neural_network_size_training_data

January 30, 2019

```python
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")
```

```python
In [3]: X, y = shuffle(X, y)
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.10)
        size_training = len(X_train)
```

```python
In [4]: from models import neural_network
        from sklearn.metrics import accuracy_score
        ks = [.05, .10, .20, .50, 1]
        report = []
        for k in ks:
            neural_network_model = neural_network.without_penalty(X_train[:int(k*size_training)]
            y_pred = neural_network_model.predict(X_test)
            score = accuracy_score(y_test, y_pred)
            data={
                'k': k,
                'score': score
            }
            report.append(data)
```

```python
In [5]: report
```

```python
Out[5]: [{'k': 0.05, 'score': 0.8996138996138996},
         {'k': 0.1, 'score': 0.9395109395109396},
         {'k': 0.2, 'score': 0.9588159588159588},
         {'k': 0.5, 'score': 0.9781209781209781},
         {'k': 1, 'score': 0.9832689832689833}]
```