# Uncertainty quantification in Machine Learning

Gabriel Gros

Under the supervision of Cédric Pradalier and Aishwarya Venkataramanan

# Summary

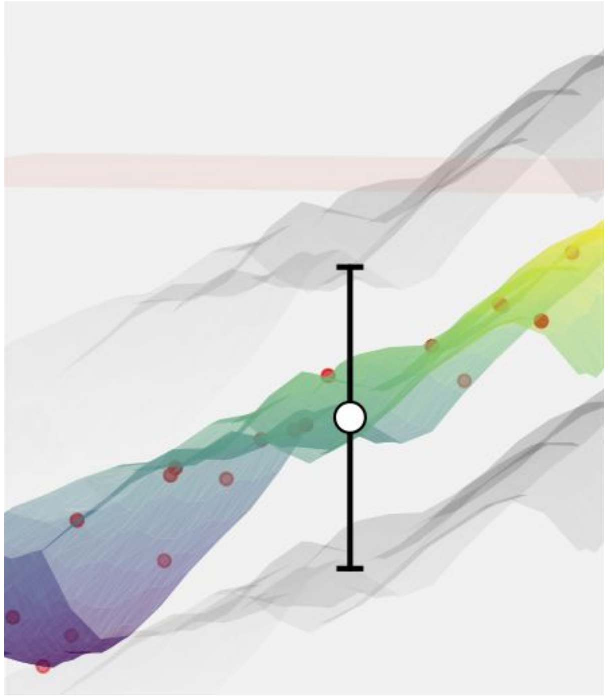
I. Presentation of the project's interest

II. The two types of uncertainties

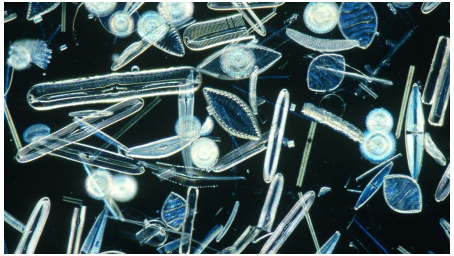
III. Tests on splitting uncertainties

IV. Trying to estimate epistemic uncertainty

V. Samples classification as functions of uncertainty



# I. Presentation of the project's interest

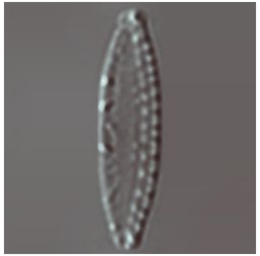


## Uncertainty quantification

- Informative purpose
- Many kinds of applications (including vision)

## The problem

Diatoms are hard to classify using AI and often require the eyes of an expert



## The proposed solution

Classify diatoms as functions of their classification uncertainty

- Identify uncertainties
- Measure uncertainties empirically
- Develop existing methods to predict potential outliers
- Use a framework to classify diatoms

# I. Presentation of the project's interest

## The pipeline

(1) Object detection

(2) Training given batch of instances

(3) Framework calculating uncertainty

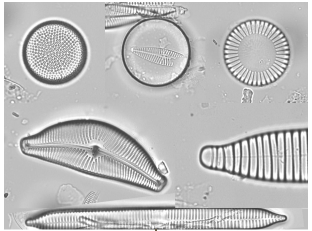
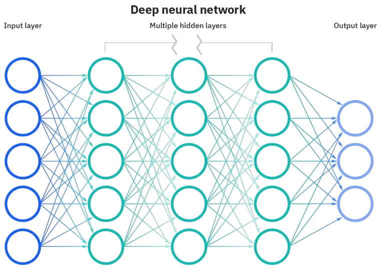


Image of diatoms

(1)

One instance

(2)




Neural Network

(3)

| |
|---|
| C1 : Inliers of c > 80% |
| C2 : Inliers of c < 20% |
| C3 : Outliers |

Classification given confidence

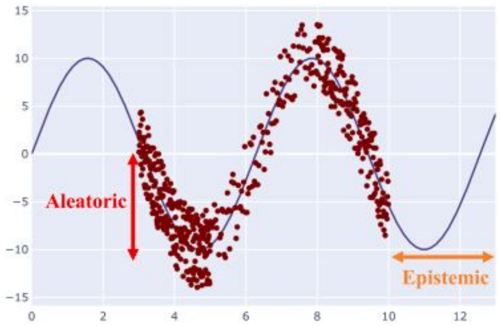
# II. The two kinds of uncertainties

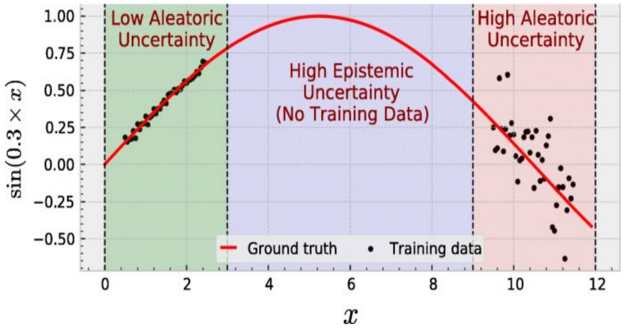
## Epistemic uncertainty

- Accounts for uncertainty in the model parameters
- Uncertainty on the model generated by the data
- Can be reduced with training

## Aleatoric uncertainty

- Captures noise inherent in the observations (sensor noise or motion noise for instance)
- Doesn't depend on the model
- Cannot be reduced



# III. Tests on splitting uncertainties

## Why ?

Could be useful to understand if uncertainty more comes from noised data or from model parameters.

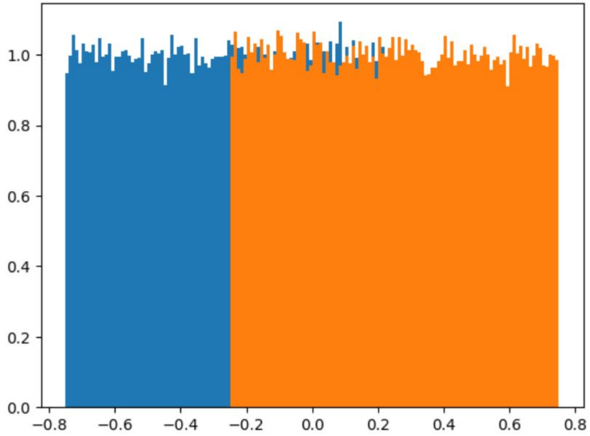
## A simple classifier : the 1D classifier

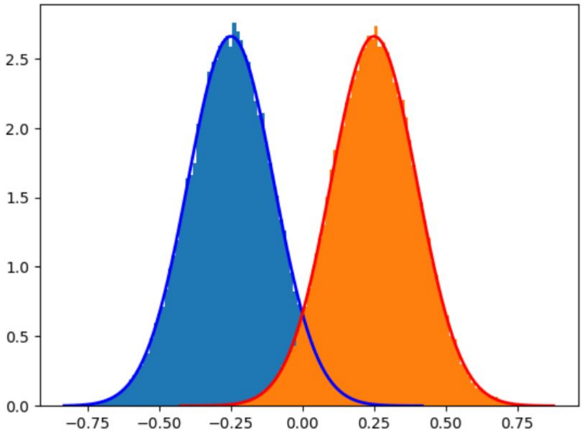
$$f(x,\lambda) = \begin{cases} -1, & x < \lambda \\ 1, & x \geq \lambda \end{cases} \qquad \text{with } \lambda \sim N(0,\sigma^2)$$

Homoscedastic case → constant standard deviation

## 2 simple datasets

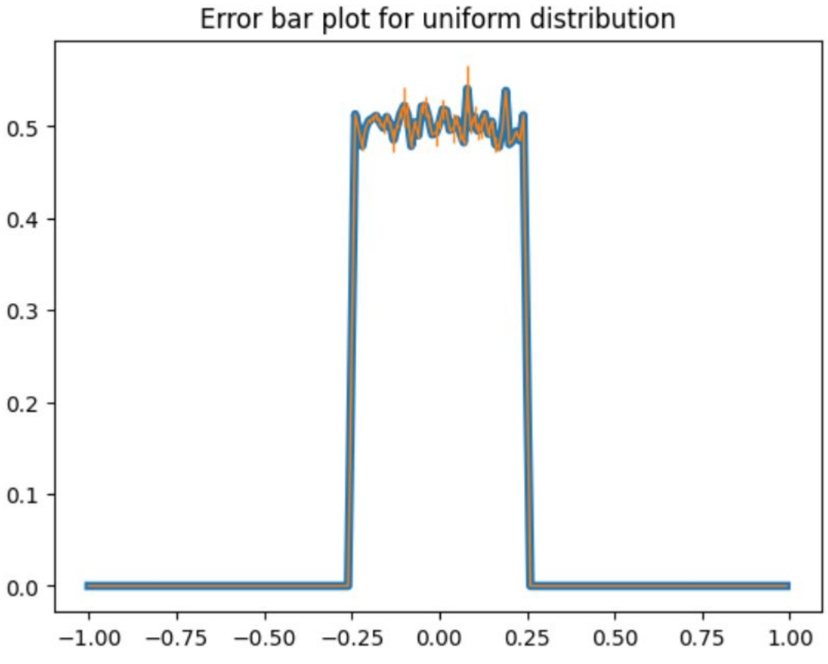
- Uniform distributions
- Gaussian distributions

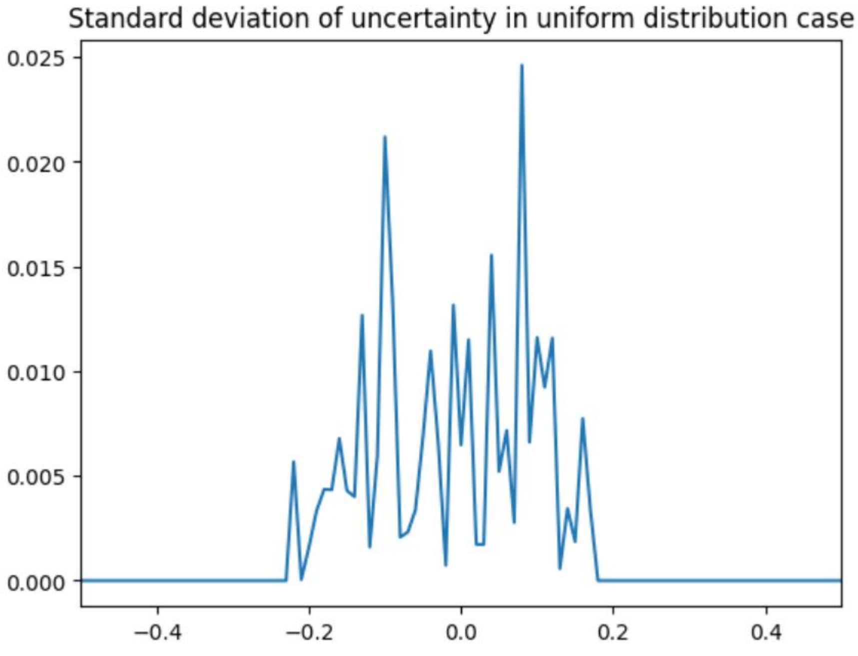


# III. Tests on splitting uncertainties

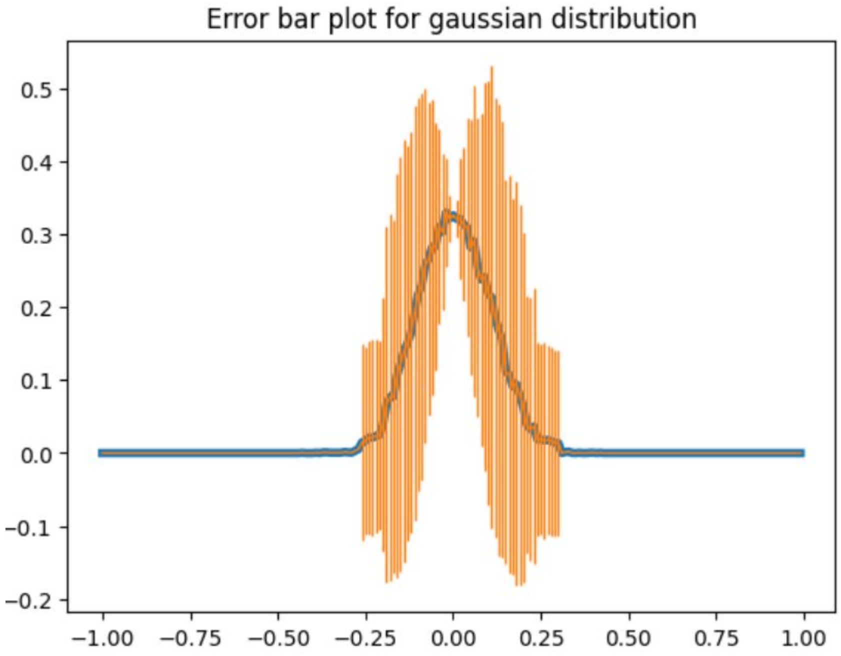
## Uniform case

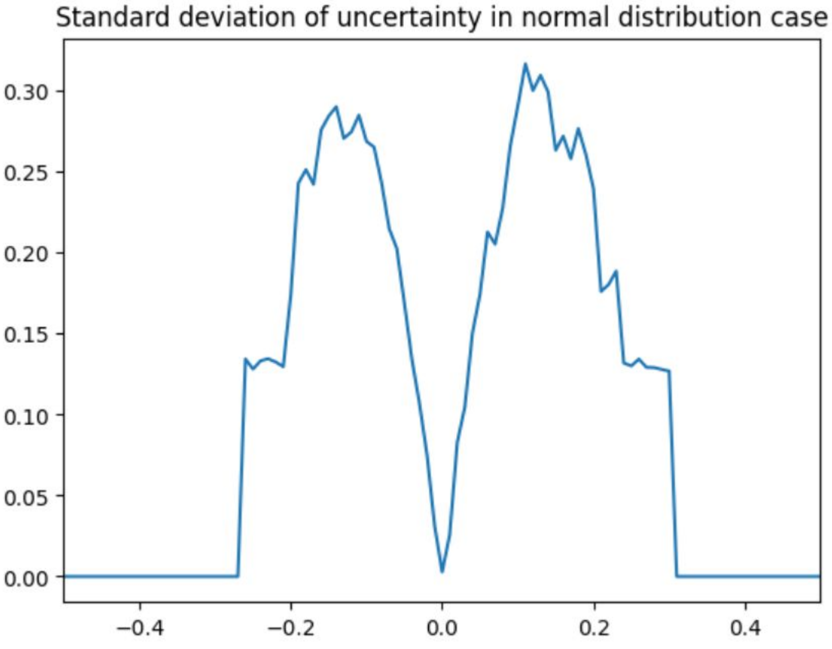


# III. Tests on splitting uncertainties

## Gaussian case



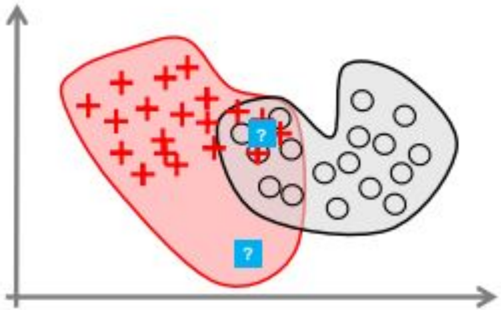



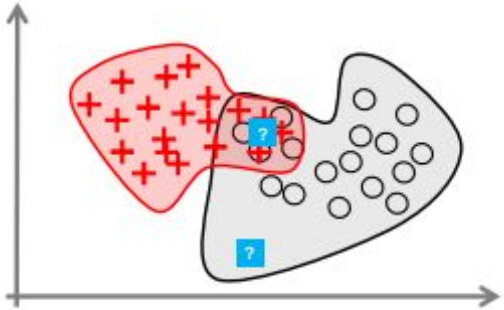
# IV. Trying to estimate epistemic uncertainty

## Fisher Information Matrix

$$I(\theta) = E\left[\left(\frac{\partial}{\partial\theta}\log f(X;\theta)\right)^2 \middle| \theta\right]$$

- Used in Cramer Rao bound (lower bound of the variance of an estimator)
- The epistemic uncertainty is high in sparsely populated regions without training examples.
- The larger the region, the higher the epistemic uncertainty
- Therefore, it can be seen as a confidence region





# IV. Trying to estimate epistemic uncertainty

## Obtaining Fisher Information

$$\theta = \sigma^2$$

$$\lambda \sim \mathrm{N}(0, \sigma^2) \xrightarrow{yields} f_\lambda(\lambda_i, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\lambda_i^2}{2\sigma^2}}$$

$$I(\sigma^2) = \frac{N^4(N-1)}{4\sigma^4} \approx \frac{N^4(N-1)}{4\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i^2 {\lambda_j}^2}$$

Regarding the results, it seems to have failed...

Therefore, it may be useless trying to split uncertainties.

## The results

```
1  fisher_uni = fisher(lambda_uni,n_test)
2  1/fisher_uni
✓ 0.4s
```

```
3.980242134859097e-10
```

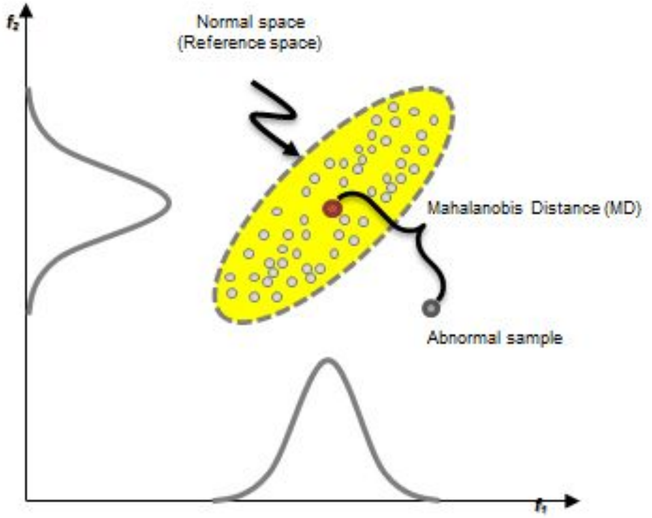
```
1  fisher_gauss = fisher(lambda_gauss,n_test)
2  1/fisher_gauss
✓ 0.5s
```

```
2.3169135765420645e-10
```

# V. Samples classification as functions of uncertainty

## Process

1. Extract instances from images (each instance is a diatom and there can be several in one image)
2. Train a network with the extracted instances
3. Calculate Mahalanobis distances of samples from centroids
4. Identify and rank the diatoms given the confidence returned for each instance



# V. Samples classification as functions of uncertainty

## Let's look at the results

Test on individuals belonging to the NPAE class give **really bad results**…

Very high confidence
(to belong to UULN)

Low confidence
(to belong to NPAE)

Predicted outlier
(while it belongs to NPAE)

# Conclusion

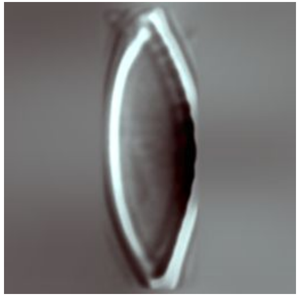
- Uncertainty can be split **empirically** but it seems hard to do it theoretically as shown with Fisher Information Matrix

- Mahalanobis distance is a good way to take into account data covariance but is not always efficient in a latent space

- Disappointing results with the neural network and the framework: **the network is overconfident to predict the wrong class**

- **Hard problem** since :
  - diatoms of different classes can look very similar
  - diatoms of the same class can look very different

**Individuals from class NPAE**



# Thank you for your attention.
# Don't hesitate if you have any questions!

**And thanks, once again, to Cédric Pradalier and Aishwarya Venkataramanan for their help during the whole project**