Homework #1 Alina Chasova

Task #1

Model Architecture:

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Linear-1             [-1, 150, 100]             500
              ReLU-2             [-1, 150, 100]               0
            Linear-3             [-1, 150, 100]          10,100
              ReLU-4             [-1, 150, 100]               0
            Linear-5               [-1, 150, 3]             303
           Softmax-6               [-1, 150, 3]               0
================================================================
Total params: 10,903
Trainable params: 10,903
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.00
Forward/backward pass size (MB): 0.46
Params size (MB): 0.04
Estimated Total Size (MB): 0.51
----------------------------------------------------------------
```

Training information of the model:

Loss - Categorical Cross Entropy
Epochs - 1000
Batch Size - 32
Metric - Accuracy
Anneal every - 1000 iterations
Min Temperature - 1e-8

Results with different Anneal Rates:

Anneal Rate	Resulted accuracy	Resulted loss	Speed
0.1	.86	.68	13m 47s
0.01	.66	.88	10m 13s
0.001	.85	.69	10m 4s
0.0001	.79	.76	9m 59s

Result of the model with SGD Optimizer:

Resulted accuracy	Resulted loss	Speed
.98	.58	5m 12s

Task #2

Unchangeable parameters of algorithm:

Initial Temperature - 10000
Aneal every - 1000
Min Temperature - 1e-8

Results with different Anneal Rates (visualization can be found in the github):

Anneal Rate	Resulted kms	Speed
0.1	19943.47141004659	2m 24s
0.01	22924.596106629215	1m 12s
0.001	22869.602753477702	1m 5s
0.0001	40861.06313776539	34s

References:

Github:
https://github.com/LinaChasova/Simulated-Annealing
Used materials:
http://www.theprojectspot.com/tutorial-post/simulated-annealing-algorithm-for-beginners/6
https://python-scripts.com/animations-with-matplotlib
https://matplotlib.org/api/_as_gen/matplotlib.animation.FuncAnimation.html
https://pandas.pydata.org/docs/user_guide/index.html
https://pynative.com/python-random-shuffle/
https://en.wikipedia.org/wiki/Haversine_formula
https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html
https://docs.scipy.org/doc/numpy/reference/generated/numpy.argmax.html
https://github.com/jramapuram/SimulatedAnnealing/blob/master/optim/sa.py
https://github.com/yangzhangalmo/pytorch-iris/blob/master/main.py
https://discuss.pytorch.org/