

# Cross evaluation for algorithms

```
def
    evaluate(train_data
    ,kmax,algo):
    test_scores = {}
    train_scores = {}
    for i in
        range(2,kmax
        ,2): kf =
            KFold(n_spli
            ts = i)
            sum_train =
            0
            s
            u
            m
            _
            t
            e
            s
            t
            =
            0
            d
            a
            t
            a
            =
            d
            f
            for train,test in
                kf.split(data):
                    train_data =
                    data.iloc[train,:]
                    test_data =
                    data.iloc[test,:]
                    x_train =
                    train_data.drop(["prog"],axis=1)
                    y_train = train_data['prognosis']
                    x_test =
                    test_data.drop(["prog"],axis=
                    1)y_test =
                    test_data["prognosis"]
                    algo_model =
                    algo.fit(x_train,y_train)
                    sum_train +=
                    algo_model.(x_train,y_train)
```

```

        y_pred =
        algo_model.predict(x_test)
        sum_test +=
        accuracy_score(y_test,y_pred)
    average_test = sum_test/i
    average_train =
    sum_train/i
    test_scores[i] =
    average_test
    train_scores[i] =
    average_train
    print("kvalue: ",i)
return(train_scores,test_scores)

```

- Finding test and train score of algorithm:

```

max_kfold = 11
for algo_name in
    algo_dict.keys():
    print(algo_name)
    trscore,tstscore=evaluate(dict[
        algo_name])
    algo_train_scores[algo_name] = tr_score
    algo_test_scores[algo_name] = tst_score

print(algo_train_scores)
print(algo_test_scores)

```