Deep learning inference of the neutron star equation of state

Giulia Ventagli[1] and **Ippocratis D. Saltas**[1]

[1]CEICO, Institute of Physics of the Czech Academy of Sciences, Na Slovance 2, 182 21 Praha 8, Czechia

Supplementary material

Classification NN						
n	N	Structure	batch size	epochs	binary accuracy	val. binary acc.
100	15	(30)→ 35, relu → 2, sigmoid	50	100	0.754 ± 0.015	0.753 ± 0.016
100	15	(30)→ 70, relu → 2, sigmoid	50	100	0.802 ± 0.008	0.802 ± 0.009
100	15	(30)→ 80, relu → 70, relu → 2, sigmoid	50	100	0.817 ± 0.002	0.809 ± 0.009
200	15	(30)→ 80, relu → 70, relu → 2, sigmoid	100	100	0.823 ± 0.002	0.818 ± 0.008
100	20	(40)→ 35, relu → 2, sigmoid	50	100	0.759 ± 0.013	0.753 ± 0.015
100	20	(40)→ 70, relu → 2, sigmoid	50	100	0.804 ± 0.007	0.798 ± 0.010
100	20	(40)→ 120, relu → 2, sigmoid	50	100	0.833 ± 0.005	0.829 ± 0.008
100	20	(40)→ 80, relu → 70, relu → 2, sigmoid	50	100	0.825 ± 0.004	0.809 ± 0.017
200	20	(40)→ 120, relu → 2, sigmoid	100	100	0.844 ± 0.003	0.842 ± 0.007
200	20	(40)→ 120, relu → 120, relu → 110, relu → 2, sigmoid	100	100	0.862 ± 0.001	0.845 ± 0.004
300	20	(40)→ 120, relu → 120, relu → 110, relu → 2, sigmoid	100	100	0.862 ± 0.002	0.849 ± 0.005
100	30	(60)→ 70, relu → 2, sigmoid	50	100	0.804 ± 0.008	0.791 ± 0.014
100	30	(60)→ 120, relu → 2, sigmoid	50	100	0.832 ± 0.006	0.818 ± 0.014
200	30	(60)→ 120, relu → 2, sigmoid	100	100	0.843 ± 0.007	0.830 ± 0.009
200	30	(60)→ 200, relu → 2, sigmoid	100	100	0.876 ± 0.003	0.866 ± 0.008
300	30	(60)→ 120, relu → 2, sigmoid	100	100	0.854 ± 0.007	0.846 ± 0.012
300	30	(60)→ 180, relu → 2, sigmoid	100	100	0.879 ± 0.002	0.873 ± 0.011
300	30	(60)→ 215, relu → 2, sigmoid	100	100	0.884 ± 0.002	0.878 ± 0.007
300	30	(60)→ 215, relu → 200 → 2, sigmoid	100	100	0.889 ± 0.001	0.865 ± 0.016

Table 1: Summary of the most efficient models we trained for the classification network for the M-R input model. We vary the number (n) of noise injections we perform on each of the doublet, the size of the observation set (N), the number of layers and nodes. We trained each model for 100 epochs, and we found that a batch size of 100 is optimal for most configurations. However, for most of the cases where $n = 100$, a batch size of 50 was preferred. We trained each network 20 times, and report the mean binary accuracy and validation binary accuracy with their respective standard deviation.

Classification NN						
n	N	Structure	batch size	epochs	binary accuracy	val. binary accuracy
100	15	(45)→ 35, relu → 2, sigmoid	50	100	0.761 ± 0.013	0.754 ± 0.013
100	15	(45)→ 90, relu → 2, sigmoid	50	100	0.808 ± 0.003	0.803 ± 0.007
100	15	(45)→ 90, relu → 70, relu → 2, sigmoid	50	100	0.816 ± 0.003	0.806 ± 0.006
200	15	(45)→ 90, relu → 2, sigmoid	100	100	0.821 ± 0.002	0.813 ± 0.008
200	15	(45)→ 90, relu → 70, relu → 2, sigmoid	100	100	0.827 ± 0.002	0.821 ± 0.008
100	20	(60)→ 90, relu → 2, sigmoid	50	100	0.808 ± 0.006	0.800 ± 0.011
100	20	(60)→ 120, relu → 2, sigmoid	50	100	0.821 ± 0.006	0.812 ± 0.009
100	20	(60)→ 120, relu → 70, relu → 2, sigmoid	50	100	0.831 ± 0.003	0.817 ± 0.006
200	20	(60)→ 120, relu → 2, sigmoid	100	100	0.839 ± 0.005	0.833 ± 0.011
200	20	(60)→ 120, relu → 70, relu → 2, sigmoid	100	100	0.845 ± 0.003	0.835 ± 0.012
300	20	(60)→ 120, relu → 2, sigmoid	100	100	0.849 ± 0.003	0.845 ± 0.006
300	20	(60)→ 120, relu → 70, relu → 2, sigmoid	100	100	0.853 ± 0.003	0.848 ± 0.005
100	30	(90)→ 120, relu → 2, sigmoid	50	100	0.818 ± 0.007	0.802 ± 0.012
100	30	(90)→ 180, relu → 2, sigmoid	50	100	0.830 ± 0.004	0.819 ± 0.012
100	30	(90)→ 180, relu → 120, relu → 2, sigmoid	50	100	0.840 ± 0.005	0.822 ± 0.011
200	30	(90)→ 180, relu → 2, sigmoid	100	100	0.850 ± 0.005	0.844 ± 0.008
200	30	(90)→ 180, relu → 120, relu → 2, sigmoid	100	100	0.867 ± 0.003	0.853 ± 0.009
300	30	(90)→ 180, relu → 2, sigmoid	100	100	0.862 ± 0.006	0.857 ± 0.008
300	30	(90)→ 180, relu → 120, relu → 2, sigmoid	100	100	0.876 ± 0.003	0.864 ± 0.012
300	30	(90)→ 270, relu → 2, sigmoid	100	100	0.878 ± 0.003	0.872 ± 0.011

Table 2: Summary of the most efficient models we trained for the classification network for the M-R-k_2 input model. We vary the number (n) of noise injections we perform on each of the triplet, the size of the observation set (N), the number of layers and nodes. We trained each model for 100 epochs, and we found that a batch size of 100 is optimal for most configurations. However, for most of the cases where $n = 100$, a batch size of 50 was preferred. We trained each network 20 times, and report the mean binary accuracy and validation binary accuracy with their respective standard deviation.

Regression NN						
n	N	Structure	batch size	epochs	msle train_loss	msle val_loss
100	15	(30)→ 35, relu → 15, sigmoid	100	100	0.01819 ± 0.00003	0.01823 ± 0.00003
100	15	(30)→ 45, relu → 15, sigmoid	100	100	0.01815 ± 0.00003	0.01814 ± 0.00003
200	15	(30)→ 35, relu → 15, sigmoid	100	100	0.01809 ± 0.00002	0.01810 ± 0.00003
200	15	(30)→ 45, relu → 15, sigmoid	100	100	0.01806 ± 0.00002	0.01807 ± 0.00003
100	20	(40)→ 35, relu → 15, sigmoid	100	100	0.01814 ± 0.00003	0.01821 ± 0.00004
100	20	(40)→ 45, relu → 15, sigmoid	100	100	0.01810 ± 0.00004	0.01818 ± 0.00004
200	20	(40)→ 35, relu → 15, sigmoid	100	100	0.01807 ± 0.00004	0.01811 ± 0.00004
200	20	(40)→ 45, relu → 15, sigmoid	100	100	0.01804 ± 0.00002	0.01808 ± 0.00003
100	30	(60)→ 45, relu → 15, sigmoid	100	100	0.01809 ± 0.00003	0.01814 ± 0.00005
200	30	(60)→ 60, relu → 15, sigmoid	100	100	0.01802 ± 0.00002	0.01807 ± 0.00005
300	30	(60)→ 60, relu → 15, sigmoid	100	100	0.01800 ± 0.00002	0.01804 ± 0.00004
300	30	(60)→ 70, relu → 15, sigmoid	100	100	0.01799 ± 0.00002	0.01802 ± 0.00004

Table 3: Summary of the most efficient models we trained for the regression network for the M-R input model. We vary the number (n) of noise injections we perform on each of the triplet, the size of the observation set (N), the number of layers and nodes. We trained each model for 100 epochs, and we found that a batch size of 100 is the most optimal choice. We trained each network 20 times, and report the mean training and validation losses with their respective standard deviation.

Regression NN						
n	N	Structure	batch size	epochs	msle train_loss	msle val_loss
100	15	(45)→ 45, relu → 15, sigmoid	100	100	0.01815 ± 0.00003	0.01816 ± 0.00006
100	15	(45)→ 65, relu → 15, sigmoid	100	100	0.01809 ± 0.00001	0.01810 ± 0.00003
200	15	(45)→ 45, relu → 15, sigmoid	100	100	0.01810 ± 0.00003	0.01817 ± 0.00005
200	15	(45)→ 65, relu → 15, sigmoid	100	100	0.01804 ± 0.00002	0.01811 ± 0.00003
200	15	(45)→ 90, relu → 15, sigmoid	100	100	0.01813 ± 0.00009	0.01808 ± 0.00003
100	20	(60)→ 65, relu → 15, sigmloid	100	100	0.01807 ± 0.00002	0.01813 ± 0.00006
100	20	(60)→ 90, relu → 15, sigmoid	100	100	0.01803 ± 0.00001	0.01808 ± 0.00002
100	20	(60)→ 120, relu → 15, sigmoid	100	100	0.01801 ± 0.00001	0.01806 ± 0.00004
200	20	(60)→ 90, relu → 15, sigmoid	100	100	0.01798 ± 0.00001	0.01805 ± 0.00002
200	20	(60)→ 120, relu → 15, sigmoid	100	100	0.01798 ± 0.00001	0.01799 ± 0.00003
300	20	(60)→ 120, relu → 15, sigmoid	100	100	0.01795 ± 0.00001	0.01800 ± 0.00002
100	30	(90)→ 90, relu → 15, sigmoid	100	100	0.01805 ± 0.00001	0.01808 ± 0.00004
100	30	(90)→ 180, relu → 15, sigmoid	100	100	0.01799 ± 0.00001	0.01804 ± 0.00003
200	30	(90)→ 90, relu → 15, sigmoid	100	100	0.01802 ± 0.00001	0.01803 ± 0.00003
200	30	(90)→ 180, relu → 15, sigmoid	100	100	0.01796 ± 0.00001	0.01799 ± 0.00004
300	30	(90)→ 90, relu → 15, sigmoid	100	100	0.01799 ± 0.00002	0.01800 ± 0.00003
300	30	(90)→ 180, relu → 15, sigmoid	100	100	0.01794 ± 0.00001	0.01796 ± 0.00006

Table 4: Summary of the most efficient models we trained for the regression network for the M-R-k_2 input model. We vary the number (n) of noise injections we perform on each of the triplet, the size of the observation set (N), the number of layers and nodes. We trained each model for 100 epochs, and we found that a batch size of 100 is the most optimal choice. We trained each network 20 times, and report the mean training and validation losses with their respective standard deviation.