ECON 425 HW8 Solutions

February 29, 2024

Due Thu, Mar 7, 6pm in Bruinlearn

Problem 1 Use the dataset 'marketing-campaign.csv' uploaded on Canvas (description can be found here: https://archive.ics.uci.edu/dataset/222/bank+marketing). Convert categorical variables of you choice (you need not use all of them) into dummies and allocate a third of your data to the testing sample.

(i) First, suppose each tree in a random forest picks a random subset of $m = \sqrt{p}$ features at each split, where p is the number of features in the data. On the training sample, fit the random forest to predict subscription to a term deposit. Vary the number of trees in the range $\{1, 2, 3, 4, 5, 10, 20, 50\}$. Plot accuracy, precision, recall, and F1 score on the testing sample against the number of trees.

(ii) Repeat (i) with *bagging*, i.e. $m = p$.

(iii) Pick the best-performing model and use the feature_importances_ attribute of RandomForestClassifier to evaluate importance of different features. Is there a clearly dominating feature? Explain.

(iv) Beyond sampling variation, is there any other explanation for the alternating pattern in some performance metrics arising when the number of trees is very small?

[3]:
```
# SOLUTION

import numpy as np
import pandas as pd
from matplotlib import pyplot as plt

from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score,␣
 ↪f1_score

data = pd.read_csv("/Users/franguri/Library/CloudStorage/Dropbox/_TEACHING_/
 ↪Econ 425 ML UCLA Winter 2024/WEEK 8 Bagging and boosting/marketing_campaign.
 ↪csv",
                        header = 0, sep = ";")
```

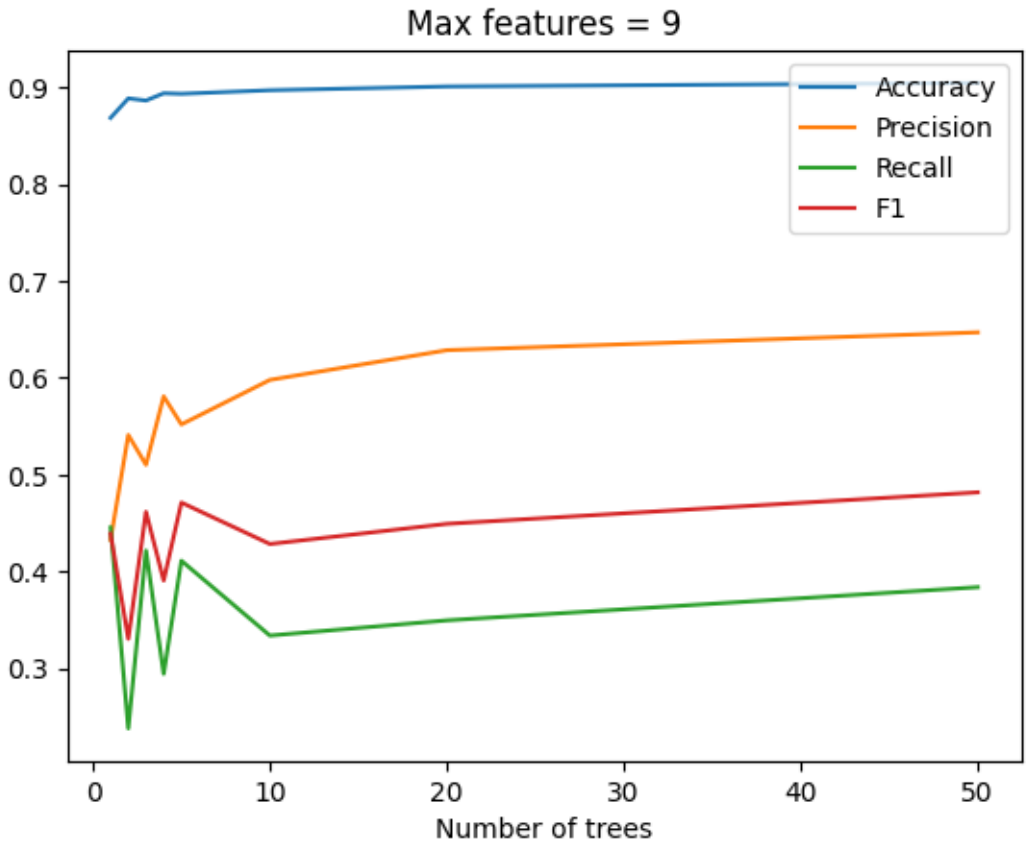
```
categorical_vars =␣
 ↪["job","marital","education","contact","day","month","housing","default","loan","poutcome"]
data_encoded = pd.get_dummies(data, columns=categorical_vars)

X = data_encoded.drop(columns='y')
y = data_encoded['y']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33,␣
 ↪random_state=1)

max_feat_list = [int(np.sqrt(X.shape[1])), X.shape[1]]
n_trees_list = [1, 2, 3, 4, 5, 10, 20, 50]

for max_features in max_feat_list:
    print('=== m =', max_features, '===')
    accuracy, precision, recall, f1 = [], [], [], []
    for n_trees in n_trees_list:
        print('Number of trees =', n_trees)
        rf = RandomForestClassifier(max_features=max_features,␣
 ↪n_estimators=n_trees, random_state=1).fit(X_train, y_train)
        y_pred = rf.predict(X_test)
        accuracy.append(accuracy_score(y_test, y_pred))
        precision.append(precision_score(y_test, y_pred, pos_label='yes'))
        recall.append(recall_score(y_test, y_pred, pos_label='yes'))
        f1.append(f1_score(y_test, y_pred, pos_label='yes'))

        feature_imp = pd.DataFrame({'importance':rf.feature_importances_},␣
 ↪index=list(X.columns))
        print(feature_imp.sort_values(by='importance', ascending=False))

    plt.figure()
    plt.plot(n_trees_list, accuracy)
    plt.plot(n_trees_list, precision)
    plt.plot(n_trees_list, recall)
    plt.plot(n_trees_list, f1)
    plt.xlabel('Number of trees')
    plt.legend(['Accuracy', 'Precision', 'Recall', 'F1'])
    plt.title('Max features = ' + str(max_features))
    plt.show()
```

```
=== m = 9 ===
Number of trees = 1
                  importance
duration            0.268994
age                 0.087055
balance             0.077133
poutcome_success    0.039625
```

```
campaign             0.038658
…                       …
day_31               0.001882
day_28               0.001297
job_unknown          0.001155
contact_telephone    0.001127
default_no           0.000766

[81 rows x 1 columns]
Number of trees = 2
               importance
duration         0.259693
age              0.080665
balance          0.077659
campaign         0.036889
pdays            0.030131
…                   …
job_housemaid    0.002260
day_31           0.002007
default_no       0.001565
default_yes      0.001436
job_unknown      0.001082

[81 rows x 1 columns]
Number of trees = 3
                  importance
duration            0.253663
balance             0.076252
age                 0.076251
poutcome_success    0.043240
campaign            0.037085
…                      …
day_24              0.001978
day_31              0.001923
job_unknown         0.001409
default_no          0.001213
default_yes         0.001122

[81 rows x 1 columns]
Number of trees = 4
                  importance
duration            0.247627
balance             0.079325
age                 0.075700
poutcome_success    0.056526
campaign            0.037471
…                      …
day_31              0.002192
```

```
day_24              0.002050
job_unknown         0.001555
default_no          0.001243
default_yes         0.001070

[81 rows x 1 columns]
Number of trees = 5
                  importance
duration            0.247184
age                 0.081561
balance             0.080012
poutcome_success    0.054146
campaign            0.037467
…                      …
day_24              0.002124
day_31              0.002092
default_no          0.001316
job_unknown         0.001296
default_yes         0.001096

[81 rows x 1 columns]
Number of trees = 10
                  importance
duration            0.253563
age                 0.084005
balance             0.080293
poutcome_success    0.047577
pdays               0.036256
…                      …
day_26              0.002565
day_31              0.002035
default_yes         0.001186
job_unknown         0.001149
default_no          0.001033

[81 rows x 1 columns]
Number of trees = 20
                  importance
duration            0.249500
balance             0.083220
age                 0.081948
poutcome_success    0.050968
campaign            0.034246
…                      …
day_24              0.002555
day_31              0.001714
job_unknown         0.001183
default_no          0.001082
```

```
default_yes         0.001024

[81 rows x 1 columns]
Number of trees = 50
                  importance
duration            0.250204
balance             0.082622
age                 0.082470
poutcome_success    0.051929
campaign            0.034792
…                        …
day_24              0.002555
day_31              0.001461
default_yes         0.001200
job_unknown         0.001140
default_no          0.001008

[81 rows x 1 columns]
```

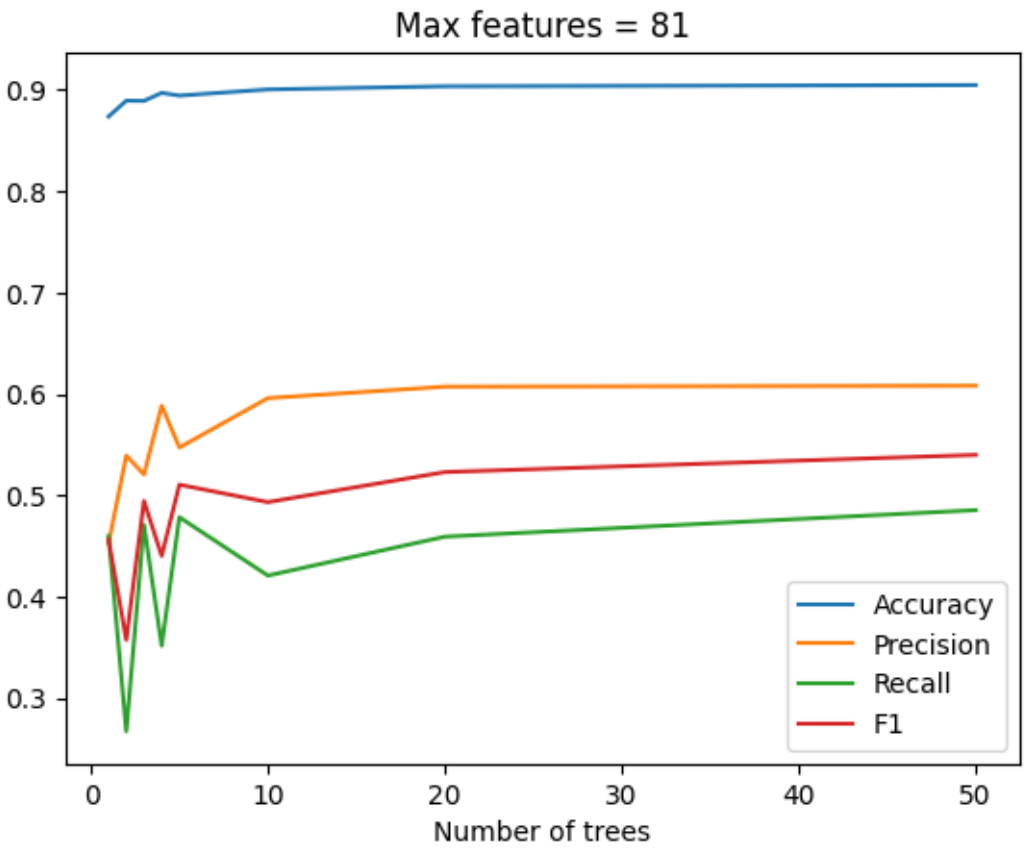



```
=== m = 81 ===
```

```
Number of trees = 1
                  importance
duration            0.288008
balance             0.088926
age                 0.087021
poutcome_success    0.077495
pdays               0.035064
…                      …
loan_yes            0.001612
job_entrepreneur    0.001557
job_unknown         0.001383
default_yes         0.001009
default_no          0.000566

[81 rows x 1 columns]
Number of trees = 2
                  importance
duration            0.280395
balance             0.089213
age                 0.086719
poutcome_success    0.084087
pdays               0.036476
…                      …
day_31              0.001338
job_entrepreneur    0.001180
job_unknown         0.001161
default_no          0.000769
default_yes         0.000534

[81 rows x 1 columns]
Number of trees = 3
                  importance
duration            0.279889
age                 0.089575
balance             0.086141
poutcome_success    0.086141
pdays               0.035379
…                      …
job_entrepreneur    0.001716
day_31              0.001593
job_unknown         0.001092
default_no          0.000737
default_yes         0.000624

[81 rows x 1 columns]
Number of trees = 4
                  importance
duration            0.278035
```

```
age                 0.088249
poutcome_success    0.087177
balance             0.086539
pdays               0.035651
…                      …
month_jan           0.002042
day_31              0.001675
job_unknown         0.000974
default_no          0.000802
default_yes         0.000649

[81 rows x 1 columns]
Number of trees = 5
                  importance
duration            0.276883
age                 0.089233
balance             0.088395
poutcome_success    0.086491
pdays               0.035737
…                      …
poutcome_unknown    0.002248
day_31              0.001530
job_unknown         0.001057
default_no          0.000807
default_yes         0.000614

[81 rows x 1 columns]
Number of trees = 10
                  importance
duration            0.273727
age                 0.088633
balance             0.087414
poutcome_success    0.087311
pdays               0.036163
…                      …
poutcome_unknown    0.001975
day_31              0.001195
job_unknown         0.001093
default_no          0.000783
default_yes         0.000529

[81 rows x 1 columns]
Number of trees = 20
                  importance
duration            0.271292
poutcome_success    0.086972
balance             0.086448
age                 0.086138
```

```
pdays                0.036925
…                       …
poutcome_unknown     0.002095
job_unknown          0.001371
day_31               0.001055
default_no           0.000715
default_yes          0.000612

[81 rows x 1 columns]
Number of trees = 50
                   importance
duration             0.269303
poutcome_success     0.087456
balance              0.085948
age                  0.084910
pdays                0.036819
…                       …
poutcome_unknown     0.002235
job_unknown          0.001187
day_31               0.001095
default_yes          0.000693
default_no           0.000656

[81 rows x 1 columns]
```


Max features = 81
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0
10
20
30
40
50
Number of trees
Accuracy
Precision
Recall
F1