4_MachineLearningExample

October 23, 2019

1 Machine learning example

Andrea Volkamer

1.1 Imports

[1]:

```
import pandas as pd
import numpy as np
```

1.2 Data collection and preparation

1.2.1 ChEMBL

- Largest freely available source for molecules and affinities
- https://www.ebi.ac.uk/chembl/

1.2.2 IC50 values

- Measure of the effectiveness of a substance in inhibiting a specific biological or biochemical function
- pIC50 = - log10(IC50)

[2]:

```
df_act = pd.read_csv('./data/Chembl_EGFR_actives_ML.csv', delimiter=',',␣
 ↪header=0)
df_inact = pd.read_csv('./data/Chembl_EGFR_inactives_ML.csv', delimiter=',',␣
 ↪header=0)
```

[3]:

```
print (len(df_act), len(df_inact))
```

```
100 100
```

[4]:

```
df_act.head()
```

[4]:

```
                                    canonical_smiles molecule_chembl_id
0             COc1ccc(NC(=O)c2ccc(cc2)N(CCCl)CCCl)cc1      CHEMBL589588
1          N(c1ccc2[nH]ccc2c1)c3ncnc4cc(sc34)c5ccccc5       CHEMBL76432
2  CS(=O)(=O)CCNCCCCOc1ccc2ncnc(Nc3ccc(F)c(Cl)c3)...      CHEMBL460731
```

```
3  NC(=O)C1CCN(Cc2ccc(cc2)c3cc4nccc(Nc5ccc6[nH]cc...      CHEMBL431977
4  CC(C)(CO)NCc1ccc(cc1)c2cc3ncnc(Nc4ccc5[nH]ccc5...      CHEMBL308498
```

Get smiles in array

[5]:

```
act_smiles = df_act['canonical_smiles'].tolist()
inact_smiles = df_inact['canonical_smiles'].tolist()
```

Calculate fingerprints

[6]:

```
from rdkit import Chem
from rdkit.Chem import rdFingerprintGenerator

# Fingerprints for active molecules
mols_act = [Chem.MolFromSmiles(x) for x in act_smiles]
# By default the RDKit generates Morgan fingerprints with radius 2 (MFP2)
fps_act = rdFingerprintGenerator.GetFPs(mols_act)

# Fingerprints for inactive molecules
mols_inact = [Chem.MolFromSmiles(x) for x in inact_smiles]
fps_inact = rdFingerprintGenerator.GetFPs(mols_inact)

# Concatenate fingerprints
fps = fps_act + fps_inact
```

Prepare class assignment

[7]:

```
# 'Active' = 1
y_act = np.ones(len(fps_act))

# 'Inactive' = 0
y_inact = np.zeros(len(fps_inact))

# Classifier
y = np.concatenate([y_act, y_inact])
```

1.3 Random forest

- Supervised classification algorithm, ensemble learning method
- Operates by constructing a multitude of decision trees at training time
- Data is normally split into train and test set
- Performance evaluation

1.3.1 Split data in train and test set

[8]:
```
from sklearn.model_selection import train_test_split

# 20% for testing, 80% for training
X_train, X_test, y_train, y_test = train_test_split(fps, y, test_size=0.20)
```

1.3.2 Train the model

See http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html for an explanation of the parameter.

[9]:
```
from sklearn.ensemble import RandomForestClassifier

forest = RandomForestClassifier(n_jobs=-1, n_estimators=100)
forest.fit(X_train, y_train) # Build a forest of trees from the training set
```

[9]:
```
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=100, n_jobs=-1,
            oob_score=False, random_state=None, verbose=0,
            warm_start=False)
```

1.3.3 Test performance of the model

[10]:
```
from sklearn import metrics
y_pred = forest.predict(X_test) # Predict class for X
accuracy = metrics.accuracy_score(y_test, y_pred)
print("Accuracy: %.2f" %accuracy)
print("Confusion matrix:")
print(metrics.confusion_matrix(y_test,y_pred))
```

```
Accuracy: 0.90
Confusion matrix:
[[17  2]
 [ 2 19]]
```

1.4 Random forest predictions for FDA approved drugs

[11]:
```
df = pd.read_csv('./data/EGFR-course.csv', delimiter=',', names=['Smiles',␣
 ↪'Name'], header=None)
df.head()
```

[11]:
```
                                       Smiles        Name
0     COc1cc2ncnc(Nc3ccc(F)c(Cl)c3)c2cc1OCCCN1CCOCC1    Gefitinib
1         C#Cc1cccc(Nc2ncnc3cc(OCCOC)c(OCCOC)cc23)c1    Erlotinib
```

```
2  CS(=O)(=O)CCNCc1ccc(-c2ccc3ncnc(Nc4ccc(OCc5ccc...    Lapatinib
3  CN(C)C/C=C/C(=O)Nc1cc2c(Nc3ccc(F)c(Cl)c3)ncnc2...     Afatinib
4  C=CC(=O)Nc1cc(Nc2nccc(-c3cn(C)c4ccccc34)n2)c(O...  Osimertinib
```

[12]:

```python
for tmp_smiles in df.Smiles.values:
    mol = Chem.MolFromSmiles(tmp_smiles)
    fps = rdFingerprintGenerator.GetFPs([mol])

    y_pred = forest.predict(fps)
    y_prob = forest.predict_proba(fps)
    print(y_pred, y_prob)
```

```
[1.] [[0. 1.]]
[1.] [[0.04 0.96]]
[1.] [[0.12 0.88]]
[1.] [[0.05 0.95]]
[0.] [[0.61 0.39]]
```

[ ]:

```

```