

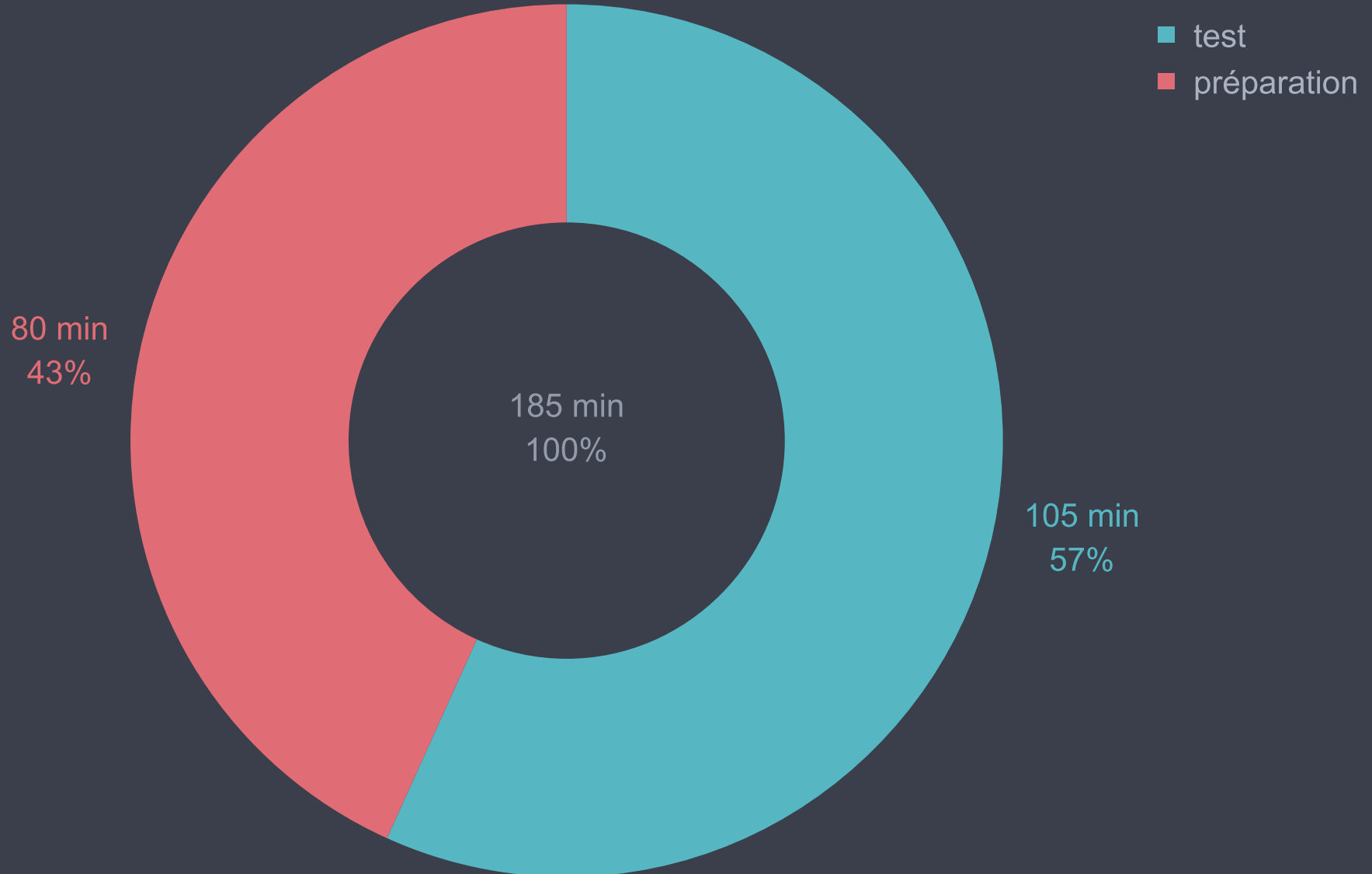
# Normalisation de signaux électromyographiques à l'aide de l'apprentissage automatique

Romain Martinez\*, Jonathan Tremblay, Mickael Begon, Fabien Dal Maso

**JOURNÉE DE LA RECHERCHE 2018**

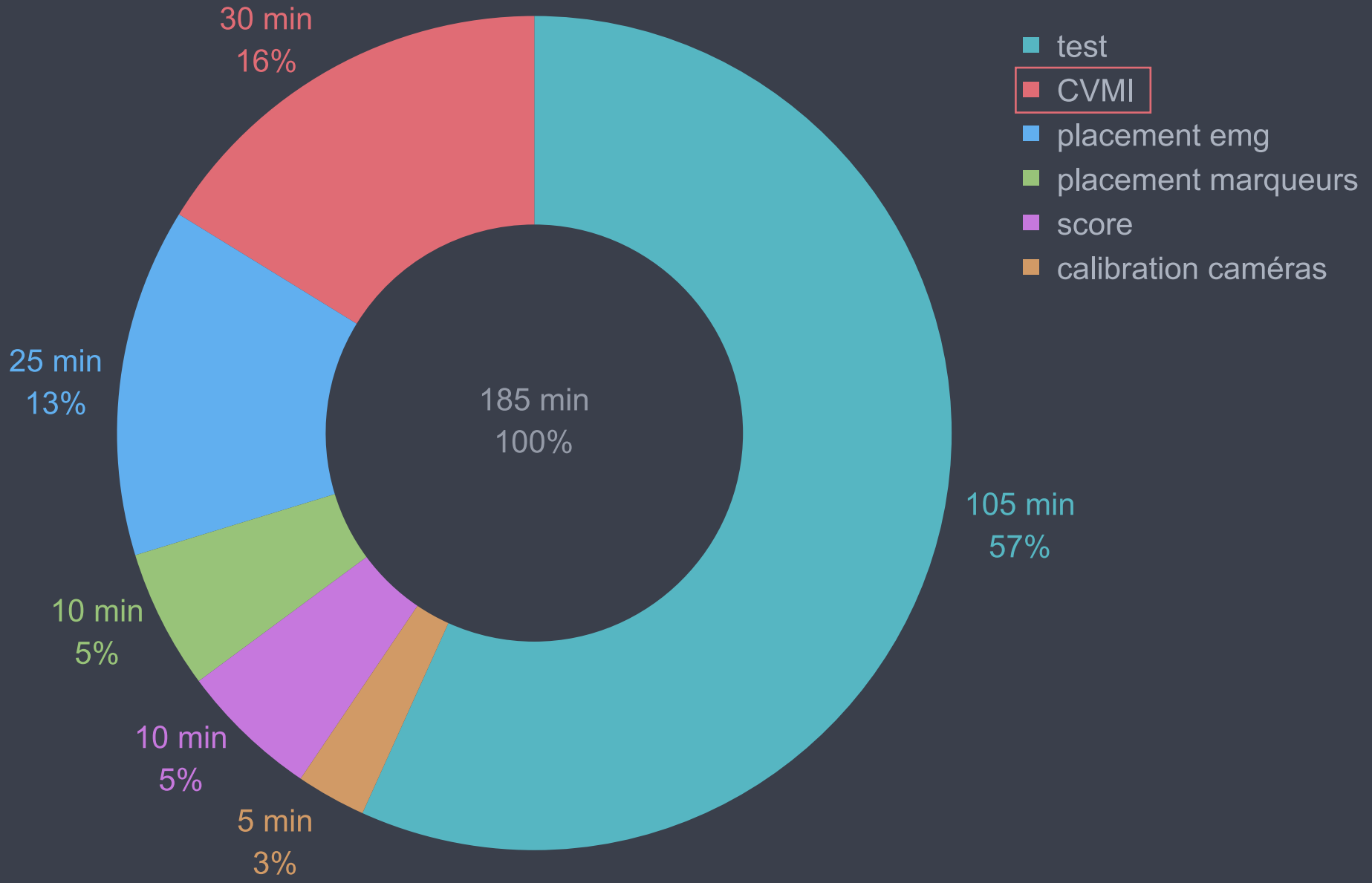
# Introduction

Nous passons trop de temps en préparation



# Introduction

Nous passons trop de temps en CVMI



# Introduction

Qu'est-ce que CVMI?

**C**ontraction **V**olontaire **M**aximale **I**sométrique (CVMI)  
Test utilisé pour normaliser un signal électromyographique

**É**lectro**M**yo**G**raphie (EMG)  
Donne une **idée** de l'activation musculaire

## Attention

L'EMG ne correspond pas directement à l'activation musculaire, mais au courant électrique mesuré sur une fibre particulière



## Normalisation

Nécessaire pour réduire la variabilité inter-participant

# Introduction

## Combinaisons optimales de CVMI

**Problème:** Aucune CVMI ne permet d'obtenir une contraction maximale pour un ensemble de muscles

## Complexe de l'épaule

### Boettcher et al., 2008

Combinaison de 4 tests pour normaliser 12 muscles

Peu de tests nécessaires

Peu représentative

### Dal Maso et al., 2016

Combinaison de 12 tests pour normaliser 12 muscles

Nombre de tests élevé

Représentative

# Introduction

## Objectif

### Boettcher et al., 2008

4 tests pour 12 muscles

Peu de tests nécessaires

Peu représentative

### Dal Maso et al., 2016

12 tests pour 12 muscles

Nombre de tests élevé

Représentative

### Martinez et al., 2018

Approches prédictive

Peu de tests nécessaires

Représentative

# Méthode

Données enregistrées

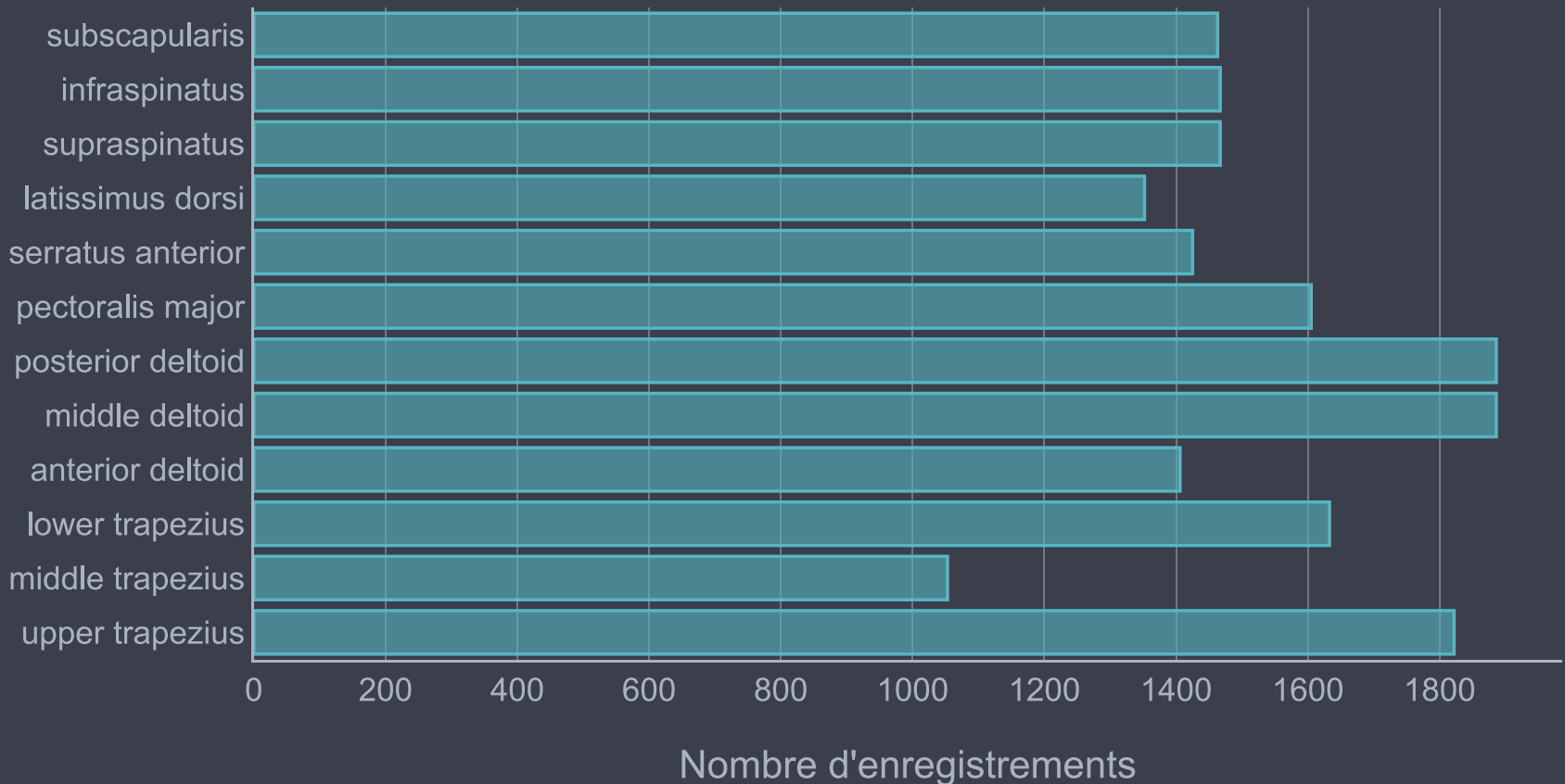
≈ 12 muscles de l'épaules

≈ 16 CVMI

184 participants



**18,465** points



# Méthode

## Organisation des données

Muscles	Participant	Test 1	Test 2	Test 3	...	Test 16
1	1	0.127825	0.124255	0.146927	...	nan
2	1	0.179864	0.294909	0.295846	...	0.199097
3	1	0.078753	nan	0.272709	...	0.106490
...	...	...	...	...	...	...
12	184	0.150353	0.104654	0.115272	...	nan



# Méthode

Données d'entrée et de sortie

## Normalisation

Les valeurs en mV sont normalisées par rapport au test 2

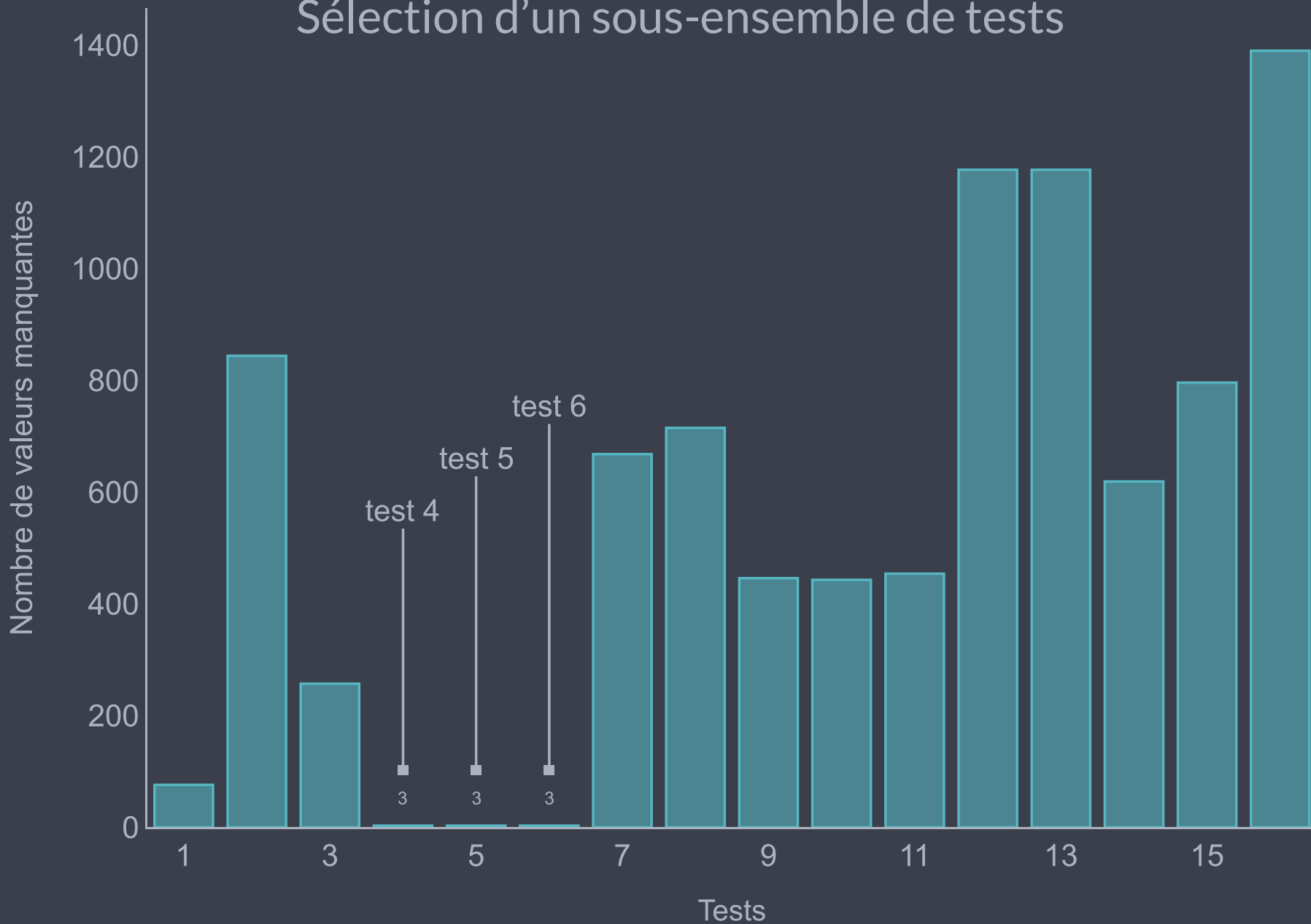
Données d'entrée

Données de sortie

Muscles	Test 1	Test 2	Test 3	...	Test 16	Maximum
1	72	100	130	...	nan	130
2	64	100	91	...	37	100
3	76	100	168	...	43	168
...	...	...	...	...	...	...
12	110	100	57	...	nan	110

# Méthode

## Sélection d'un sous-ensemble de tests



# Méthode

## Positions

- Martinez
- Boettcher
- Dal Maso

CVMI 1



CVMI 2



CVMI 3



CVMI 4



CVMI 5



CVMI 6



CVMI 7



CVMI 8



CVMI 9



CVMI 10



CVMI 11



CVMI 12



CVMI 13



CVMI 14



CVMI 15



CVMI 16



# Méthode

## Choix du modèle

**No Free Lunch** theorem (Wolpert et al., 1997)

Aucun modèle fonctionne bien pour tous types de problèmes

**Faux** en pratique

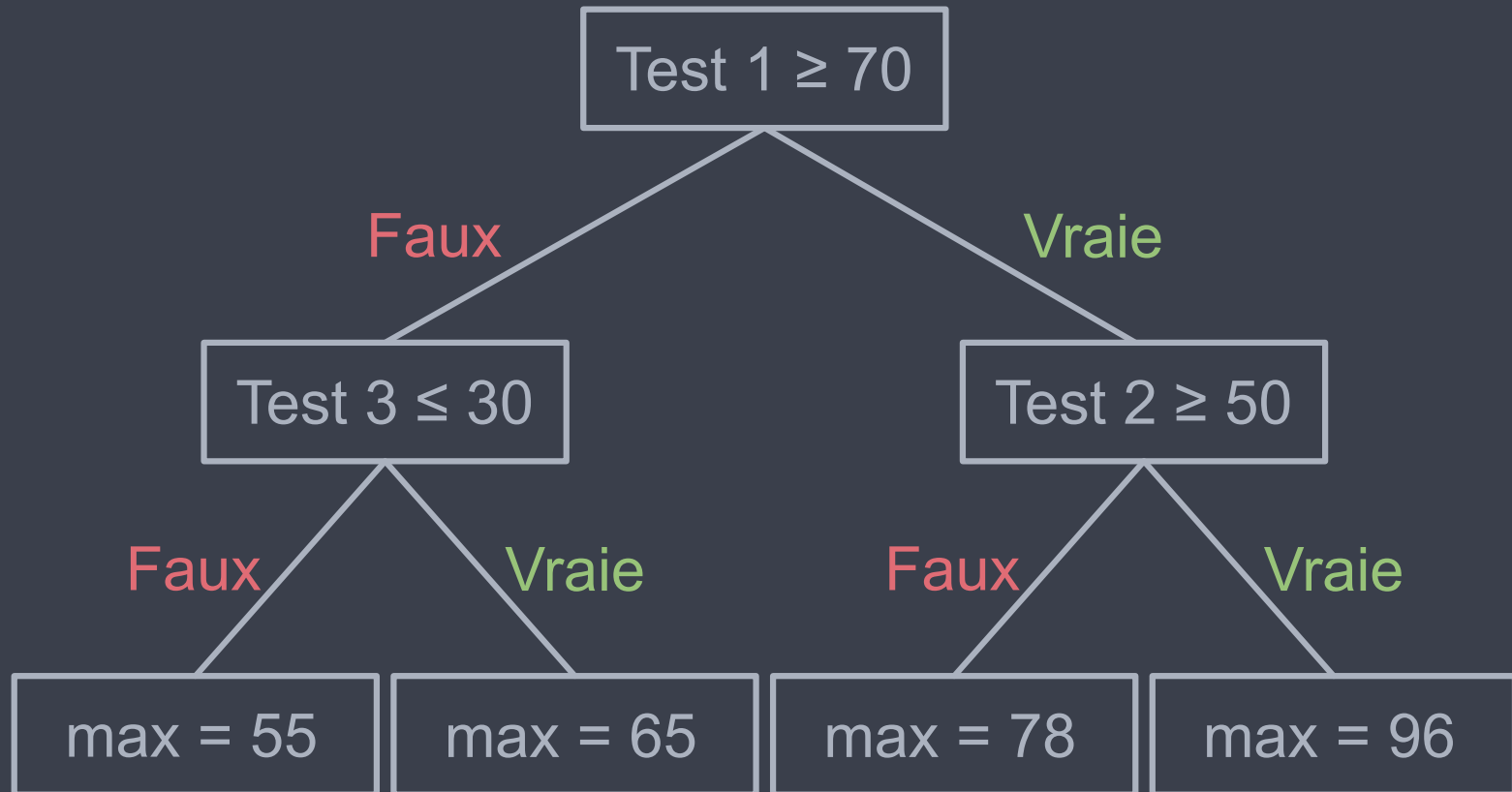
**Forêt aléatoire** (*random forest*)

Bon modèle de départ:

- Algorithme universel
- Pas besoin d'hypothèse sur les données (*distribution normale, relations linéaires*)
- Peu de prétraitements nécessaires

# Méthode

Forêt aléatoire



**Sagesse des foules** (*wisdom of the crowds*)

Un grand nombre d'amateurs répond mieux qu'un expert

# Méthode

## Évaluation du modèle

Ensemble **entraînement -test**

Permet de s'assurer que le modèle généralise bien

80%  
Entraînement

20%  
Évaluation

**Erreur relative**

$$\left| \left| \frac{\text{réel} - \text{prédiction}}{\text{réel}} \right| \right| \times 100$$

4 méthodes: Martinez, Boettcher, Dal Maso, Dal Maso actualisé

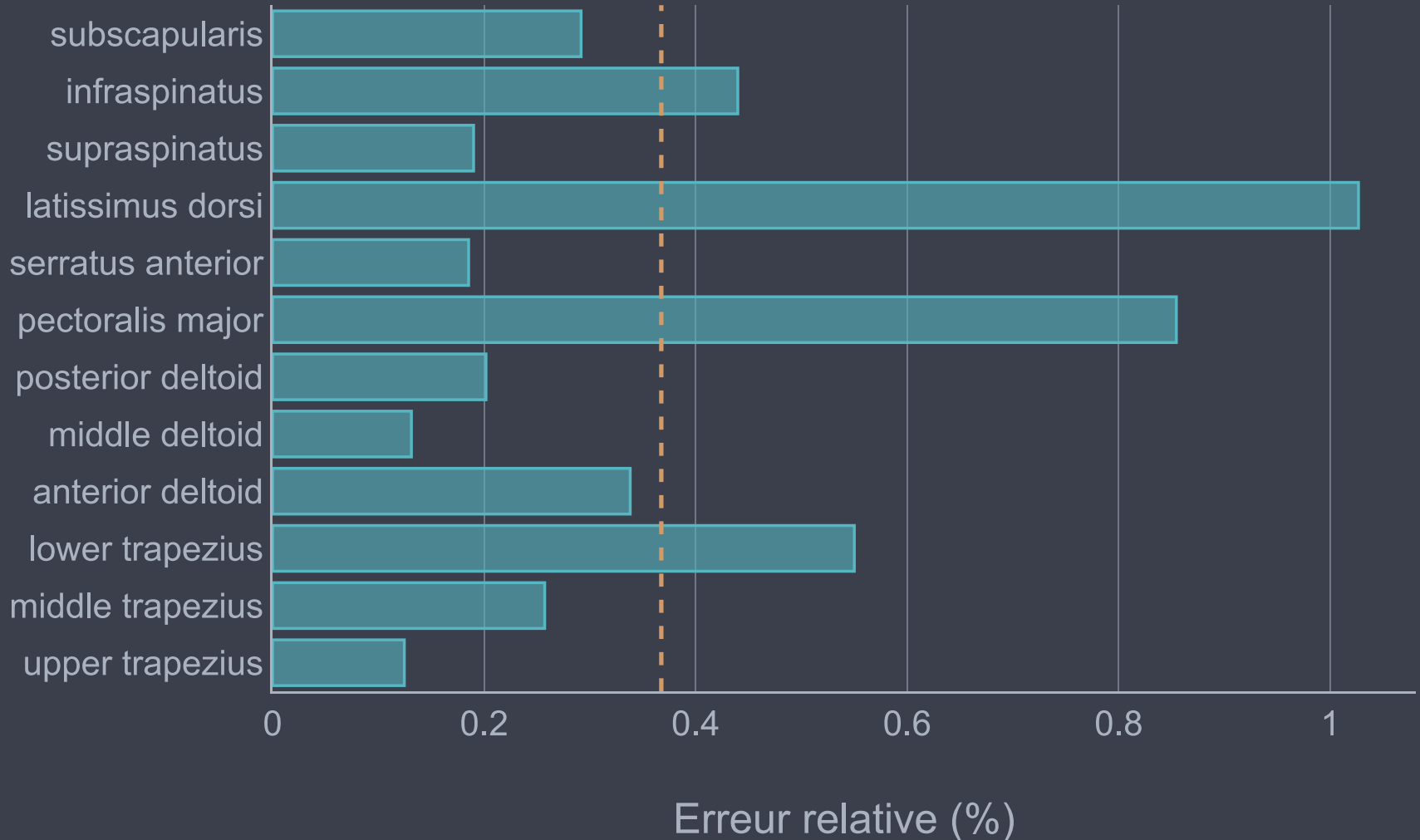
**T-tests non paramétriques**

Pour comparer l'erreur relative de ces différentes méthodes

# Résultats

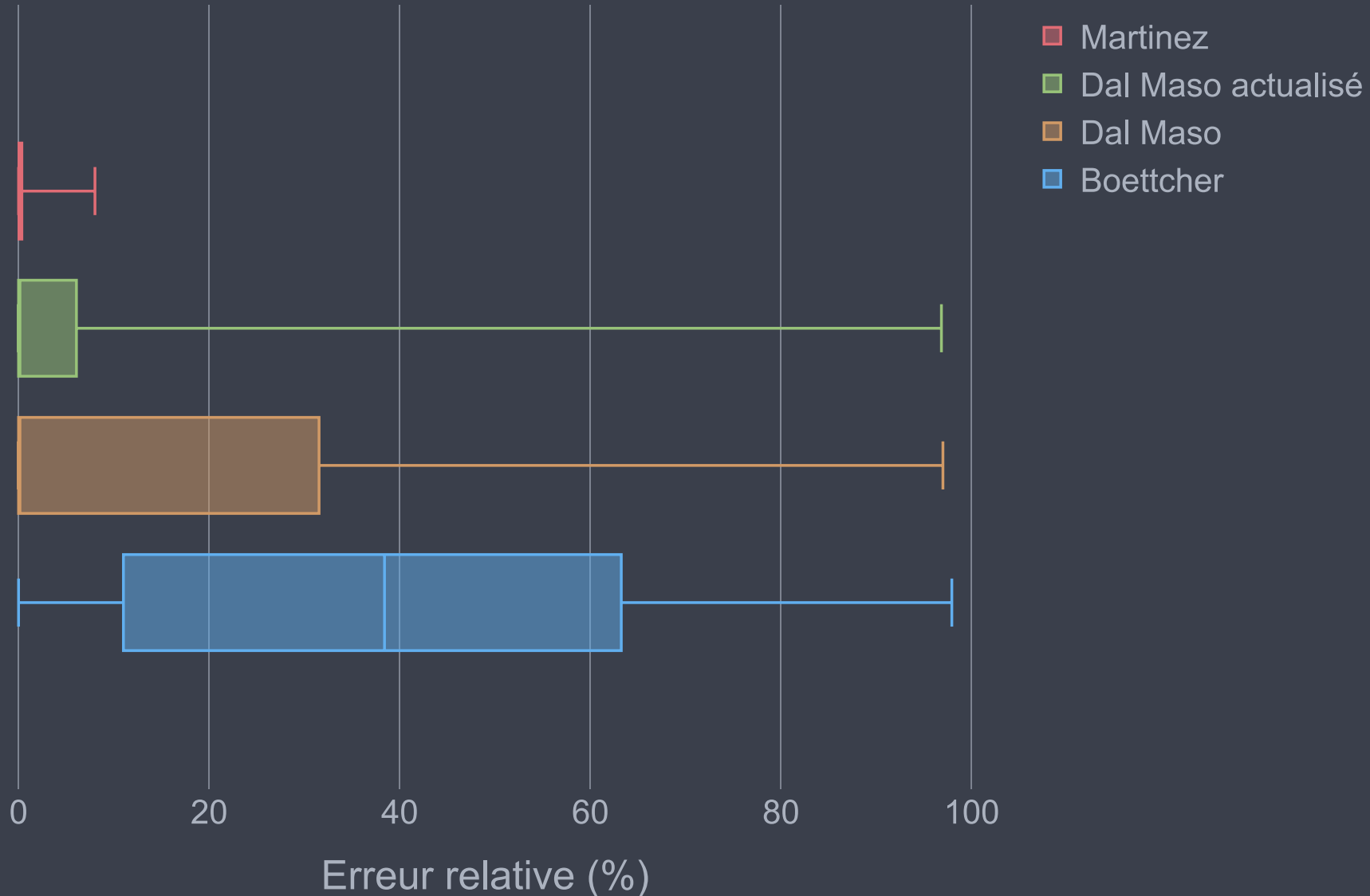
Performance du modèle

Erreur moyenne = 0.37%



# Résultats

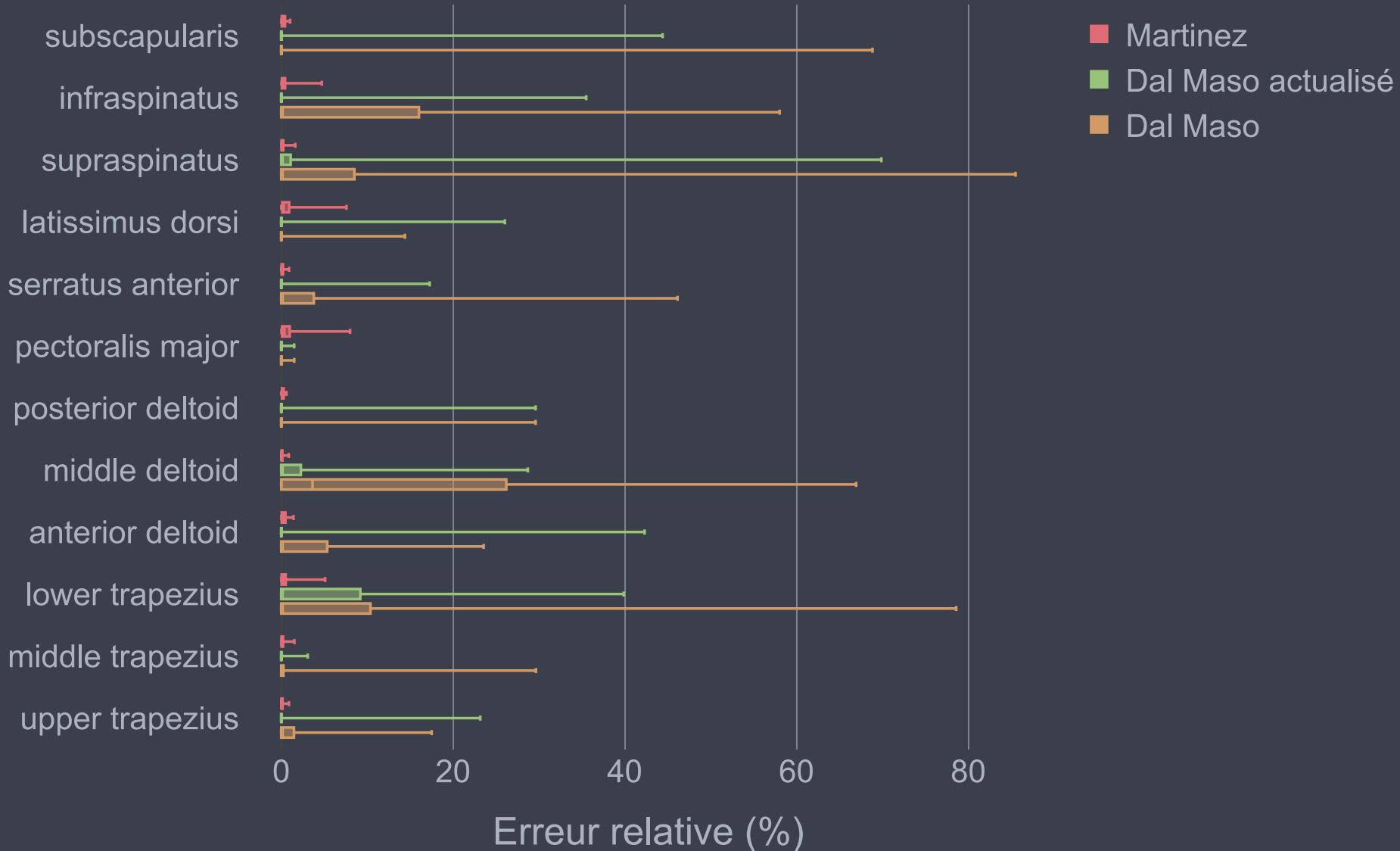
Comparaison avec la littérature





# Résultats

## Comparaison avec la littérature



# Discussion

Gain de temps et de performance

## Gain de temps

3 tests (5 minutes vs. 22 minutes)

## Gain de performance

Erreur relative très faible (0.37% vs. 9.36%)

Méthode	Erreur relative (%)	Temps d'exécution (min)
Boettcher	40.30	6.67
Dal Maso	20.00	22
Dal Maso actualisé	9.36	22
Martinez	0.37	4.7

# Discussion

Application pratique

## Utilisation

Modèle pré-entraîné disponible pour la communauté

1. Télécharger le modèle pré-entraîné

[github.com/romainmartinez/mvc/](https://github.com/romainmartinez/mvc/)

2. Installer *Python* et la librairie *Scikit-Learn*

3. Faire la prédiction depuis le modèle

```
from sklearn.externals import joblib
```

```
X = joblib.load('my_data.pkl')
```

```
model = joblib.load('model.pkl')
```

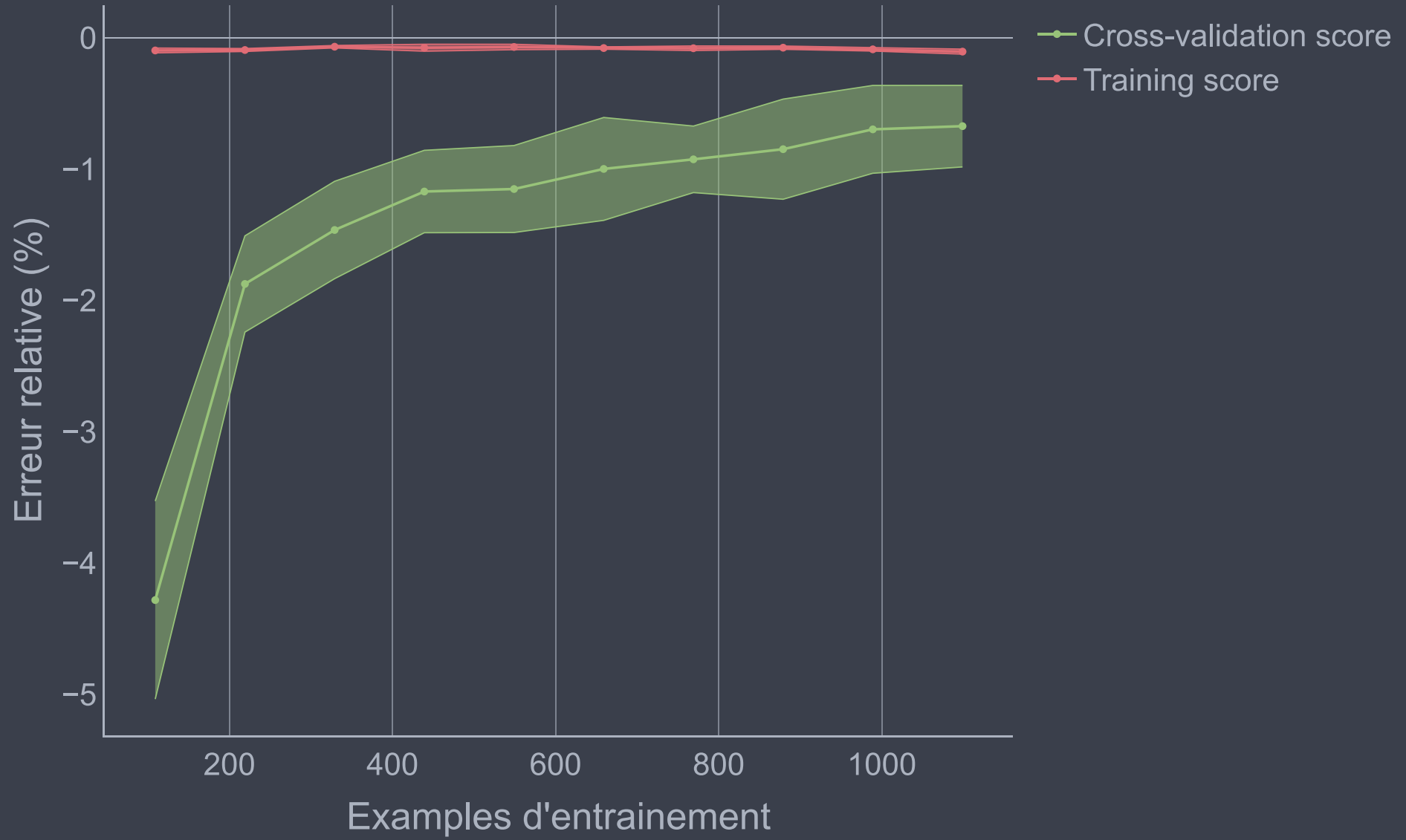
```
predictions = model.predict(X)
```

# Normalisation de signaux électromyographiques à l'aide de l'apprentissage automatique

Romain Martinez\*, Jonathan Tremblay, Mickael Begon, Fabien Dal Maso

**JOURNÉE DE LA RECHERCHE 2018**

# Bonus



# Bonus

