(GIS) OSmOSE
Open Science meets
Ocean Sound Explorers

User guide of OSmOSE analytics platform

OSmOSE Working Report

Authorship This document was drafted by

- Dorian Cazau[1)]

belonging to the following institute (at the time of their contribution): 1) Lab-STICC, ENSTA Bretagne.

Document Review Though the views in this document are those of the authors, it was reviewed by a panel of acousticians before publication. This enabled a degree of consensus to be developed with regard to the contents, although complete unanimity of opinion is inevitably difficult to achieve. Note that the members of the review panel and their employing organisations have no liability for the contents of this document.

The Review Panel consisted of the following experts (listed in alphabetical order):

- Maelle Torterotot[1)]

belonging to the following organisms / research institutes (at the time of their contribution): 1) Lab-STICC, IMT Atlantique.

Last date of modifications June 16, 2022

Recommended citation "User guide for OSmOSE notebooks", OSmOSE working report (version dating from June 16, 2022, distributed openly on https://osmose.xyz/)

Future revisions Revisions to this document will be considered at any time, as well as suggestions for additional material or modifications to existing material, and should be communicated to Dorian Cazau (dorian.cazau@ensta-bretagne.fr).

Contents

1 Pre-requirements

1.1 Getting your logins

Before, you need to have an account on Datarmor, that will provide you with

- extranet logins (username + password)
- intranet logins (username + password)

To get an account, please contact your IT assistant if you are from an institute partner of Datarmor, otherwise contact dorian.cazau@ensta-bretagne.fr.

1.2 Connecting to Datarmor

Datarmor can be accessed in two ways.

- via VPN and ssh following the document "Extranet access to DATARMOR." provided in annex ;
- via a user-friendly portal at `https://domicile.ifremer.fr/dana-na/auth/url_default/welcome.cgi`.

We highly recommend the second solution to new users.

1.3 Executing a notebook on Datarmor

Most of our services require the use of Jupyter notebooks deployed on Datarmor. So let's see how to use them.

Step 1: launch a jupyterhub session

Now you can click on JUPYTER on the portal, enter your intranet logins and click on Start My Server. You will then have to select a job profile, corresponding to the computer resources on which you wish to run your jupyterhub. By default, select `Datarmor-Jupyterlab-8cores,32GBRAM,2hours` (quite equivalent to a personal computer but that is enough to use our notebook, unless you have specific needs such as running a AI model in your jupyterhub session). You are now on the Jupyter-Hub of Datarmor!

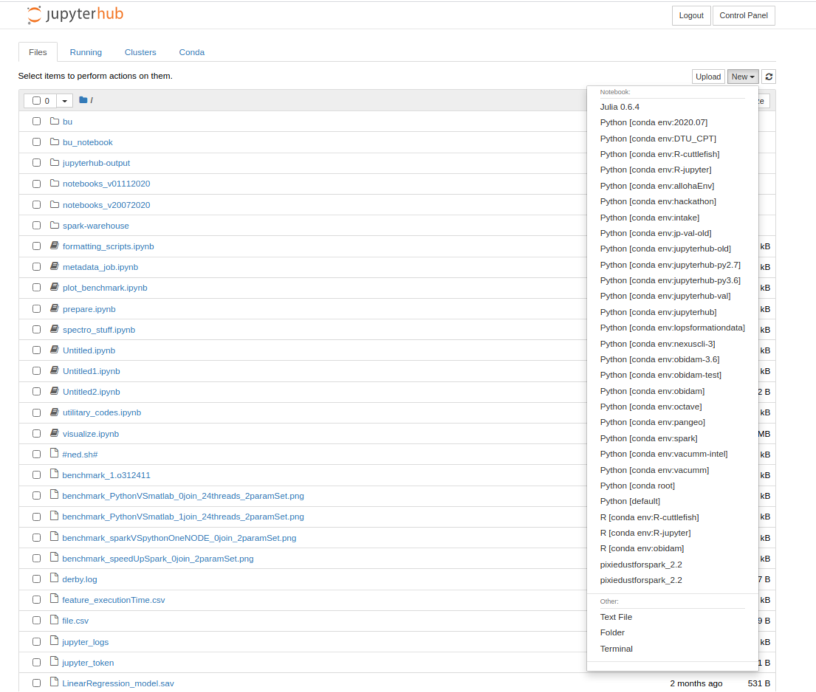
Step 2: copy and paste OSmOSE notebooks in your datahome

OSmOSE notebooks are located on our workspace at `/home/datawork-osmose/`. You will have to make a copy of them into your personal workspace. One way of doing that is to to open a terminal from your jupyterhub portal (see figure 1) and run the following command:

```
>> cp -R /home/datawork-osmose/osmoseNotebooks_v0/notebooks/ . ; chmod -R 700
    ↪ osmoseNotebooks_v0/
```

Step 3: use a notebook

To open a notebook, just double click on it. If your notebook kernel is not automatically set to allohaEnv (see Fig. 2), change it in Kernel → Change Kernel → Python [conda env:allohaEnv]

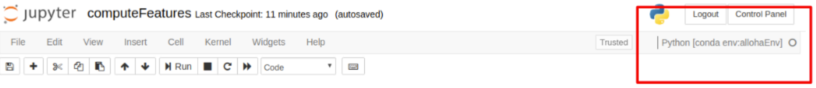

jupyterhub
Logout
Control Panel
Files
Running
Clusters
Conda
Upload
New
Notebook:
Julia 0.6.4
Python [conda env:2020.07]
Python [conda env:DTU_CPT]
Python [conda env:R-cuttlefish]
Python [conda env:R-jupyter]
Python [conda env:allohaEnv]
Python [conda env:hackathon]
Python [conda env:intake]
Python [conda env:jp-val-old]
Python [conda env:jupyterhub-old]
Python [conda env:jupyterhub-py2.7]
Python [conda env:jupyterhub-py3.6]
Python [conda env:jupyterhub-val]
Python [conda env:jupyterhub]
Python [conda env:lopsformationdata]
Python [conda env:nexuscli-3]
Python [conda env:obidam-3.6]
Python [conda env:obidam-test]
Python [conda env:obidam]
Python [conda env:octave]
Python [conda env:pangeo]
Python [conda env:spark]
Python [conda env:vacumm-intel]
Python [conda env:vacumm]
Python [conda root]
Python [default]
R [conda env:R-cuttlefish]
R [conda env:R-jupyter]
R [conda env:obidam]
pixiedustforspark_2.2
pixiedustforspark_2.2
Other:
Text File
Folder
Terminal

Select items to perform actions on them.

□ 0 /

- □ bu
- □ bu_notebook
- □ jupyterhub-output
- □ notebooks_v01112020
- □ notebooks_v20072020
- □ spark-warehouse
- □ formatting_scripts.ipynb
- □ metadata_job.ipynb
- □ plot_benchmark.ipynb
- □ prepare.ipynb
- □ spectro_stuff.ipynb
- □ Untitled.ipynb
- □ Untitled1.ipynb
- □ Untitled2.ipynb
- □ utilitary_codes.ipynb
- □ visualize.ipynb
- □ #ned.sh#
- □ benchmark_1.o312411
- □ benchmark_PythonVSmatlab_0join_24threads_2paramSet.png
- □ benchmark_PythonVSmatlab_1join_24threads_2paramSet.png
- □ benchmark_sparkVSpythonOneNODE_0join_2paramSet.png
- □ benchmark_speedUpSpark_0join_2paramSet.png
- □ derby.log
- □ feature_executionTime.csv
- □ file.csv
- □ jupyter_logs
- □ jupyter_token
- □ LinearRegression_model.sav 2 months ago 531 B

Figure 1: Open a terminal from jupyterhub.



computeFeatures Last Checkpoint: 11 minutes ago (autosaved)

Python [conda env:allohaEnv] ○

Figure 2: Notebook kernel need to be set to allohaEnv.

Step 4: list of must-do and things to keep in mind

Except that, the use of our notebooks should be straight forward even for users without basic skills in Python. Just be careful with a few “Must and Must not do” :

- Do not implement computationally expensive processing codes on notebooks. This should be done with job batch - better that you contact dorian.cazau@ensta-bretagne before ;
- Avoid underusing your notebook, especially do not forget to close your notebook session once you are done. To do so, follow this procedure :
  1. on the Jupyter Hub main page → click tab Running → click on all Shutdown
  2. then click on the button Logout on the Jupyter Hub main page

  Please see our Note in annex regarding how to shut down a jupyter hub session on Datarmor;

And keep in mind that :

- In case of a long computation (for example, resampling high-frequency data), your notebook might close itself due to its limited wall-time. This is not a problem, all your computation will keep running safely, however you will not be able to track progress of them through the notebook ... maybe it is time for you to learn the following command to be run in the jupyter hub terminal :

  ```
  >> qstat - u myusername
  ```

2 How using our platform ?

2.1 Use case 1 : Importing a new dataset

Let’s start with some constraints you will have to comply with :

- all your audio files must be .wav files ;
- all your audio files must have the same characteristics, especially duration and sample frequency ;
- so far, we are dealing with files with a maximal duration of 388 minutes and a maximal volume of 230 MB ; if your files happen to exceed drastically one or both of these values, please contact dorian.cazau@ensta-bretagne.fr before going further.

Then follow these instructions to import your dataset :

1. build the dataset folder `/home/datawork-osmose/dataset/mydatasetname/raw/audio/original` , where you have to replace `mydatasetname` by your dataset name (can be done through FileZilla or SSH terminal) ;
2. upload your audio files in this folder. This step MUST be done through FileZilla using datacopy as host ;
3. download the file `/home/datawork-osmose/osmoseNotebooks_v0/file_examples/timestamp.csv`, fill it for your campaign respecting the exact same format, and then put it in `/home/datawork-osmose/dataset/mydatasetname/raw/audio/original` (can be done through FileZilla or SSH terminal). Check our code snippets in Annex for this operation!
4. in case you have a mobile hydrophone, download the file `/home/datawork-osmose/osmoseNotebooks_v0/file_examples/gps_depth.csv`, fill it for your campaign respecting the exact same format **and variable names** (depth is not mandatory), and then put it in `/home/datawork-osmose/dataset/mydatasetname/raw/auxiliary/` after creating the corresponding folder (can be done through FileZilla or SSH terminal) ;

5. in case you have auxiliary data, download the file `/home/datawork-osmose/osmoseNotebooks_v0/file_examples/ECMWF_1H.csv`, fill it for your campaign respecting the exact same format, and then put it in `/home/datawork-osmose/dataset/mydatasetname/raw/auxiliary/` after creating the corresponding folder (can be done through FileZilla or SSH terminal) ;
6. in case you have any other data or documents you want to store along with your dataset[1], please put them in `/home/datawork-osmose/dataset/mydatasetname/other/` after creating the corresponding folder (can be done through FileZilla or SSH terminal) ;
7. open and execute the notebook `build_dataset.ipynb` from our notebook package (for information on notebooks see section 1.3).

2.2 Use case 2 : Creating a set of spectrograms for an APLOSE campaign

This will be done in two steps :

1. use the notebook `fileScaleAnalysis.ipynb` to create the set of spectrograms for your campaign
2. without waiting the end of the generation of your spectrograms, use the notebook `buildDatasetsCSV_temporaryNotebook.ipynb` to create an internal file that will be used to import your dataset into our annotation tool APLOSE ;

2.3 Use case 3 : Performing soundscape analysis

Our notebooks offer various approaches to perform soundscape analysis through the notebook `datasetScaleAnalysis.ipynb`.

2.3.1 Raw welch

You can compute and retrieve welch spectra at different time resolutions in a .pkl format, specified in the variables `welch_timeScale` (eg `welch_timeScale=['30S','10T']`).

2.3.2 SPL timeseries

Timeseries of median-averaged SPL can be

- filtered between two frequencies ;
- generated in a sequential mode, ie one plot for each successive time period. The duration of the time period is constant and set through the `sequential_timePeriod` variable ;
- generated in a warping mode, ie all spectra from a same time period are welch averaged and plotted successively, eg all spectra from one particular day of the week (cf figure 3).

2.3.3 Empirical Plot Density

EPDs can be

- filtered between two frequencies ;
- generated in a sequential mode, ie one plot for each successive time period. The duration of the time period is constant and set through the `sequential_timePeriod` variable ;
- generated in a warping mode, ie all spectra from a same time period are processed altogether (cf figure 4).

[1] Better to keep it all in one place!

2.3.4 Long Term Averaged Spectra

LTAs can be

- generated in a sequential mode, ie one plot for each successive time period. The duration of the time period is constant and set through the `sequential_timePeriod` variable ;
- generated in a warping mode, ie all spectra from a same time period (variable `warp_timePeriod`) are welch averaged and plotted successively, eg all spectra from one particular day of the week (cf figure 5).

2.3.5 Two-level recurrent bar plots

Two-level recurrent bar plots can be obtained by setting a second time resolution `small_timeres`, which must be smaller than `warp_timePeriod`.

- generated in a sequential mode, ie one plot for each successive time period. The duration of the time period is constant and set through the `sequential_timePeriod` variable ;
- generated in a warping mode, ie all spectra from a same time period are welch averaged and plotted successively, eg all spectra from one particular day of the week (cf figure 6).

A one-level recurrent bar plot is obtained by setting `smallwarp_timePeriod_recurBox=['']` (cf figure 7).

2.3.6 Join auxiliary variables

You can also join auxiliary variables along your spectrograms and/or features through `aux_variable`.

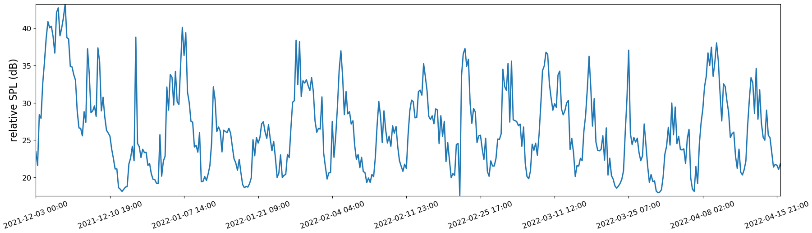


Figure 3: timeSPL over all Saturdays

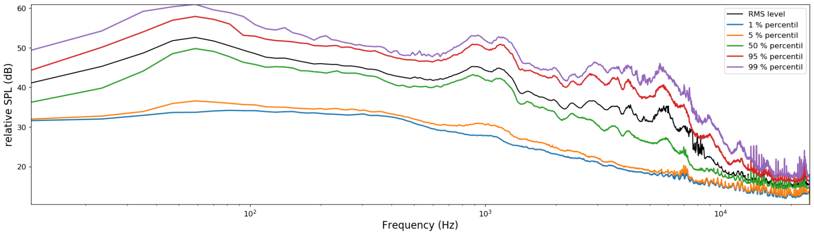


Figure 4: EPD over all Fridays

2.4 Use case 4 : How to benchmark a AI model ?

!! Not really easy to go through this use case for the moment sorry, we are working on this !!

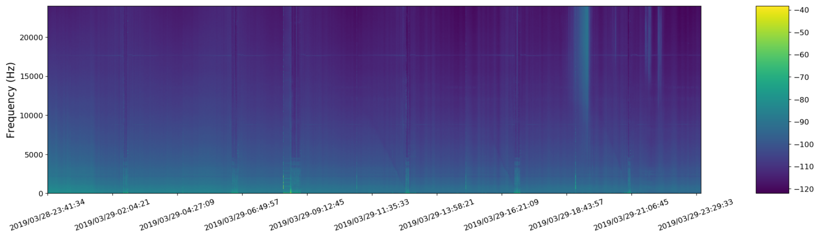


Figure 5: LTAS over one day

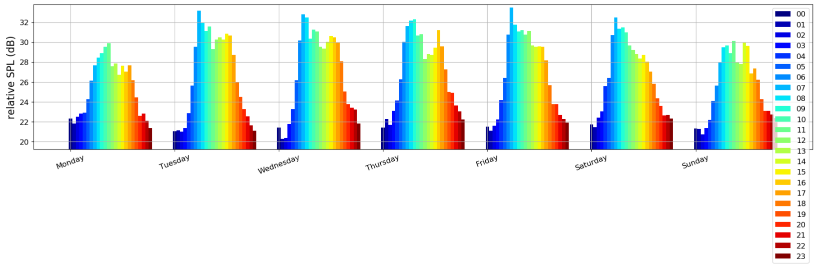


Figure 6: Two-level recurrent bar plots, showing filtered median-averaged SPL over hours within the different days of week

2.4.1 Choose your task

Choose one of the following AI tasks :

- task 1 : presence / absence of biological sound events in fixed-duration sound clips / weak labels / 2 classes ;
- task 2 : biological sound event detection

2.4.2 Integrate your model

Several conditions must be filled to integrate your models in our platform :

- your model must formatted as a python function, inserted in the module `/home/datawork-osmose/osmoseNotebooks_v0/source/module_AImodels.py` (see current models in this module as examples) ;
- this function should return an accuracy score and the path of your model metadata (e.g. the architecture of your network model) ;
- you also have to edit `/home/datawork-osmose/osmoseNotebooks_v0/source/launcher_AI.py` to launch your model (here also take currently implemented models as examples)

2.4.3 Execute your model

Choose one of the following execution environment :

- notebook in jupyter hub session : use the `/home/datawork-osmose/osmoseNotebooks_v0/notebooks/AI.ipynb`. In this mode, the execution will be done within your jupyterhub session, so allocate computational resources adapted to your need. Multi-threaded models can use for example `Datarmor-Jupyterlab-28cores,115GBRAM,2hours`, and convolutional deep learning models can use GPUs `Datarmor-Jupyterlab-1GPU,8cores,64GBRAM,4hours` or even `Datarmor-Jupyterlab-2GPU,16cores,128GBRAM,4hours`. Be also careful to time limit associated with your session. Several kernels are already available for machine learning (eg tensorflow and pytorch with their last versions)

3 How contributing to the development of our platform ?

3.1 Contribution 1 : Proposing new features and reporting bugs

Any proposition of new features and bug reporting must be done through the following documents hosted on our google drive https://docs.google.com/document/d/1U7Y4YEQaowNztZXBJCXHdHOVnXzjwp4pIaOJrZOmYGI/edit#. In case you need access to these documents, please send a mail to Dorian Cazau (dorian.cazau@ensta-bretagne.fr).

3.2 Contribution 2 : Helping in code reviewing and development

To go further with this second contribution, please go to our GITHUB page : https://github.com/Project-ODE/jupyter_scripts.

ANNEX

Code snippets

We provide the following set of code snippets here `/home/datawork-osmose/codeSnippets/`, useful for different operations on our platform:

- `create_timestampCSV.ipynb` : to be used in Use Case 1 Step 3, i.e. to generate the file `timestamp.csv` from your imported audio files.

Note on How to close a Jupyter Hub session

On the 24 Mars 2022 16:07, ticket RIC 2022032420001218

malheureusement, pour dtruire entirement un serveur de notebooks ou un lab lanc depuis jupyterhub, il faut effectivement la fois stopper le serveur ct web, et qdel le job ct datarmor.

On travaille amliorer cela.

Ceci dit, il n'est pas ncessaire de tuer ton job datarmor pour relancer un nouveau serveur via le web. Stopper le serveur suffit.

Extranet access to DATARMOR

When you are outside your institute (IFREMER/SHOM/IUEM/UBO/ENSTA Bretagne/Ecole Naval) you have to use Pulse Secure to be able to use a ssh connection to DATARMOR.

There are 3 steps :

- Installation
- Using Pulse Secure
- Access to datarmor

1 Installation

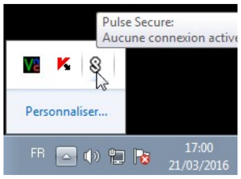
You can download Pulse Secure here : https://cloud.ifremer.fr/index.php/s/WC9GArY8Eo51yZE
There is a version for :

- Windows 64 bits : JunosPulse.x64.msi
- Windows 32 bits : JunosPulse.x86.msi
- Mac OS X 64 bits : JunosPulse.dmg
- Linux 64 bits
  - CentOS 6.4 / Fedora 23 : pulse-8.1R11.1.i386.rpm
  - Fedora 24 or 25 : ps-pulse-linux-5.2r6.0-b977-centos-rhel-installer.rpm
    - ***rpm -ivh pulse-xxxx.rpm***
  - Ubuntu 14.04 : pulse-8.1R11.1.i386.deb
    - ***dpkg -i pulse-xxxx.deb***
    - *With ubuntu, you'll probably have to install these libraries :*
    - **apt-get install libc6-i386**
    - **apt-get install lib32z1**
  - Ubuntu 17.04 is not officially supported but it works with : pulse-8.2R5.i386.deb
    - **sudo dpkg -i pulse-xxxx.deb**
    - **sudo apt-get install libstdc++6:i386**

2 Using Pulse Secure

For Windows and MacOS

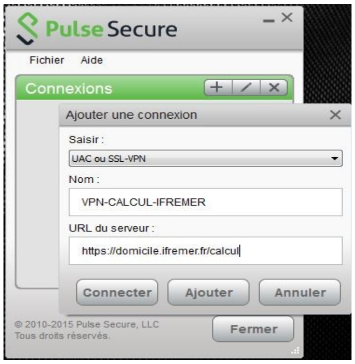
For the first time, you'll have to configure the connection :
Execute Pulse Secure :



Windows : launch Pulse Secure

MacOs : launch Pulse Secure

and add a new connection with « + » like this :

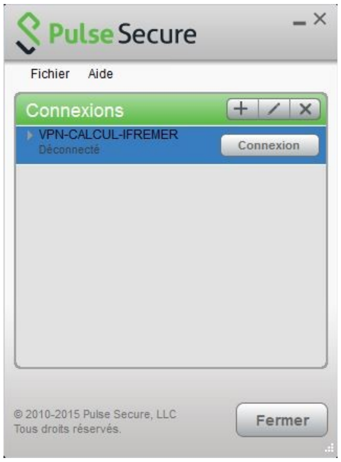


Saisir : UAC ou SSL-VPN

Nom : VPN-CALCUL-IFREMER

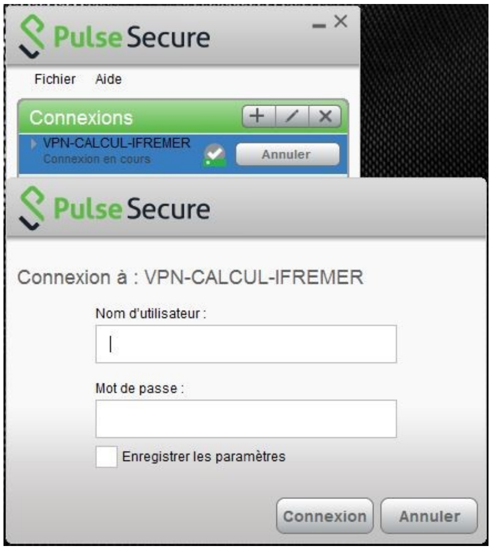
URL du serveur : https://domicile.ifremer.fr/calcul

and click Add



VPN-CALCUL-IFREMER
Déconnecté

Click Login and enter your extranet login and password



VPN-CALCUL-IFREMER
Connexion en cours

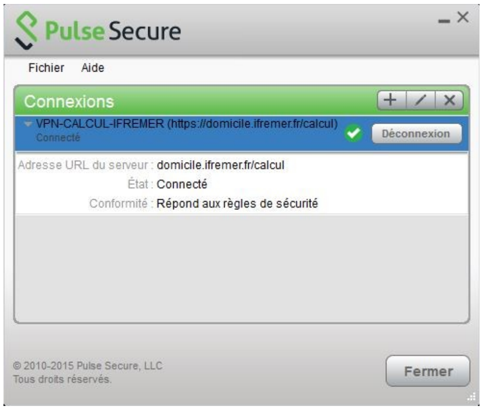
Connexion à : VPN-CALCUL-IFREMER

Nom d'utilisateur :

Mot de passe :

□ Enregistrer les paramètres

You are connected :



VPN-CALCUL-IFREMER (https://domicile.ifremer.fr/calcul) ✓
Connecté

Adresse URL du serveur : domicile.ifremer.fr/calcul
État : Connecté
Conformité : Répond aux règles de sécurité

At the end, just click disconnect.

For Linux

To connect, just use the command line (Think about logging off in firefox before) :

```
/usr/local/pulse/PulseClient.sh -h domicile.ifremer.fr -u your_extranet_login  -U https://domicile.ifremer.fr/calcul -r vpn-Ifremer
VPN Password:
```

Give your password and you should be connected.
(With Ubuntu, you have to use "sudo /usr/local/pulse/PulseClient.sh ...." just for the 1st connection)

To check the status, you can use this command in an another terminal window :

```
/usr/local/pulse/PulseClient.sh -S
Connection Status :
        connection status : Connected
        bytes sent : 584
        bytes received : 0
        Connection Mode : ESP
        Encryption Type : AES128/SHA1
        Comp Type : None
        Assigned IP : 134.246.222.xxx
```

To disconnect, simply do "Control + C" on the 1st connection window or launch from another Terminal window:

```
/usr/local/pulse/PulseClient.sh -K
```

3 *Access to datarmor*

When you are connected with pulse secure, you can connect to datarmor via ssh :

```
ssh datarmor.ifremer.fr
```

or

ssh datarmor0-10g.ifremer.fr (same as **ssh 134.246.184.4)**
ssh datarmor1-10g.ifremer.fr (same as **ssh 134.246.184.5)**
ssh datarmor2-10g.ifremer.fr (same as **ssh 134.246.184.6)**
ssh datarmor2-10g.ifremer.fr (same as **ssh 134.246.184.7)**

For windows, you can use a ssh client like putty or mobaXterm

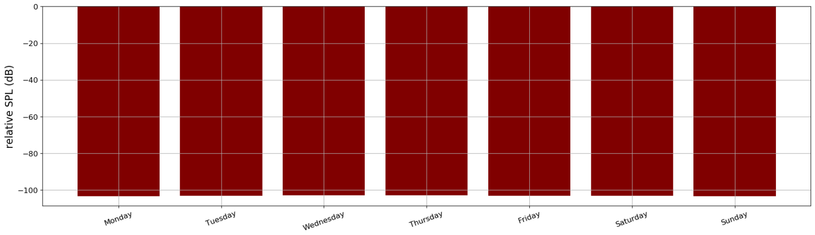


Figure 7: One-level recurrent bar plot over the different days of the week.