

Machine Learning for Medicine TP1

Decision Trees and Random Forest

The goal of the TME is to develop practical skills to use decision trees and random forest for real biological applications.

We will use the *scikit-learn Python* library <http://scikit-learn.org> which is already installed on the computers.

Data

- Diabetes Remission Prediction. The problem is to predict whether a diabetic patient will resolve or will not resolve his diabetes after a gastric bypass surgery.
 1. `patients_data.txt` – Observations: 200 patients, 4 clinical variables: age of patients (continuous), HbA1C (continuous), insuline taken (categorical, yes or not), other anti-diabetic drugs are taken (categorical, yes or not)
 2. `patients_classes.txt` – Classes: 0 (Diabetes Remission) and 1 (Non-Remission) for 200 patients

Libraries

You will need to load the following packages:

```
import pandas as pd
import graphviz
from sklearn import tree
from sklearn.ensemble import RandomForestClassifier
```

Analysis

Read the data

```
data_diabetes = pd.read_table('patients_data.txt',sep='\t',header=None)
classes_diabetes = pd.read_table('patients_classes.txt',sep='\t',header=None)
```

1. Decision trees

- You can learn more about decision trees in Python here:
<http://scikit-learn.org/stable/modules/tree.html>
- Run the classifier to learn a model

```
clf = tree.DecisionTreeClassifier()
clf = clf.fit(data_diabetes, classes_diabetes)
```
- Visualize the tree and save it as a .pdf

```
feature_names = ['age', 'hba1c', 'insuline taken', 'other drugs taken']
classes = ['DR', 'NDR']
dot_data = tree.export_graphviz(clf, out_file=None,
                                feature_names=feature_names,
                                class_names=classes,
```

```

        filled=True, rounded=True,
        special_characters=True)
graph = graphviz.Source(dot_data)
graph.render("diabetes remission")

```

2. Random forest

- You can learn more about the Random Forest in Python:

<http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

- To estimate a model:

```

clf = RandomForestClassifier(max_depth=2, random_state=0)
clf.fit(data_diabetes, classes_diabetes)

```

- To make prediction with the random forest:

```

clf.predict(data_diabetes)

```

- To plot the influence of each variable in the model:

```

clf.feature_importances_

```

3. Interpretable Models: comparison with the state-of-the-art clinical score DiaRem

The DiaRem (Diabetes Remission score) was introduced recently by *Still et al., 2013* (see the references below), and can be summarized by the following table:

	Score
Age (years)	
<40	0
40-49	1
50-59	2
≥60	3
HbA_{1c} (%)	
<6.5%	0
6.5-6.9%	2
7.0-8.9%	4
≥9.0%	6
Other diabetes drugs	
No sulfonylureas or insulin-sensitising agent other than metformin	0
Sulfonylureas and insulin-sensitising agent other than metformin	3
Treatment with insulin	
No	0
Yes	10
Total score calculated by adding scores for each of the four variables.	
Table 5: Calculation of DiaRem score for prediction of the probability of diabetes remission after Roux-en-Y gastric bypass surgery	

For a patient, if the sum of the scores over all clinical variables is < 7 , we will classify this patient as one having the diabetes remission, otherwise, we will put him in the class of non-remission.

4. Compare the predictive power of the considered models (decision trees, random forest, and the DiaRem). What can you conclude?

References:

1. "The use of classification trees for bioinformatics"
<http://moult.ibbr.umd.edu/JournalClubPresentations/Maya/Maya-04Feb2011-paper.pdf>
2. "Preoperative prediction of type 2 diabetes remission after Roux-en-Y gastric bypass surgery: a retrospective cohort study" <https://www.ncbi.nlm.nih.gov/pubmed/24579062>