Layer 1 – 51 input x 32 + 1 bias x 32 = 1696

Layer 2 – 32 x 16 + 16 = 528

Layer 3 – 16 x 1 + 1 = 17

Total number of parameters in initial model – 2241

Number of layers used in initial architecture – 3

Added another layer with 24 neurons significantly improved the results. I noticed that MAPE decreased while others increased as a better result.

Added another layer with 8 neurons led to better results but not as big of a jump.

When epochs size was increased up to 30, it took 4.5mins for training time and better results. However, when the batch size was increased to 64 too, the training time accelerated, but compromised by a marginally worse result, compared to batch size 32.

With epochs 30 and batch size 64, 100, 75, 50, 25, 1 model training time didn't take longer despite having more neurons.

(150, 100, 75, 50, 25, 1) & (400, 350, 300, 250, 200, 150, 100, 75, 50, 25, 1) model results difference was not significant despite the huge difference in number of layer and neurons.

The main reason why the deeper model improved results was because the dataset was too large while the initial model was too shallow, but deeper models can deal with more complexity and avoid underfitting.

Appendix

32, 16, 1 2m 36.8s	,Train,Test Root Mean Squared Error,87804.67,87678.33 Mean Aboslute Error,66581.69,66432.94 Mean Aboslute Percentage Error,11.19,11.18 R2 score,0.81,0.81	
32, 24, 16, 1 1m 38.5s	,Train,Test Root Mean Squared Error,65399.31,65208.67 Mean Aboslute Error,47588.78,47498.03 Mean Aboslute Percentage Error,7.76,7.75 R2 score,0.9,0.9	MAPE decreased while others increased Adding a layer significant increased results
32, 24, 16, 8, 1 1m 42.1s	,Train,Test Root Mean Squared Error,62169.1,62035.74 Mean Aboslute Error,45205.54,45149.15 Mean Aboslute Percentage Error,7.37,7.38 R2 score,0.91,0.91	better results but not a big jump
100, 75, 50, 25, 1 1m 41.7s	,Train,Test Root Mean Squared Error,59549.8,59602.35 Mean Aboslute Error,43298.5,43426.1 Mean Aboslute Percentage Error,7.11,7.15 R2 score,0.91,0.91	
400, 300, 200, 100, 1 2m 38.2s	,Train,Test Root Mean Squared Error,56469.18,56565.21 Mean Aboslute Error,40377.47,40403.28 Mean Aboslute Percentage Error,6.49,6.5 R2 score,0.92,0.92	
32, 24, 16, 8, 1 Epochs - 30 4m 34.4s	,Train,Test Root Mean Squared Error,59324.25,59214.45 Mean Aboslute Error,42555.36,42443.29 Mean Aboslute Percentage Error,6.86,6.86 R2 score,0.91,0.91	Longest training time
32, 24, 16, 8, 1 Epochs – 30 Batch size - 64 2m 39.8s	,Train,Test Root Mean Squared Error,59906.21,59927.08 Mean Aboslute Error,43199.52,43270.01 Mean Aboslute Percentage Error,7.04,7.06 R2 score,0.91,0.91	Increase epochs -> increase batch size Helps with slow training But worse results

100, 75, 50, 25, 1 Epochs – 30 Batch size - 64 2m 27s	,Train,Test Root Mean Squared Error,56769.82,56837.26 Mean Aboslute Error,41179.77,41217.77 Mean Aboslute Percentage Error,6.72,6.74 R2 score,0.92,0.92	Didn't take longer despite more neurons
150, 100, 75, 50, 25, 1 Epochs – 30 Batch size - 64 2m 54.4s	,Train,Test Root Mean Squared Error,54782.08,55042.06 Mean Aboslute Error,39173.2,39222.67 Mean Aboslute Percentage Error,6.32,6.34 R2 score,0.93,0.93	
400, 350, 300, 250, 200, 150, 100, 75, 50, 25, 1 Epochs – 30 Batch size - 64 7m 36.9s	,Train,Test Root Mean Squared Error,54165.67,54839.63 Mean Aboslute Error,38735.77,39043.55 Mean Aboslute Percentage Error,6.19,6.25 R2 score,0.93,0.93	The results increase was marginal
400, 350, 300, 250, 200, 150, 100, 75, 50, 25, 1 Epochs – 10 Batch size - 32 4m 58.2s	,Train,Test Root Mean Squared Error,54480.33,54914.13 Mean Aboslute Error,39275.98,39474.93 Mean Aboslute Percentage Error,6.37,6.41 R2 score,0.93,0.93	