. **SkatchIII**: as pandas serie

https://colab.research.google.com/drive/1_qKyDnQUmAWExNZ9nnEbkGymXeoaC2jC#scrollTo=eEJ-CvvqDCEx&line=1&uniqifier=1

2.**SkatchII**:

https://colab.research.google.com/drive/1cbRVF2xStXvf4jPj6Vj_drd2rE2rP-xq#scrollTo=sVj4X5DfK2Pk&line=2&uniqifier=1

3. **AK**: as pandas daraframe

https://colab.research.google.com/drive/14MMx4_2ZxiBWYwKA9meGE5TU2_0BRRq8#scrollTo=jSuEjgACAgh5&line=1&uniqifier=1

3. **Nova**: as numpy narray:

https://colab.research.google.com/drive/1P2emeTMGNg9Nzx2x7q6KSYiMxrfm5v3n#scrollTo=TknsO29UrRVE&line=2&uniqifier=1

4. **MLC**:

https://colab.research.google.com/drive/1xo-LADG0aG44Jfkq04V1PHUhf0XjlaYl#scrollTo=2Py2lil2UMj_&line=7&uniqifier=1

5. Prmu: numpy array:

https://colab.research.google.com/drive/13JYZ9_sRTxul9C8LONgo0FuRHbsnloci#scrollTo=6TnUZVfSs-am&line=5&uniqifier=1

FORMING A MATRIX FOR SIGNALS WHICH COME TO PHOTO MULTIPLIERS

One of the our difficulties is about changing elements of this matrix from 0 to one. This task needs adequate ram

ML

We tried to train the input data- which has been tagged by two mark: ev refers to neutrino events and cos refers to cosmic long-baseline events.

Up to now, we used two supervised learning methods: Neural Network and Random Forest. The new hybrid technique combination of decision tree and NN could be tune on this dataframe.

The issue here is the execution of these algorithms need a lot power-RAM, CPU or GPU and for instance to exploit Keras Tuner or other methods for extraction hyperparameters, it is required to run in cluster or cloud.

ML BY NN

1. Nu:

https://colab.research.google.com/drive/1JmetWaTR_cePTNVKo8FZ9xc507UWViz4#scrollTo=OfgoIMLvYavU&line=2&uniqifier=1

RF

1. mlf:

https://colab.research.google.com/drive/14MMx4_2ZxiBWYwKA9meGE5TU2_0BRRq8#scrollTo=jSuEjgACAgh5&line=1&uniqifier=1

Combination of NN and RF

- 0. combDT&NN
- https://colab.research.google.com/drive/1hoDUkMaSL4g78HO4lfpcjlo072b6j21B
-