custom_assessment_function_with_search_grid

February 17, 2019

In [1]:
```
from sklearn.metrics import log_loss, make_scorer
Log_loss = make_scorer(log_loss, greater_is_better=False, needs_proba=True) # logarith
```

In [2]:
```
from sklearn.datasets import load_digits
digits = load_digits()
X, Y = digits.data, digits.target

from sklearn import svm
hp = svm.SVC(probability=True, random_state=1)

from sklearn.model_selection import GridSearchCV
search_grid = [
    {'C': [1, 10, 100, 1000], 'kernel':['linear']},
    {'C': [1, 10, 100, 1000], 'gamma': [0.001, 0.0001], 'kernel':['rbf'] },
]
```

In [3]:
```
search_func = GridSearchCV(estimator=hp, param_grid=search_grid, scoring=Log_loss, n_j
search_func.fit(X, Y)
print(search_func.best_score_)
print(search_func.best_params_)
```

```
-0.16138394081976534
{'C': 1, 'gamma': 0.001, 'kernel': 'rbf'}
```

In [4]:
```
# new lose function
import numpy as np
from sklearn.preprocessing import LabelBinarizer
def custom_log_loss_func(ground_truth, p_predictions, penalty=list(), eps=1e-15):
    adj_p= np.clip(p_predictions, eps, 1-eps)
    lb = LabelBinarizer()
    g = lb.fit_transform(ground_truth)
    if g.shape[1] == 1:
        g.append(1-g, g, axis=1)
    if penalty:
        g[:,penalty] = g[:,penalty] * 2
    summation = np.sum(g * np.log(adj_p))
    return summation * (-1.0/len(ground_truth))
```

In [5]:
```
# scorrer thats add penalty for 4 and 9 becouse thoes numbers are easy to confused wit
my_custom_scorer = make_scorer(custom_log_loss_func, greater_is_better=False, needs_pr
```

In [7]:
```
search_grid = [
    {'C': [1, 10, 100, 1000], 'kernel':['linear']},
    {'C': [1, 10, 100, 1000], 'gamma': [0.001, 0.0001], 'kernel':['rbf'] },
]
search_func = GridSearchCV(estimator=hp, param_grid=search_grid,
                           scoring=my_custom_scorer,
                           n_jobs=1, iid=False,
                           refit=True, cv=3)
search_func.fit(X, Y)
print(search_func.best_score_)
print(search_func.best_params_)
```

```
-0.19961027129825612
{'C': 1, 'gamma': 0.001, 'kernel': 'rbf'}
```

In [ ]: