Intel® Extension for Scikit-learn Random Forest for Yolanda dataset

```
! python -m pip install --upgrade pip
! python -m pip install scikit-learn-intelex
```

```
Requirement already satisfied: pip in /usr/local/lib/python3.10/dist-packages (23.1.2)
Collecting pip
  Downloading pip-23.2.1-py3-none-any.whl (2.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.1/2.1 MB 23.5 MB/s eta 0:00:00
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 23.1.2
    Uninstalling pip-23.1.2:
      Successfully uninstalled pip-23.1.2
Successfully installed pip-23.2.1
Collecting scikit-learn-intelex
  Obtaining dependency information for scikit-learn-intelex from https://files.pythonhosted.org/packages/3d/89/36bf3b024b86ae3c1de86fcaa11ef6e6dfc4d2
  Downloading scikit_learn_intelex-2023.2.1-py310-none-manylinux1_x86_64.whl.metadata (12 kB)
Collecting daal4py==2023.2.1 (from scikit-learn-intelex)
  Obtaining dependency information for daal4py==2023.2.1 from https://files.pythonhosted.org/packages/18/7a/a3ecef294e068a38d8e8d370e84127afca3f95d72
  Downloading daal4py-2023.2.1-py310-none-manylinux1_x86_64.whl.metadata (7.4 kB)
Requirement already satisfied: scikit-learn>=0.22 in /usr/local/lib/python3.10/dist-packages (from scikit-learn-intelex) (1.2.2)
Collecting daal==2023.2.1 (from daal4py==2023.2.1->scikit-learn-intelex)
  Obtaining dependency information for daal==2023.2.1 from https://files.pythonhosted.org/packages/e0/51/9369c49b50a51279660c31b0ec22afcb649f082fd275
  Downloading daal-2023.2.1-py2.py3-none-manylinux1_x86_64.whl.metadata (1.1 kB)
Requirement already satisfied: numpy>=1.19 in /usr/local/lib/python3.10/dist-packages (from daal4py==2023.2.1->scikit-learn-intelex) (1.23.5)
Requirement already satisfied: tbb==2021.* in /usr/local/lib/python3.10/dist-packages (from daal==2023.2.1->daal4py==2023.2.1->scikit-learn-intelex)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (from scikit-learn>=0.22->scikit-learn-intelex) (1.11.2)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn>=0.22->scikit-learn-intelex) (1.3.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn>=0.22->scikit-learn-intelex) (3.2.0
Downloading scikit_learn_intelex-2023.2.1-py310-none-manylinux1_x86_64.whl (128 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 128.7/128.7 kB 6.9 MB/s eta 0:00:00
Downloading daal4py-2023.2.1-py310-none-manylinux1_x86_64.whl (14.0 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 14.0/14.0 MB 79.5 MB/s eta 0:00:00
Downloading daal-2023.2.1-py2.py3-none-manylinux1_x86_64.whl (75.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 75.3/75.3 MB 9.9 MB/s eta 0:00:00
Installing collected packages: daal, daal4py, scikit-learn-intelex
Successfully installed daal-2023.2.1 daal4py-2023.2.1 scikit-learn-intelex-2023.2.1
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended
```

Resolution for Known Issue on Google Colab

```
import sys
import os
import site

sys.path.append(os.path.join(os.path.dirname(site.getsitepackages()[0]), "site-packages"))
```

```
print(os.path.join(os.path.dirname(site.getsitepackages()[0]), "site-packages"))
```

```
/usr/local/lib/python3.10/site-packages
```

```
print(os.path.dirname(site.getsitepackages()[0]))
```

```
/usr/local/lib/python3.10
```

```
print(sys.path.append(os.path.join(os.path.dirname(site.getsitepackages()[0]), "site-packages")))
```

```
None
```

```
from timeit import default_timer as timer
from sklearn import metrics
from IPython.display import HTML
from sklearn.datasets import fetch_openml
from sklearn.model_selection import train_test_split
```

```
!python --version
```

```
Python 3.10.12
```

Download the data

```
x, y = fetch_openml(name="Yolanda", return_X_y=True)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/datasets/_openml.py:968: FutureWarning: The default value of `parser` will change from `'liac-arff'`
  warn(
```

Split the data into train and test sets

```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=72)
x_train.shape, x_test.shape, y_train.shape, y_test.shape
```

```
((280000, 100), (120000, 100), (280000,), (120000,))
```

Patch original Scikit-learn with Intel® Extension for Scikit-learn

Intel® Extension for Scikit-learn (previously known as daal4py) contains drop-in replacement functionality for the stock Scikit-learn package. You can take advantage of the performance optimizations of Intel® Extension for Scikit-learn by adding just two lines of code before the usual Scikit-learn imports:

```python
from sklearnex import patch_sklearn

patch_sklearn()
```

```
Intel(R) Extension for Scikit-learn* enabled (https://github.com/intel/scikit-learn-intelex)
```

Intel® Extension for Scikit-learn patching affects performance of specific Scikit-learn functionality. Refer to the list of supported algorithms and parameters for details. In cases when unsupported parameters are used, the package fallbacks into original Scikit-learn. If the patching does not cover your scenarios, submit an issue on GitHub.

Training Random Forest algorithm with Intel® Extension for Scikit-learn for Yolanda dataset

```python
from sklearn.ensemble import RandomForestRegressor

params = {"n_estimators": 150, "random_state": 44, "n_jobs": -1}
start = timer()
rf = RandomForestRegressor(**params).fit(x_train, y_train)
train_patched = timer() - start
f"Intel® extension for Scikit-learn time: {train_patched:.2f} s"
```

```
'Intel® extension for Scikit-learn time: 2081.74 s'
```

Predict and get a result of the Random Forest algorithm with Intel® Extension for Scikit-learn

```python
y_pred = rf.predict(x_test)
mse_opt = metrics.mean_squared_error(y_test, y_pred)
f"Intel® extension for Scikit-learn Mean Squared Error: {mse_opt}"
```

```
'Intel® extension for Scikit-learn Mean Squared Error: 83.60018104634611'
```

Train the same algorithm with original Scikit-learn

In order to cancel optimizations, we use *unpatch_sklearn* and reimport the class RandomForestRegressor.

```
from sklearnex import unpatch_sklearn

unpatch_sklearn()
```

Training Random Forest algorithm with original Scikit-learn library for Yolanda dataset

```
from sklearn.ensemble import RandomForestRegressor

start = timer()
rf = RandomForestRegressor(**params).fit(x_train, y_train)
train_unpatched = timer() - start
f"Original Scikit-learn time: {train_unpatched:.2f} s"
```

```
    'Original Scikit-learn time: 4453.85 s'
```

Predict and get a result of the Random Forest algorithm with original Scikit-learn

```
y_pred = rf.predict(x_test)
mse_original = metrics.mean_squared_error(y_test, y_pred)
f"Original Scikit-learn Mean Squared Error: {mse_opt}"
```

```
    'Original Scikit-learn Mean Squared Error: 83.60018104634611'
```

```
HTML(
    f"<h3>Compare MSE metric of patched Scikit-learn and original</h3>"
    f"MSE metric of patched Scikit-learn: {mse_opt} <br>"
    f"MSE metric of unpatched Scikit-learn: {mse_original} <br>"
    f"Metrics ratio: {mse_opt/mse_original} <br>"
    f"<h3>With Scikit-learn-intelex patching you can:</h3>"
    f"<ul>"
    f"<li>Use your Scikit-learn code for training and prediction with minimal changes (a couple of lines of code);</li>"
    f"<li>Fast execution training and prediction of Scikit-learn models;</li>"
    f"<li>Get the similar quality</li>"
    f"<li>Get speedup in <strong>{(train_unpatched/train_patched):.1f}</strong> times.</li>"
    f"</ul>"
)
```

Compare MSE metric of patched Scikit-learn and original

MSE metric of patched Scikit-learn: 83.60018104634611
MSE metric of unpatched Scikit-learn: 83.80131297814816
Metrics ratio: 0.9975998952205618

With Scikit-learn-intelex patching you can:

- Use your Scikit-learn code for training and prediction with minimal changes (a couple of lines of code);
- Fast execution training and prediction of Scikit-learn models;
- Get the similar quality
- Get speedup in **2.1** times.