

NeuralNetsGeneticAlgorithms

March 29, 2019

1 Genetic Algorithm evolving Neural Net

This notebook presents the performance of genetic algorithm that aims in evolving neural net's weights.

This neural net is trained to recognize the pictures containing plus symbol.

1.1 Dataset

Firstly, we need to prepare dataset. It consists of 2000 pictures, half of which are target cases, and second half is noise.

```
In [17]: from GANN import *

n_cases = 1000
height = 4
width = 4
placeholders_target = np.zeros((n_cases,height,width))
placeholders_noise = np.zeros((n_cases,height,width))

placeholders_target = np.array([create_instance(placeholders_target[n])
                                for n in range(n_cases)])
placeholders_noise = np.array([create_random_inst(placeholders_noise[n])
                                for n in range(n_cases)])

target = [[1,0] for n in range(n_cases)]
noise = [[0,1] for n in range(n_cases)]

dataset = np.vstack([placeholders_target,placeholders_noise])
target = np.vstack([target,noise])
dataset = dataset.reshape((n_cases*2,height*width))
```

1.2 Setting parameters

I am checking which initial number of population provides best results.

```
In [6]: results = dict()
        for n in range(4,11):
```

```

n = n * 100
ga = GenAlWeightsNN(n_pop=n,max_gen=50)
ga.fit(dataset,target)
ga.transform()
results[n] = (ga.best_ind,ga.validate())
#     ga.plot_fitness()

result_dict = pd.DataFrame(index = results.keys())
result_dict['Accuracy on test'] = [round(n[0]*100,2) for n in results.values()]
result_dict['Accuracy validate'] = [round(n[1],2) for n in results.values()]
result_dict

```

```

Out[6]:

```

	Accuracy on test	Accuracy validate
400	79.41	78.23
500	77.64	77.30
600	79.13	79.04
700	80.22	78.69
800	80.12	78.90
900	78.99	77.69
1000	80.17	79.49

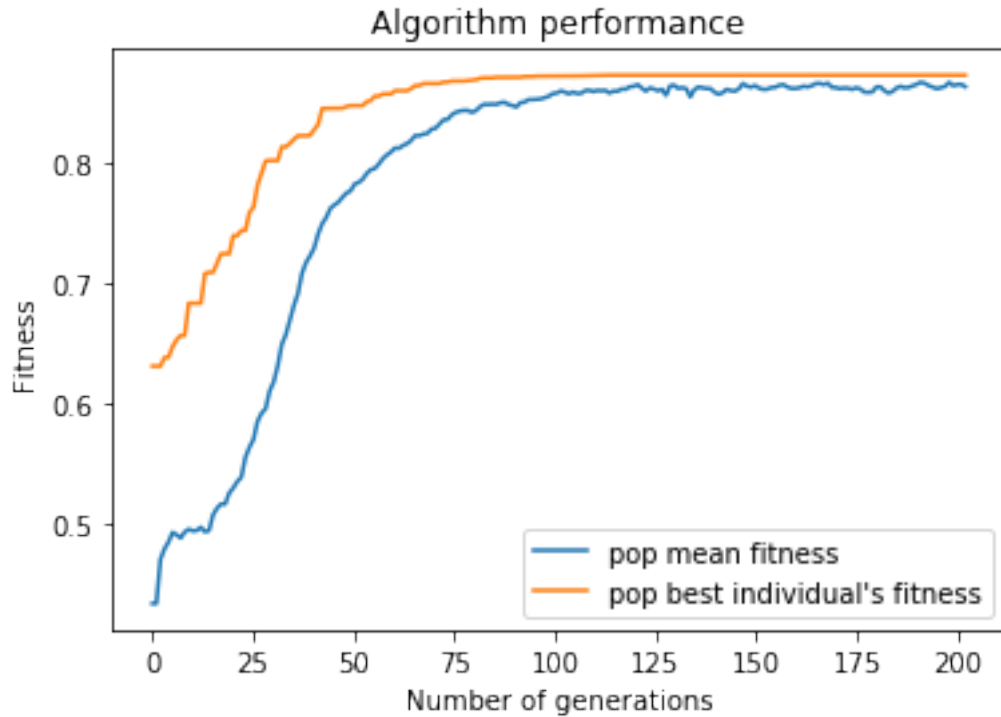
2 The performance of algorithm

After testing some paramters I have found the fine parameters that allows for achieving best results. In purpose of maximazing the search space I used 1000 initial population.

```

In [9]: ga = GenAlWeightsNN(n_pop=1000,max_gen=200,desired_fit=0.95)
ga.fit(dataset,target)
ga.transform()
ga.plot_fitness()
print('Achieved accuracy:{}'.format(round(ga.best_ind*100,2)))
print('Achieved accuracy on validate dataset:{}'.format(
    round(ga.validate(),2)))

```



Achieved accuracy:87.26

Achieved accuracy on validate dataset:87.3

2.1 Summary

Algorithm performed correctly, optimizing weights in such way that now neural network is classifying with 87% accuracy on both training and test datasets.

2.1.1 TO DO

Next thing to do is to adapt the algorithm so it evolves the architecture of neural nets.

2.1.2 References

Above implementation used a parts of numpy-based neural net taken from:

<https://www.kaggle.com/niyipop/2-layer-neural-network-from-scratch-using-numpy/data/>

Genetic algorithm originates from:

https://github.com/stakar/Sig_Feat_Selector