3 Multilayer Perceptron Regression

In this task, we implement a Multi-layer Perceptron (MLP) for regression from scratch. Using the Boston Housing dataset, we have to predict housing prices
while following standard machine learning practices.

Arnav Negi

Data exploration

The Boston Housing Dataset contains info about houses and their values in 1,000 dollars. We will be trying to use regression to predict the values.

Description:

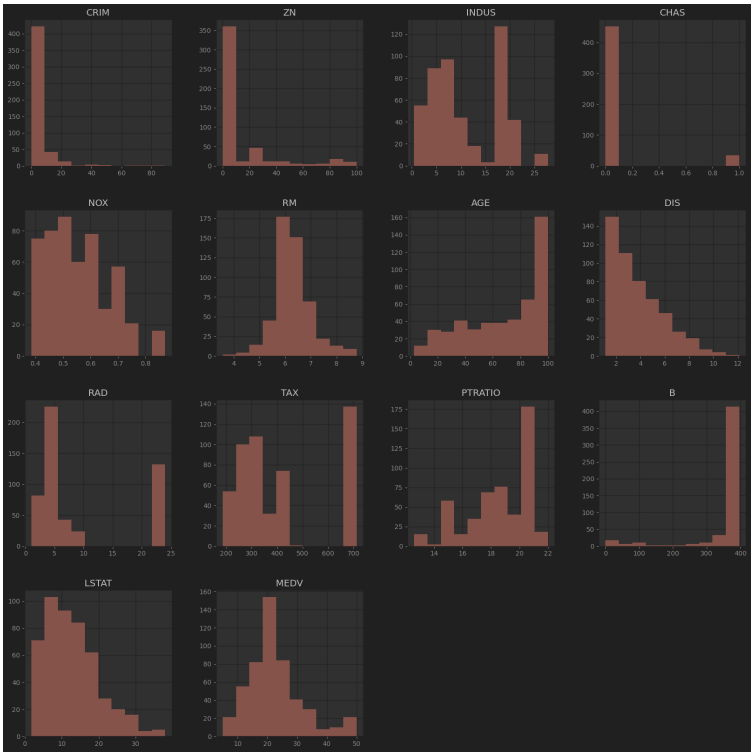
```
boston.describe()
```

8 rows × 14 columns pd.DataFrame

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
count	486.000000	486.000000	486.000000	486.000000	506.000000	506.000000	486.000000	506.000000	506.000000	506.000000	506.000000	506.000000	486.00000
mean	3.611874	11.211934	11.083992	0.069959	0.554695	6.284634	68.518519	3.795043	9.549407	408.237154	18.455534	356.674032	12.71543
std	8.720192	23.388876	6.835896	0.255340	0.115878	0.702617	27.999513	2.105710	8.707259	168.537116	2.164946	91.294864	7.15587
min	0.006320	0.000000	0.460000	0.000000	0.385000	3.561000	2.900000	1.129600	1.000000	187.000000	12.600000	0.320000	1.73000
25%	0.081900	0.000000	5.190000	0.000000	0.449000	5.885500	45.175000	2.100175	4.000000	279.000000	17.400000	375.377500	7.12500
50%	0.253715	0.000000	9.690000	0.000000	0.538000	6.208500	76.800000	3.207450	5.000000	330.000000	19.050000	391.440000	11.43000
75%	3.560263	12.500000	18.100000	0.000000	0.624000	6.623500	93.975000	5.188425	24.000000	666.000000	20.200000	396.225000	16.95500
max	88.976200	100.000000	27.740000	1.000000	0.871000	8.780000	100.000000	12.126500	24.000000	711.000000	22.000000	396.900000	37.97000

We needed to impute missing data, for which we have used the median values.

The data distribution is as follows:



The MEDV value had outliers, houses valued much higher than 50,000 USD were censored. Hence any sample with value >= 50,000 is removed.

Then the following features were dropped:

- CHAS
- ZN

These values were mostly 0s and would not help with our task.

We then preprocess the data by imputing and normalizing it.

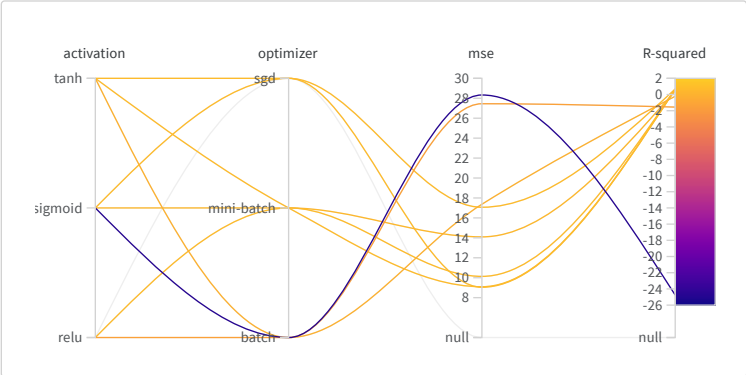
Regression model

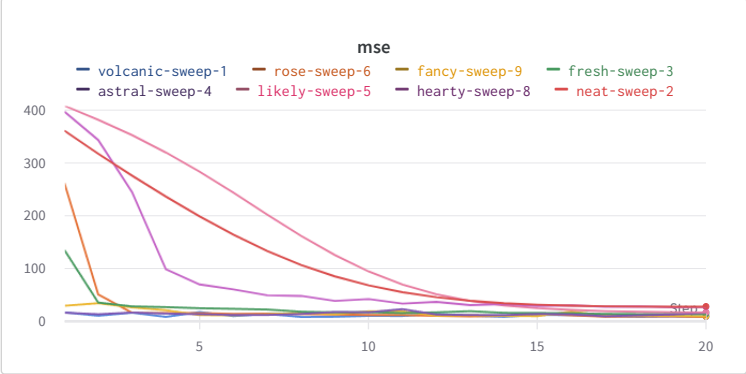
We use a multilayer perceptron based regressor for this task. This includes the following hyperparameters:

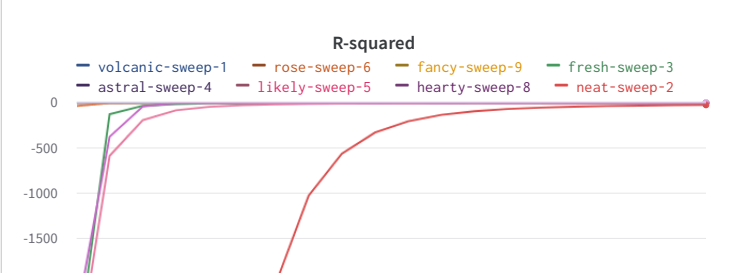
- epochs
- learning rate
- hidden layers
- activation
- optimizer

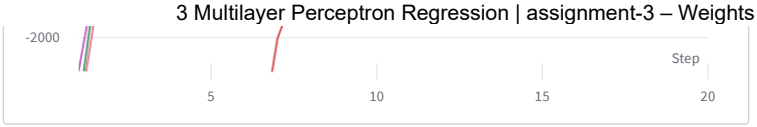
We hyperparameter tune as follows:

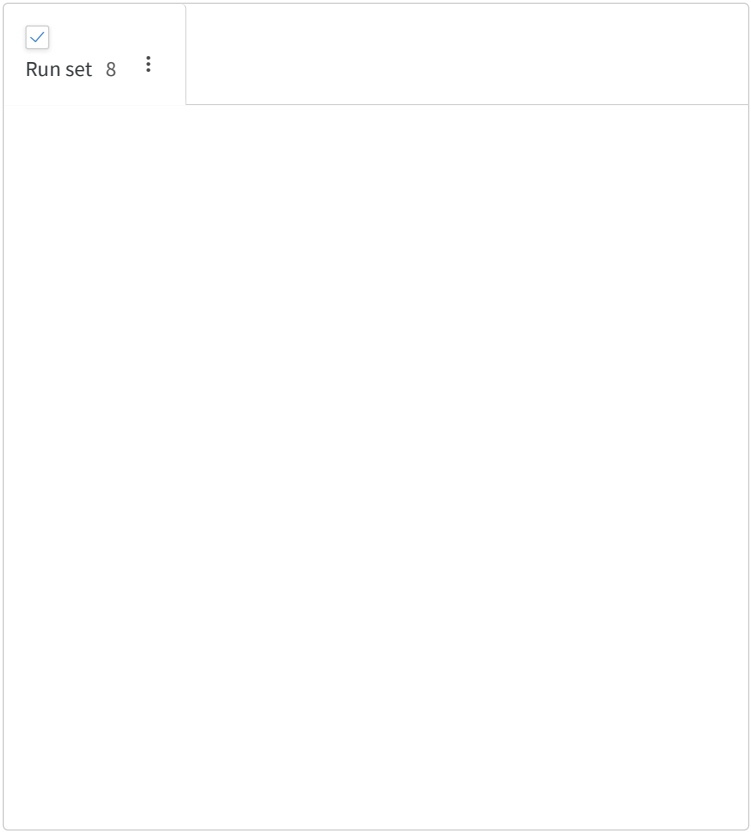
First we tune for activations and optimizers







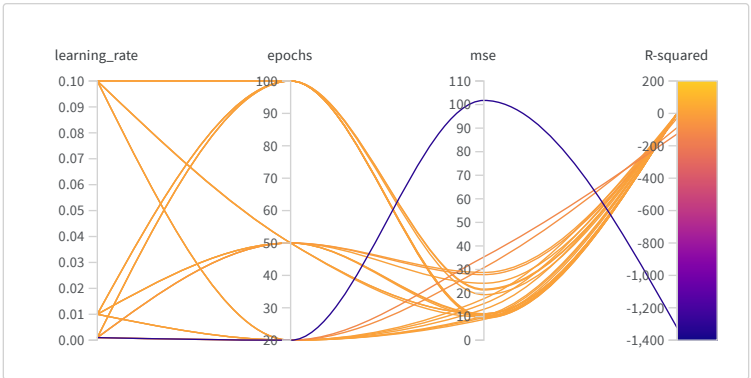


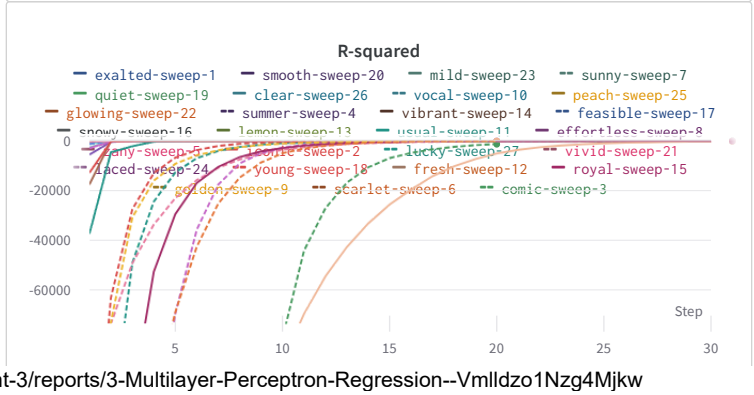


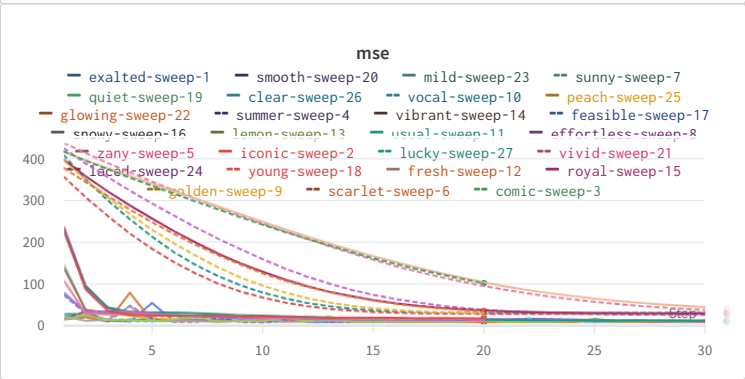
The best performing model seems to be with 9.045 mse and

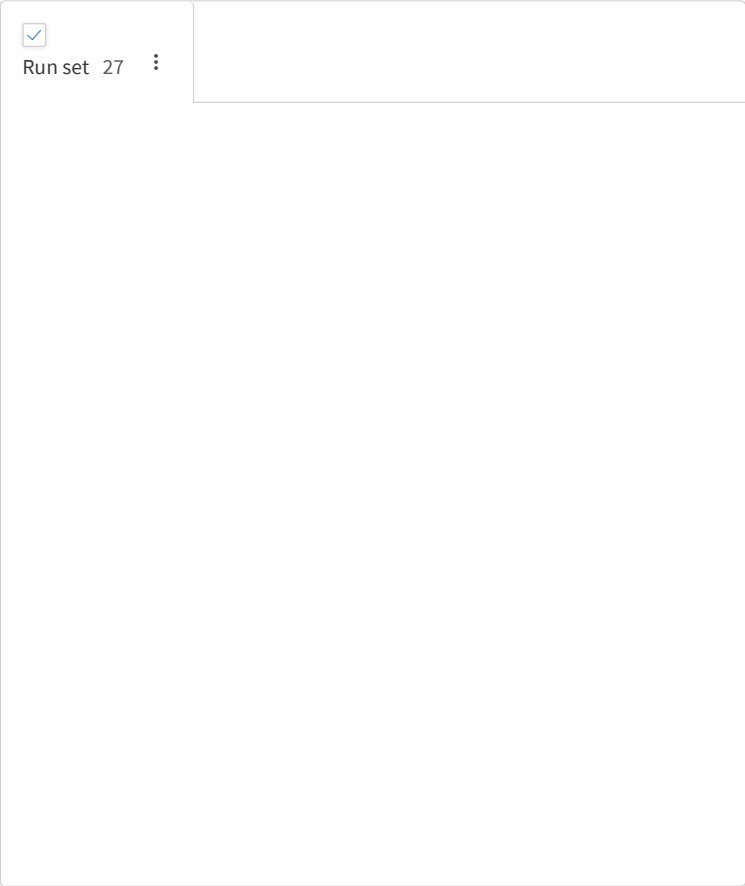
- activation = sigmoid
- optimizer = SGD

Next we tune for hidden layers, epochs and learning rate









We can see the best performing combination has mse 8.826 and has

- epochs = 20
- hidden layers = [10, 10]
- learning rate = 0.1

Created with ❤️ on Weights & Biases.

https://wandb.ai/arnav-team/assignment-3/reports/3-Multilayer-Perceptron-Regression--Vmlldzo1Nzg4Mjkw