



Initiation à l'Intelligence Artificielle

Pr. Mohammed AMEKSA

Ancien professeur à l'École Marocaine des Sciences de l'Ingénieur - EMSI

Enseignant Chercheur à la Faculté des Sciences Semlalia – FSSM

E-mail :

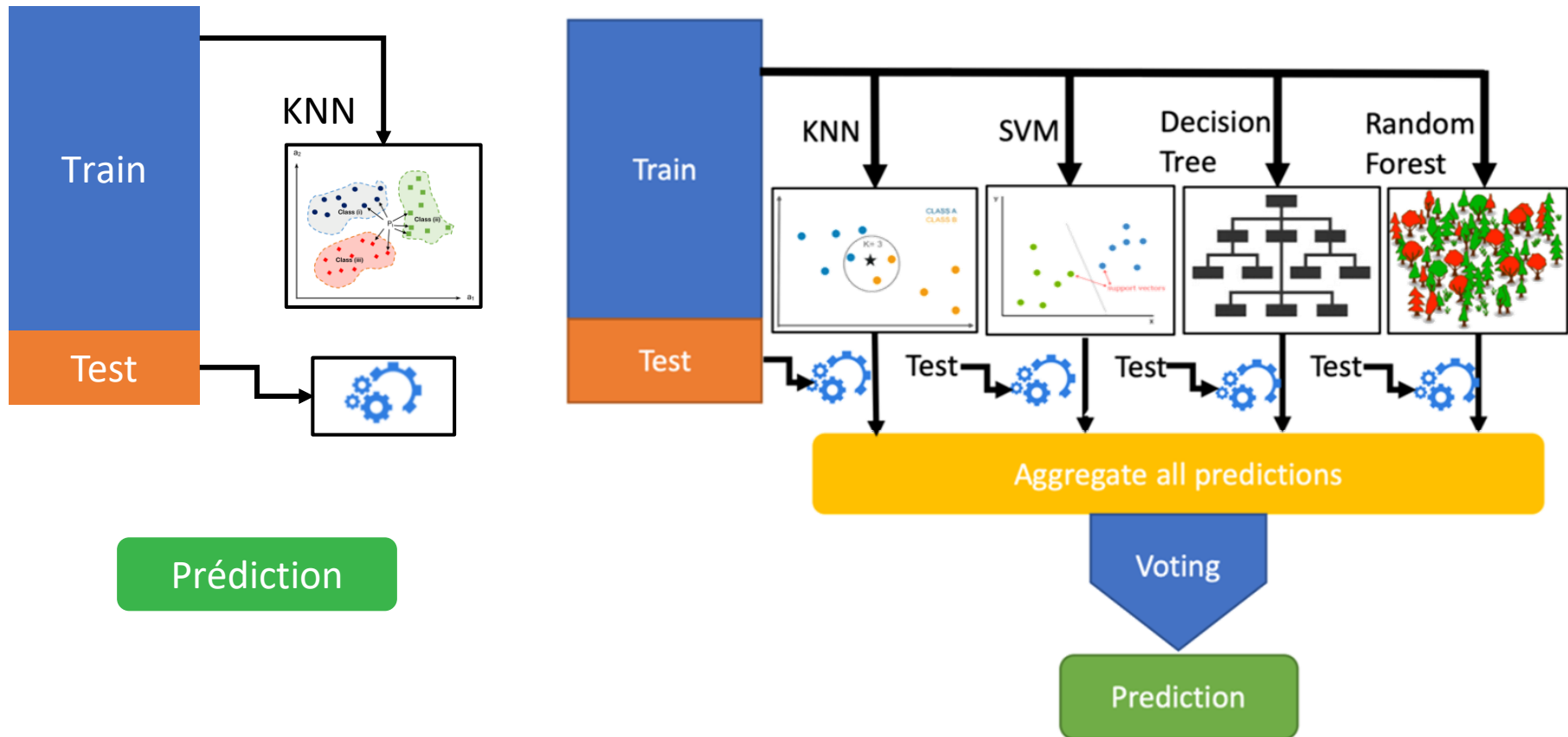
info.ameksa@gmail.com

Random Forest – RF



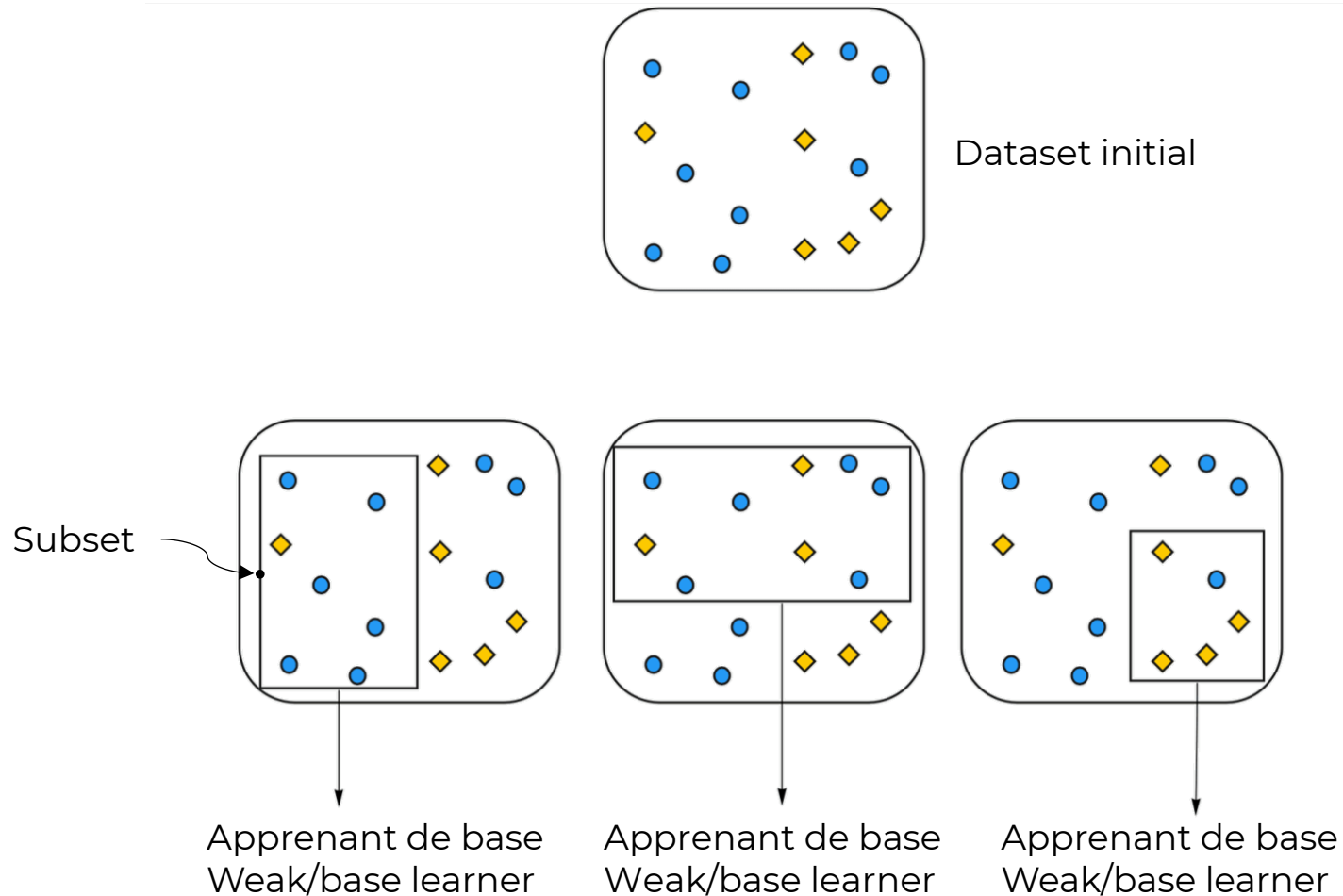
2. Comprendre l'algorithme random forest (RF)

ENSEMBLISTE



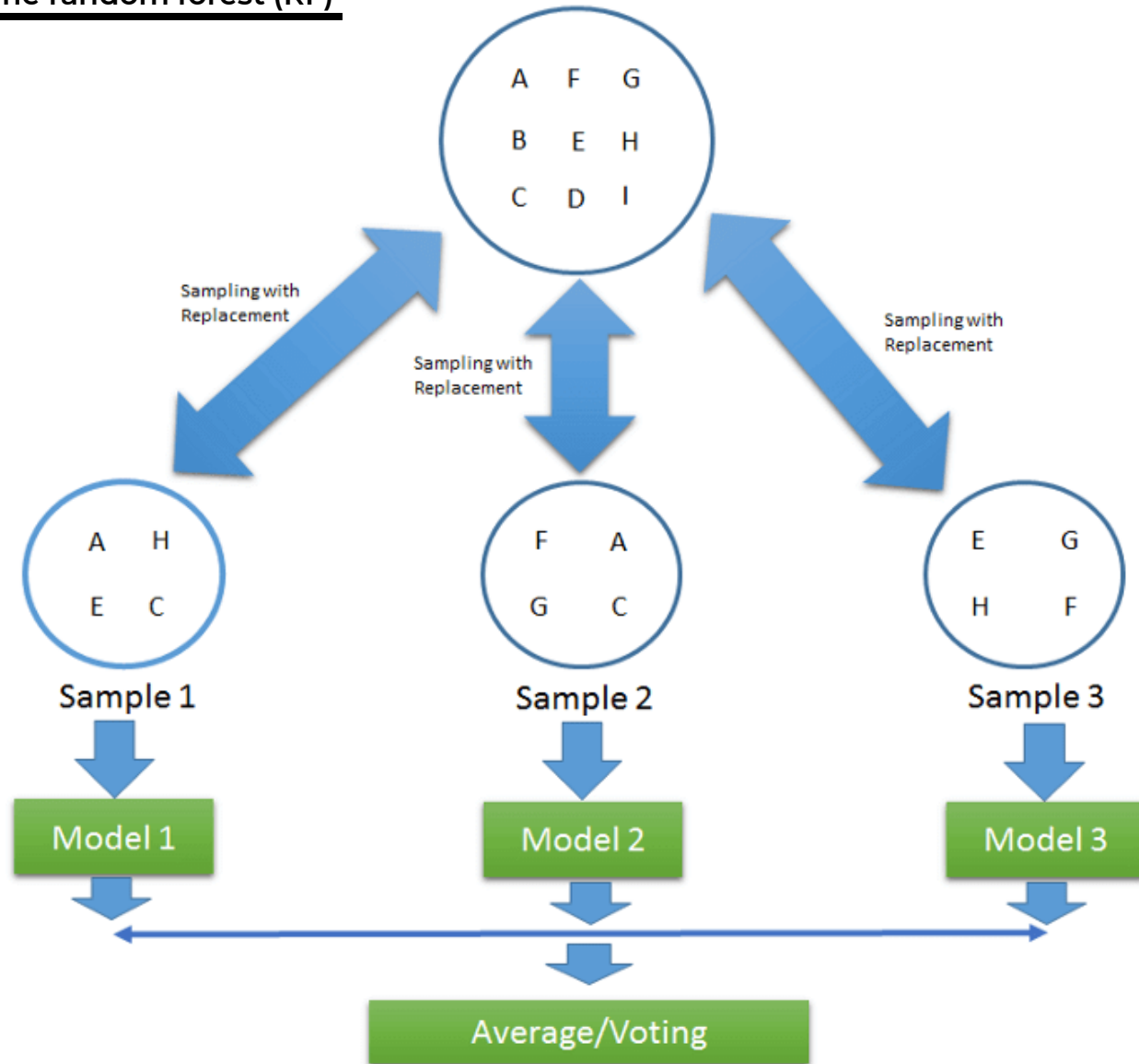
2. Comprendre l'algorithme random forest (RF)

BAGGING



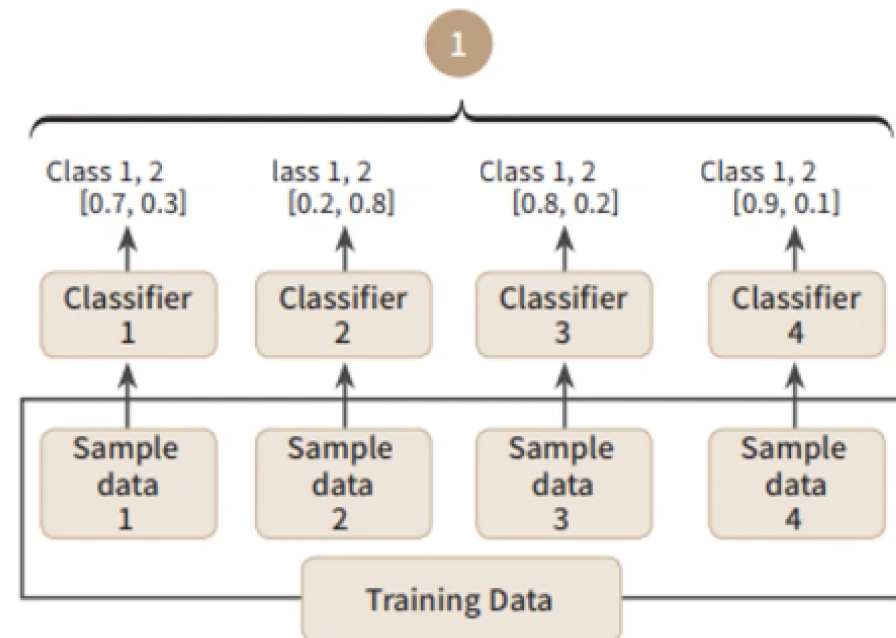
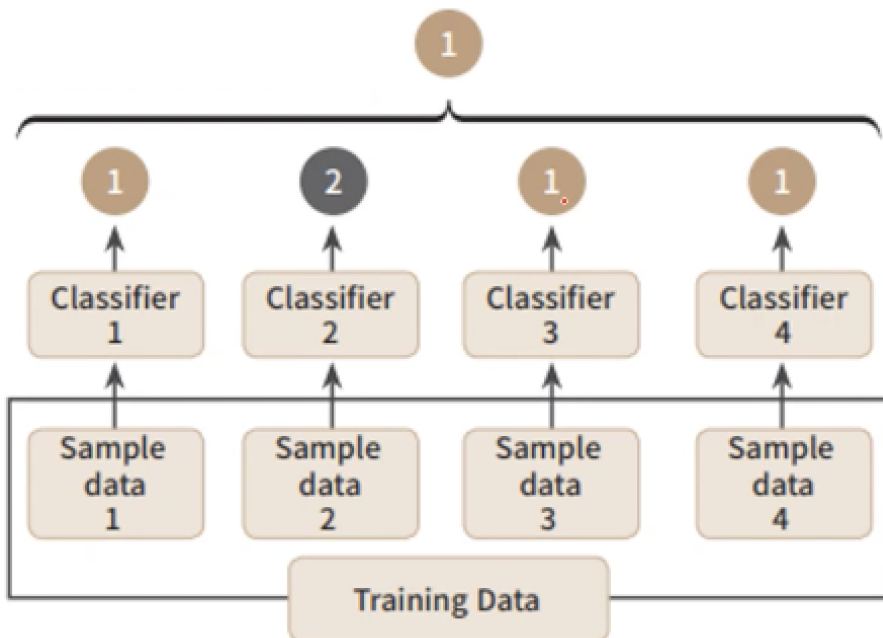
2. Comprendre l'algorithme random forest (RF)

BAGGING



2. Comprendre l'algorithme random forest (RF)

BAGGING



2. Comprendre l'algorithme random forest (RF)

BAGGING : Syntaxe Hands ON



Importation

```
from sklearn.ensemble import BaggingClassifier
```

Création du modèle

```
BC = BaggingClassifier(n_estimators=50)
```

Entrainement

```
BC = BC.fit(X_train, y_train)
```

Faire des prédictions

```
y_predict = BC.predict(X_test)
```


2. Comprendre l'algorithme random forest (RF)

Random Forest

Étape 1 : Création d'un ensemble de données **bootstrappées**

échantillons de données avec remplacement

Dataset initial

Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
--------	-------	--------	-----	---------------

2. Comprendre l'algorithme random forest (RF)

Random Forest

Étape 1 : Création d'un ensemble de données **bootstrappées**
échantillons de données avec remplacement

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes

→ Pour créer un ensemble de données « bootstrapped » de même taille que l'original, il suffit de sélectionner des échantillons dans l'ensemble de données original.



Nous sommes autorisés à choisir le même échantillon plus d'une fois

2. Comprendre l'algorithme random forest (RF)

Random Forest

Étape 2 : créer un **arbre de décision** à l'aide de l'ensemble de données **bootstrappées**, tout en utilisant un **sous-ensemble aléatoire de variables** (ou de colonnes) à chaque étape.

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes

→ Dans cet exemple nous considérons 2 variables / étape



Il existe des règles pour bien choisir le nombre optimal des variables à considérer

2. Comprendre l'algorithme random forest (RF)

Random Forest

Étape 2 : créer un **arbre de décision** à l'aide de l'ensemble de données **bootstrappées**, tout en utilisant un **sous-ensemble aléatoire de variables** (ou de colonnes) à chaque étape.

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes

→ Donc, au lieu de considérer les 4 variables pour déterminer comment diviser le nœud racine ...



2. Comprendre l'algorithme random forest (RF)

Random Forest

Étape 2 : créer un **arbre de décision** à l'aide de l'ensemble de données **bootstrappées**, tout en utilisant un **sous-ensemble aléatoire de variables** (ou de colonnes) à chaque étape.

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes

→ ... nous sélectionnons au hasard 2



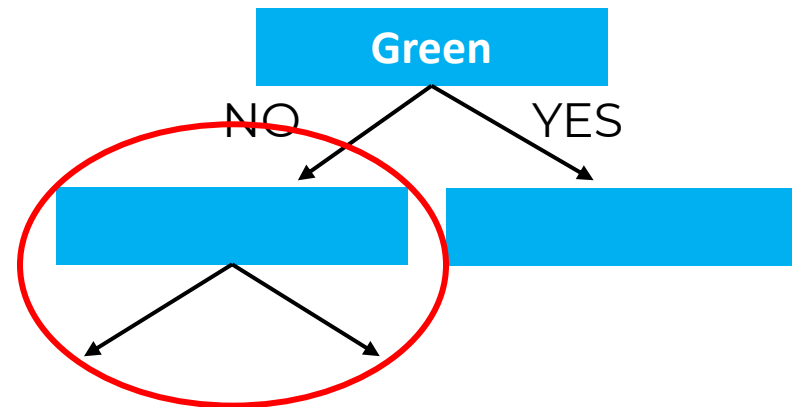
2. Comprendre l'algorithme random forest (RF)

Random Forest

Étape 2 : créer un **arbre de décision** à l'aide de l'ensemble de données **bootstrappées**, tout en utilisant un **sous-ensemble aléatoire de variables** (ou de colonnes) à chaque étape.

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes



→ Maintenant, nous devons déterminer comment diviser les échantillons au niveau de ce nœud

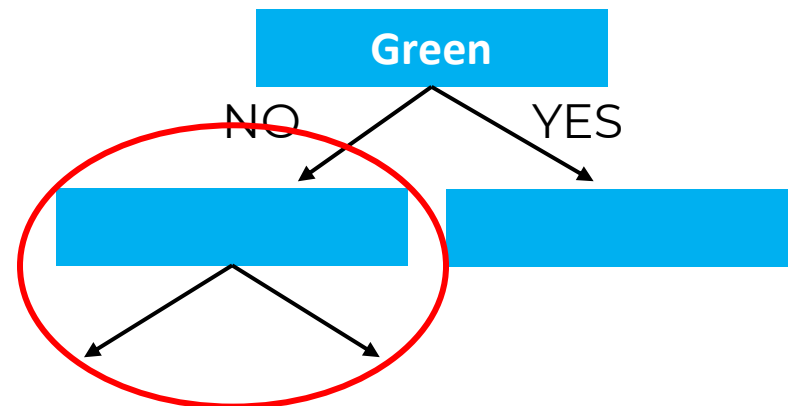
2. Comprendre l'algorithme random forest (RF)

Random Forest

Étape 2 : créer un **arbre de décision** à l'aide de l'ensemble de données **bootstrappées**, tout en utilisant un **sous-ensemble aléatoire de variables** (ou de colonnes) à chaque étape.

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes



→ De la même manière que pour la racine, nous sélectionnons au hasard 2 variables comme candidates, au lieu des 3 colonnes restantes.

2. Comprendre l'algorithme random forest (RF)

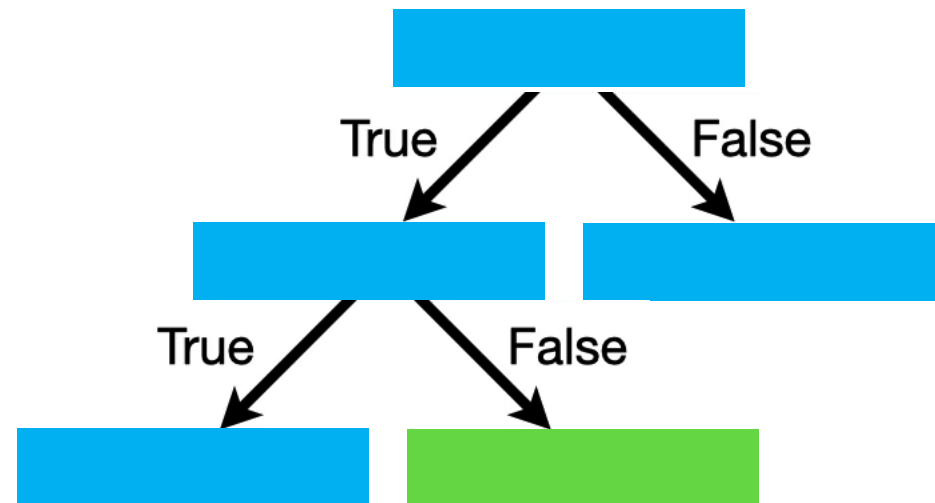
Random Forest

Étape 2 : créer un **arbre de décision** à l'aide de l'ensemble de données **bootstrappées**, tout en utilisant un **sous-ensemble aléatoire de variables** (ou de colonnes) à chaque étape.

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes

→ Et nous construisons l'arbre comme d'habitude, mais en ne considérant qu'un sous-ensemble aléatoire de variables à chaque étape



2. Comprendre l'algorithme random forest (RF)

Random Forest

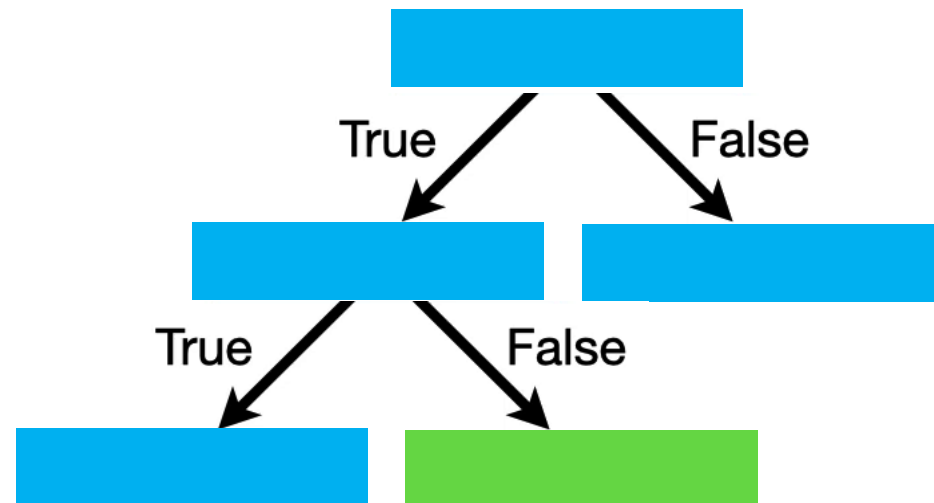
Récap :

nous construisons un arbre :

1. à l'aide d'un ensemble de données bootstrapées
2. en ne considérant qu'un sous-ensemble aléatoire de variables à chaque étape

Bootstrapped dataset

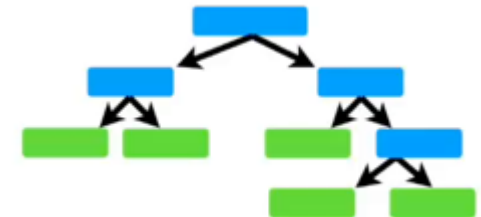
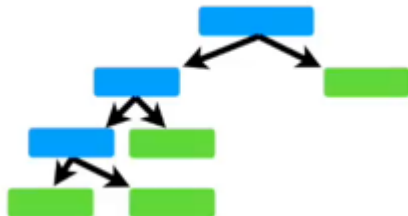
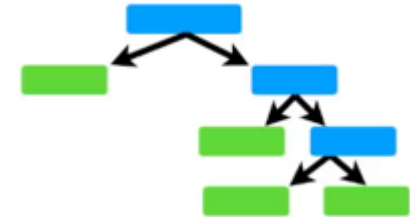
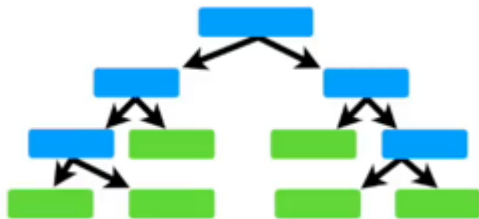
Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes



2. Comprendre l'algorithme random forest (RF)

Random Forest

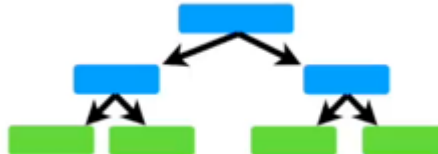
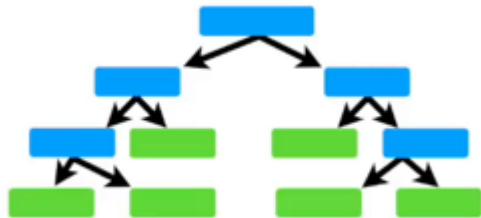
Ensuite, revenez à l'étape 1 et répétez l'opération : créez un nouvel ensemble de données bootstrappé et construisez un arbre en considérant un sous-ensemble de variables à chaque étape.



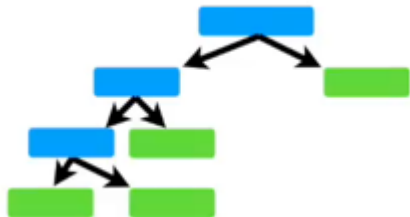
2. Comprendre l'algorithme random forest (RF)

Random Forest

idéalement, vous feriez cela des centaines de fois, mais ici, juste pour avoir une idée



La variété est ce qui rend les algorithmes RF plus efficaces que les arbres de décision individuels.



Comment l'utiliser ?

2. Comprendre l'algorithme random forest (RF)

Random Forest

- Pour commencer, nous recevons une nouvelle plante
- dont nous avons pris toutes les mesures ...
- et nous souhaitons savoir si elle est malade ou non.

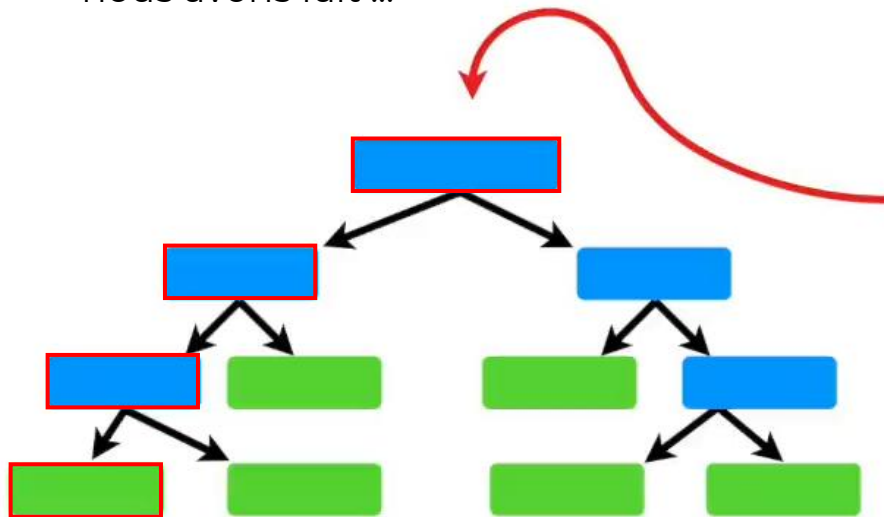
Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	No	No	168	???

2. Comprendre l'algorithme random forest (RF)

Random Forest

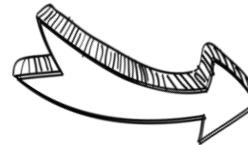
→ Nous prenons donc les données et nous les faisons descendre dans le premier arbre que nous avons fait ...



Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
YES	NO	NO	168	

le premier arbre dit « oui »



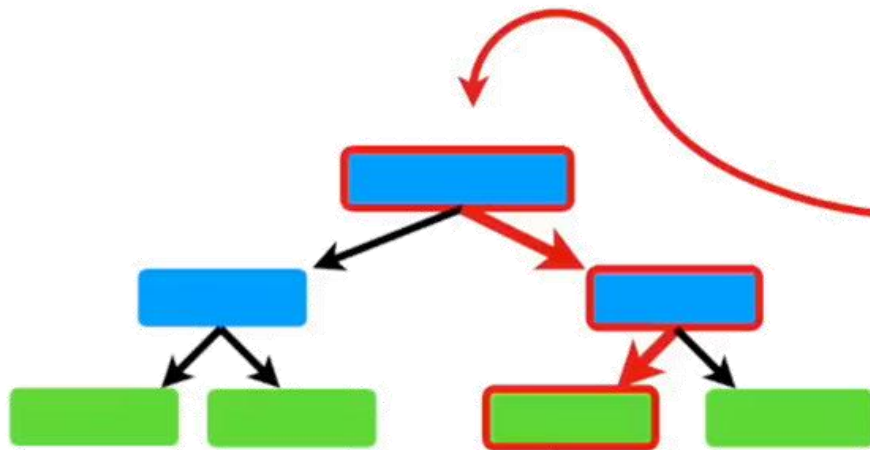
→ Nous gardons le suivi de cela ici

Plant Disease	
YES	NO
1	0

2. Comprendre l'algorithme random forest (RF)

Random Forest

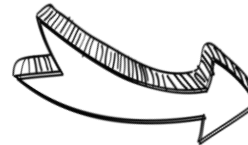
→ Nous prenons donc les données et nous les faisons descendre dans le 2^{ème} arbre ...



Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
YES	NO	NO	168	

2^{ème} arbre dit aussi « oui »



Plant Disease	
YES	NO
2	0

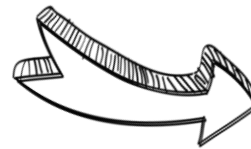
2. Comprendre l'algorithme random forest (RF)

Random Forest

- Après avoir parcouru les données dans tous les arbres de la forêt aléatoire, nous voyons quelle est l'option qui recueille le plus de votes.

Bootstrapped dataset

Stinky	Green	Indoor	Age	Plant Disease
YES	NO	NO	168	YES



Plant Disease	
YES	NO
5	1


- Dans ce cas, c'est le « YES » qui a recueilli le plus grand nombre de votes, ce qui nous permet de conclure que cette plante est malade.

2. Comprendre l'algorithme random forest (RF)

Random Forest

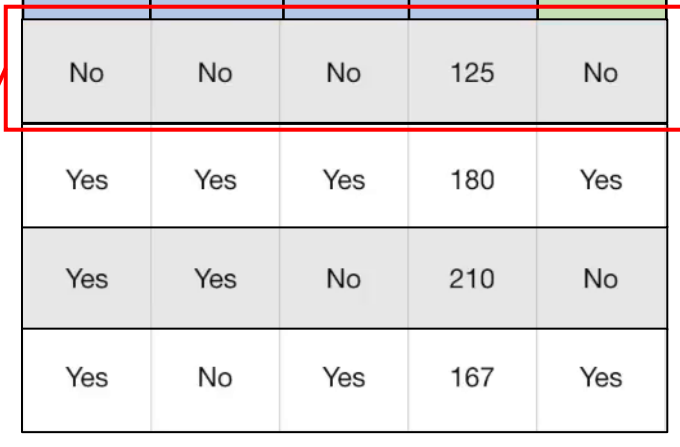
→ nous avons autorisé des doublons dans l'ensemble de données bootstrappées ...

Bootstrapped dataset



Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes
Yes	Yes	Yes	180	Yes

Dataset initial



Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes

- Par conséquent, cette entrée n'a pas été intégrée dans l'ensemble de données bootstrappé.
- En général, environ 1/3 des données originales ne se retrouve pas dans l'ensemble de données obtenu.

2. Comprendre l'algorithme random forest (RF)

Random Forest

- Voici l'entrée qui n'a pas été retenue dans l'ensemble de données bootstrappé
- c'est ce qu'on appelle le « Out-Of-Bag Dataset »

Out-Of-Bag Dataset

Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No

Dataset initial

Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes

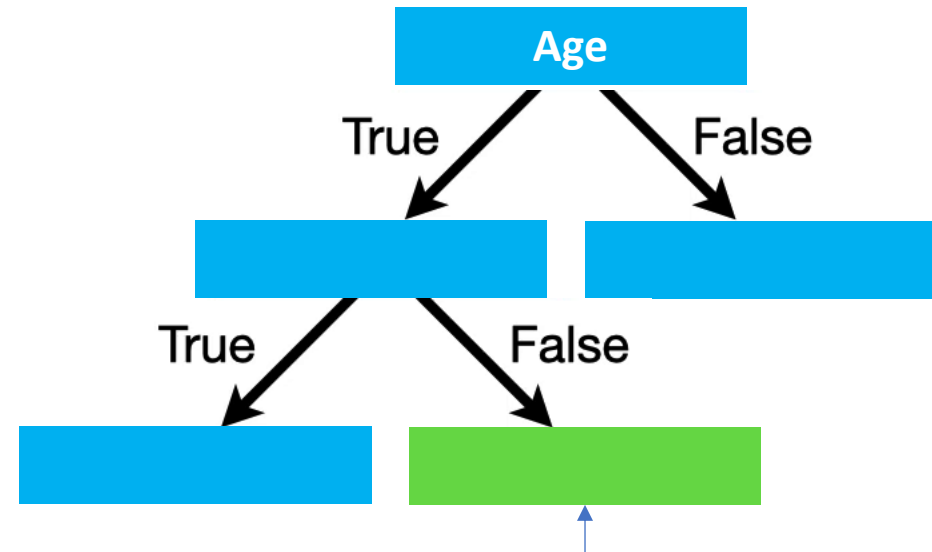
2. Comprendre l'algorithme random forest (RF)

Random Forest

- Étant donné que les données Out-Of-Bag n'ont pas été utilisées pour créer cet arbre
- nous pouvons l'exécuter et voir s'il permet de classer l'échantillon dans la catégorie « Absence de maladie des plantes »

Out-Of-Bag Dataset

Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No



- Dans ce cas, l'arbre étiquette correctement l'échantillon « Out of Bag » comme étant « No ».

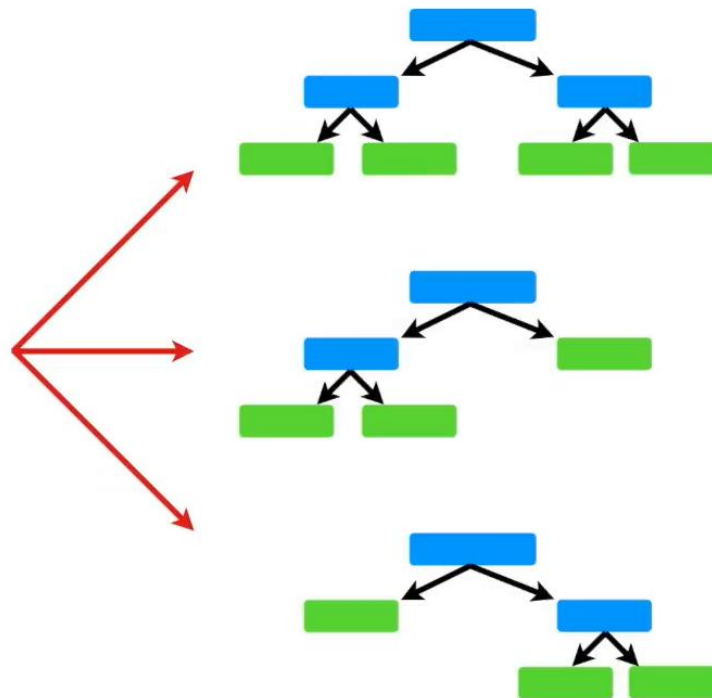
2. Comprendre l'algorithme random forest (RF)

Random Forest

→ puis nous passons cet échantillon « out of bag » à travers tous les autres arbres qui ont été construits sans lui.

Out-Of-Bag Dataset

Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No



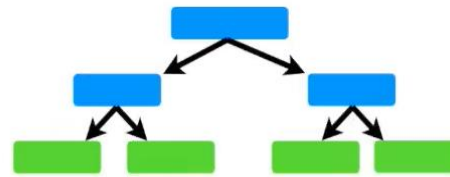
2. Comprendre l'algorithme random forest (RF)

Random Forest

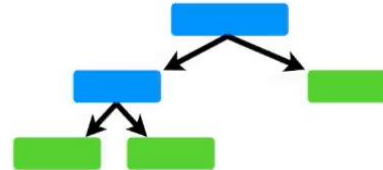
→ puis nous passons cet échantillon « out of bag » à travers tous les autres arbres qui ont été construits sans lui.

Out-Of-Bag Dataset

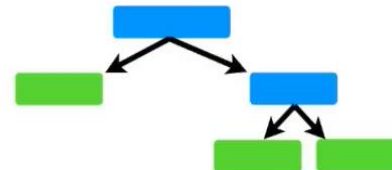
Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No



→ cet arbre a incorrectement étiqueté l'échantillon



→ cet arbre a correctement étiqueté l'échantillon



→ cet arbre a correctement étiqueté l'échantillon

2. Comprendre l'algorithme random forest (RF)

Random Forest

- puis nous passons cet échantillon « out of bag » à travers tous les autres arbres qui ont été construits sans lui.

Out-Of-Bag Dataset

Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No

Classification of the Out-Of-Bag sample	
Yes	No
1	3

- Puisque l'étiquette qui obtient le plus grand nombre de votes gagne, c'est l'étiquette que nous attribuons à l'échantillon out-of-bag
- Dans ce cas, l'échantillon out-of-bag est correctement étiqueté par la Random Forest.

2. Comprendre l'algorithme random forest (RF)

Random Forest

Out-Of-Bag Dataset

Stinky	Green	Indoor	Age	Plant Disease
No	No	No	125	No

Classification of the Out-Of-Bag sample	
Yes	No
1	3

→ nous faisons ensuite la même chose pour tous les autres échantillons « out of bag » pour tous les arbres.

2. Comprendre l'algorithme random forest (RF)

Random Forest

Out-Of-Bag Dataset

Stinky	Green	Indoor	Age	Plant Disease
YES	YES	YES	180	YES

Classification of the
Out-Of-Bag sample

Yes

No

1

3

Classification of the
Out-Of-Bag sample

Yes

No

4

0

→ Cet échantillon a également été correctement étiqueté

Stinky	Green	Indoor	Age	Plant Disease
NO	NO	NO	125	NO

Classification of the
Out-Of-Bag sample

Yes

No

3

1

→ Cet échantillon a a été incorrectement étiqueté.

2. Comprendre l'algorithme random forest (RF)

Random Forest

- En fin de compte, nous pouvons mesurer la précision de notre RF par la proportion d'échantillons Out-Of-Bag qui ont été correctement classés.
- La proportion d'échantillons Out-Of-Bag qui ont été incorrectement classés est « l'erreur Out-Of-Bag »

Classification of the Out-Of-Bag sample

Yes	No
1	3

Classification of the Out-Of-Bag sample

Yes	No
4	0

Classification of the Out-Of-Bag sample

Yes	No
3	1

etc... etc... etc...

2. Comprendre l'algorithme random forest (RF)

Random Forest

Récap :

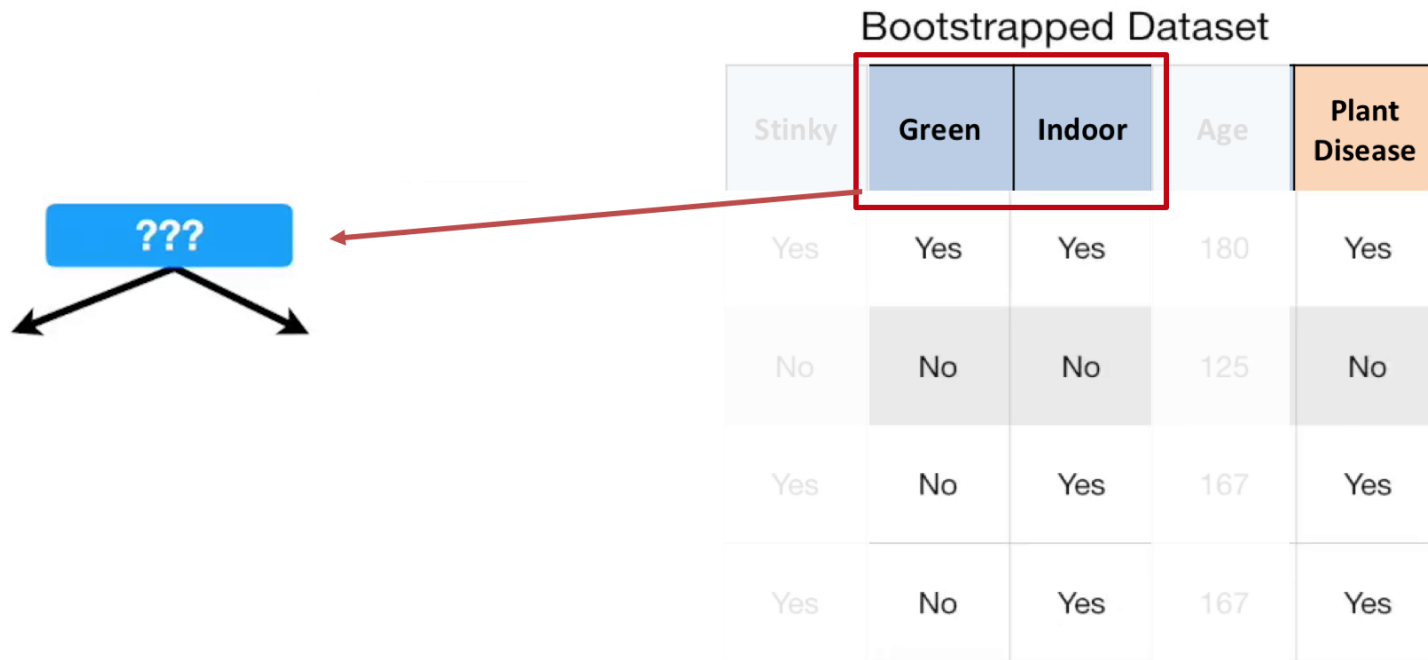
1. Construire un modèle Random Forest
2. Utiliser ce type des modèles
3. estimer la précision de ce modèle

Nous pouvons approfondir notre discussion sur la façon de faire **le 1^{er} point.**

2. Comprendre l'algorithme random forest (RF)

Random Forest

→ nous sélectionnons au hasard 2 variables



→ Nous pouvons maintenant comparer l'erreur Out-Of-Bag pour une « random forest » construite en utilisant seulement 2 variables par étape

2. Comprendre l'algorithme random forest (RF)

Random Forest



Bootstrapped Dataset

Stinky	Green	Indoor	Age	Plant Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

- pour construire le modèle random forest à l'aide de 3 variables par étape
- et nous testons un certain nombre de configurations différentes et choisissons le modèle le plus précise.

2. Comprendre l'algorithme random forest (RF)

Random Forest : Hands On

```
from sklearn.ensemble import RandomForestClassifier

RC = RandomForestClassifier(n_estimators=100, max_features=10)

RC = RC.fit(X_train, y_train)

y_predict = RC.predict(X_test)
```

3. Hands-On: Live Coding Demonstration with Python and Scikit-Learn

Hands On



2. **Chargement des données** : Importer et charger le jeu de données
3. **Prétraitement des données** : Nettoyer, transformer et préparer les données pour l'entraînement.
4. **Entraînement du modèle** : entraîner Random Forest à l'aide des données prétraitées.
5. **Évaluation du modèle** : Évaluer les performances en calculant le MSE de chaque modèle entraîné et effectuer des ajustements.
6. **Sauvegarde du modèle entraîné.**
7. **Déploiement du modèle** : Utiliser le framework Django pour exploiter les modèles entraînés.

3. Hands-On: Live Coding Demonstration with Python and Scikit-Learn

Hands On



Questions à réfléchir

1. Quel modèle donne la meilleure précision entre Bagging et RF ? Pourquoi ?
2. En quoi RF améliore le simple Bagging ?
3. Essayez de visualiser l'importance des features (feature_importances_) avec le modèle RF.
4. Changez le nombre d'estimateurs (par exemple 10, 100, 500) pour observer l'impact.