

## svm

January 30, 2019

```
In [1]: import numpy
        from sklearn.utils import shuffle
        from sklearn.model_selection import train_test_split
        X = numpy.loadtxt("./data/Train/X_train.txt")
        y = numpy.loadtxt("./data/Train/y_train.txt")

In [5]: X, y = shuffle(X, y)
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20)

In [5]: from models import svm
        from sklearn.metrics import accuracy_score
        ls = [5, 10, 50, 100, 561]
        report = {}

In [6]: from sklearn.feature_selection import SelectKBest, mutual_info_classif
        report['mutual_info_classif'] = []
        for l in ls:
            transformer = SelectKBest(mutual_info_classif, k=l)
            X_train_new = transformer.fit_transform(X_train, y_train)
            svm_model = svm.without_penalty(X_train_new, y_train)
            y_pred = svm_model.predict(transformer.transform(X_test))
            score = accuracy_score(y_test, y_pred)
            data={
                'l': l,
                'score': score
            }
            report['mutual_info_classif'].append(data)

In [9]: report['mutual_info_classif']

Out[9]: [{'l': 5, 'score': 0.5495495495495496},
          {'l': 10, 'score': 0.5669240669240669},
          {'l': 50, 'score': 0.7213642213642214},
          {'l': 100, 'score': 0.9202059202059202},
          {'l': 561, 'score': 0.971042471042471}]

In [10]: from sklearn.feature_selection import SelectFromModel
         clf = svm.penalty_l1(X_train, y_train, 0.01)
```

```

report['sfm_svml1_0.01'] = []
for l in ls:
    transformer = SelectFromModel(clf, prefit=True, max_features=l)
    X_train_new = transformer.transform(X_train)
    svm_model = svm.without_penalty(X_train_new, y_train)
    y_pred = svm_model.predict(transformer.transform(X_test))
    score = accuracy_score(y_test, y_pred)
    data={
        'l': l,
        'score': score
    }
    report['sfm_svml1_0.01'].append(data)

```

/home/mahdi/.local/share/virtualenvs/machine-learning-final-project-uk2p9d2v/lib/python3.6/site-  
 "the number of iterations.", ConvergenceWarning)

In [11]: report['sfm\_svml1\_0.01']

```

Out[11]: [{'l': 5, 'score': 0.7767052767052767},
          {'l': 10, 'score': 0.8931788931788932},
          {'l': 50, 'score': 0.9562419562419563},
          {'l': 100, 'score': 0.9581724581724582},
          {'l': 561, 'score': 0.9607464607464608}]

```

```

In [12]: from sklearn.feature_selection import SelectFromModel
clf = svm.penalty_l2(X_train, y_train, 10)
report['sfm_svml2_10'] = []
for l in ls:
    transformer = SelectFromModel(clf, prefit=True, max_features=l)
    X_train_new = transformer.transform(X_train)
    svm_model = svm.without_penalty(X_train_new, y_train)
    y_pred = svm_model.predict(transformer.transform(X_test))
    score = accuracy_score(y_test, y_pred)
    data={
        'l': l,
        'score': score
    }
    report['sfm_svml2_10'].append(data)

```

In [13]: report['sfm\_svml2\_10']

```

Out[13]: [{'l': 5, 'score': 0.646074646074646},
          {'l': 10, 'score': 0.8037323037323038},
          {'l': 50, 'score': 0.9543114543114544},
          {'l': 100, 'score': 0.9697554697554698},
          {'l': 561, 'score': 0.9716859716859717}]

```