

---

**avant**

***Release 0.0.1***

**Nicola Farmer**

**Aug 27, 2021**



# CONTENTS

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Installation</b>   | <b>3</b>  |
| <b>2</b> | <b>Features</b>   | <b>5</b>  |
| <b>3</b> | <b>Examples</b>   | <b>7</b>  |
| <b>4</b> | <b>Frequently Asked Questions</b>   | <b>9</b>  |
| <b>5</b> | <b>API reference</b>  | <b>11</b> |
| 5.1      | avant - Creating informed priors for reflectometry analysis in Python . . . . . | 11        |



**avant** is a python package to extract values from `refl_database` to create informed or uniform priors to be used in reflectometry analysis.

The aim of **avant** is to improve reflectometry analysis by applying Bayesian Statistics and creating ‘informed priors’ which take into account literature values in the prior probability distributions of the parameters. The priors are created in a way where they can be directly implemented into `refnx` to perform reflectometry analysis. Plus, the package has plotting functionalities so you can see what the prior probability distributions look like.



## INSTALLATION

avant can be installed from the PyPi package manager with pip

```
pip install avant
```

The package can also be found on [Github](#) along with installation instructions.



## **FEATURES**

Currently, **avant** only contains priors for the following five parameters for DMPC: head volume, tail volume, head thickness, tail thickness and roughness. It can create an informed prior, *Gauss*, with the following methods:

- **pdf**: probability distribution function
- **logpdf**: natural log of the probability distribution function
- **cdf**: cumulative distribution function
- **ppf**: percentile point function (also known as quantile function or inverse cdf)
- **rvs**: random variate sampling

It can also create a uniform prior which is an upper and lower bound for the prior range.

The following plotting functionalities are available:

- **plotGauss(name)**: Plot an informed *Gauss* prior probability distribution.
- **plotUniform(name)**: Plot a uniform prior probability distribution.



## EXAMPLES

For example, to plot the informed prior for the head volume for DMPC:

```
[1]: import avant.parameter.vh as vh
     vh.plotGauss('DMPC')

-----
ModuleNotFoundError                                Traceback (most recent call last)
/tmp/ipykernel_124/3786636547.py in <module>
----> 1 import avant.parameter.vh as vh
      2 vh.plotGauss('DMPC')

ModuleNotFoundError: No module named 'avant'
```

For example, to plot the uniform prior for the head volume for DMPC:

```
[2]: import avant.parameter.vh as vh
     vh.plotUniform('DMPC')

-----
ModuleNotFoundError                                Traceback (most recent call last)
/tmp/ipykernel_124/4090706304.py in <module>
----> 1 import avant.parameter.vh as vh
      2 vh.plotUniform('DMPC')

ModuleNotFoundError: No module named 'avant'
```

Here is an example of setting the value of the parameter to that of the Gauss object - this is a class containing statistical information about the prior and can be fed straight into Refnx.

For example, to set a parameter for the head volume of DMPC:

```
[3]: import avant.parameter.vh as vh
     x = vh.Gauss('DMPC')

-----
ModuleNotFoundError                                Traceback (most recent call last)
/tmp/ipykernel_124/3562674503.py in <module>
----> 1 import avant.parameter.vh as vh
      2 x = vh.Gauss('DMPC')

ModuleNotFoundError: No module named 'avant'
```

Here is an example of setting the value of the parameter to a tuple containing the lower and upper bound of the uniform range of values for the parameter. This can be fed straight into Refnx.

For example, to set a parameter for the head volume of DMPC:

```
[4]: import avant.parameter.vh as vh
x = vh.uniform('DMPC')
```

```
-----
ModuleNotFoundError                                Traceback (most recent call last)
/tmp/ipykernel_124/3915618590.py in <module>
----> 1 import avant.parameter.vh as vh
      2 x = vh.uniform('DMPC')

ModuleNotFoundError: No module named 'avant'
```

## FREQUENTLY ASKED QUESTIONS

- Where can I find the database where the values are stored?

The `refl`-database can be found on [GitHub](#) .



## API REFERENCE

### 5.1 avant - Creating informed priors for reflectometry analysis in Python

#### 5.1.1 Modules

##### `avant.parameter`

- `genindex`
- `modindex`