*Homework 1: Linear Regression and Neural Network Regression*

Step 1:

1. How many data samples are included in the dataset?

Ans. 3047 rows each having 34 features, so total is 103598.

1. Which problem will this dataset try to address?

Ans. The dataset aims to predict the results of Cancer Mortality Rates. In other words, it's a regression problem where you want to predict the "TARGET\_deathRate."

1. What is the minimum value and the maximum value in the dataset?

Ans.

*Minimum values:*

avgAnnCount 6.000000

avgDeathsPerYear 3.000000

TARGET\_deathRate 59.700000

incidenceRate 201.300000

medIncome 22640.000000

popEst2015 827.000000

povertyPercent 3.200000

studyPerCap 0.000000

binnedInc 0.000000

MedianAge 22.300000

MedianAgeMale 22.400000

MedianAgeFemale 22.300000

Geography 0.000000

AvgHouseholdSize 0.022100

PercentMarried 23.100000

PctNoHS18\_24 0.000000

PctHS18\_24 0.000000

PctSomeCol18\_24 7.100000

PctBachDeg18\_24 0.000000

PctHS25\_Over 7.500000

PctBachDeg25\_Over 2.500000

PctEmployed16\_Over 17.600000

PctUnemployed16\_Over 0.400000

PctPrivateCoverage 22.300000

PctPrivateCoverageAlone 15.700000

PctEmpPrivCoverage 13.500000

PctPublicCoverage 11.200000

PctPublicCoverageAlone 2.600000

PctWhite 10.199155

PctBlack 0.000000

PctAsian 0.000000

PctOtherRace 0.000000

PctMarriedHouseholds 22.992490

BirthRate 0.000000

*Maximum values:*

avgAnnCount 3.815000e+04

avgDeathsPerYear 1.401000e+04

TARGET\_deathRate 3.628000e+02

incidenceRate 1.206900e+03

medIncome 1.256350e+05

popEst2015 1.017029e+07

povertyPercent 4.740000e+01

studyPerCap 9.762309e+03

binnedInc 9.000000e+00

MedianAge 6.240000e+02

MedianAgeMale 6.470000e+01

MedianAgeFemale 6.570000e+01

Geography 3.046000e+03

AvgHouseholdSize 3.970000e+00

PercentMarried 7.250000e+01

PctNoHS18\_24 6.410000e+01

PctHS18\_24 7.250000e+01

PctSomeCol18\_24 7.900000e+01

PctBachDeg18\_24 5.180000e+01

PctHS25\_Over 5.480000e+01

PctBachDeg25\_Over 4.220000e+01

PctEmployed16\_Over 8.010000e+01

PctUnemployed16\_Over 2.940000e+01

PctPrivateCoverage 9.230000e+01

PctPrivateCoverageAlone 7.890000e+01

PctEmpPrivCoverage 7.070000e+01

PctPublicCoverage 6.510000e+01

PctPublicCoverageAlone 4.660000e+01

PctWhite 1.000000e+02

PctBlack 8.594780e+01

PctAsian 4.261942e+01

PctOtherRace 4.193025e+01

PctMarriedHouseholds 7.807540e+01

BirthRate 2.132616e+01

1. How features in each data sample?

Ans. 34

1. Does the dataset have any missing information? E.g., missing features.

Ans. Yes, Columns with missing values along with the total cells missing:

PctSomeCol18\_24 2285

PctEmployed16\_Over 152

PctPrivateCoverageAlone 609

1. What is the label of this dataset?

Ans. The label of this dataset is "TARGET\_deathRate."

1. How many percent of data will you use for training, validation and testing?

Ans. 70% for training, 15% for validation, and 15% for testing.

1. What kind of data pre-processing will you use for your training dataset?

Ans. I would first handle the missing values by replacing them with the mean of the values of that column. I would convert the non-integer variables to integer by applying label encoding. I would also split the data 70-15-15 for training, validation and testing respectively. Then I would ensure that all the features are on a similar scale by checking for non-numeric columns and clipping my y validation and y predicted values to prevent outliers.

**Step 2:**

|  |  |
| --- | --- |
| Model | MSE |
| Linear Regression | 0.00459 |
| ANN-oneL-16 | 0.00605 |
| ANN-twoL-32-8 | 0.00436 |
| ANN-threeL-32-16-8 | 0.00323 |
| ANN-fourL-32-16-8-4 | 0.00335 |

**Step 3: Objective**

I used the MSE as the loss function to train my models.

**Step 4: Optimization**

I used the Adam optimizer for training purposes.

**Step 5: Model Selection**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | LR: 0.1 | LR: 0.01 | LR: 0.001 | LR: 0.0001 |
| ANN-oneL-16 | MSE: 0.00766  R^2:  0.70824 | MSE: 0.00605  R^2:  0.76957 | MSE: 0.01185  R^2:  0.55899 | MSE: 0.01373  R^2:  0.47733 |
| ANN-twoL-32-8 | MSE: 0.00436  R^2:  0.83401 | MSE: 0.00691  R^2:  0.73675 | MSE: 0.01571  R^2:  0.40203 | MSE: 0.011773  R^2:  0.55191 |
| ANN-threeL-32-16-8 | MSE: 0.00567  R^2:  0.78406 | MSE: 0.00323  R^2:  0.87688 | MSE: 0.01118  R^2:  0.57414 | MSE: 0.01469  R^2:  0.44065 |
| ANN-fourL-32-16-8-4 | MSE: 0.00335  R^2:  0.79614 | MSE: 0.00709  R^2:  0.72987 | MSE: 0.01309  R^2:  0.50166 | MSE: 0.01328  R^2:  0.49435 |

Step 6: Model Selection

I’ve reported all the MSE values along with their respective R-squared values in my last step. All of my model performance plots are in the screenshots folder that I would have submitted on Canvas alongside with this document and my code. My best performing ANN model was ‘ANN-threeL-32-16-8’ with a learning rate of 0.01. To try increase my accuracy, I kept playing around the number of epochs and batch size. Please take note that the numbers presented in step 5 may not represent the highest level of accuracy I have attained. I have indeed achieved superior levels of accuracy during previous attempts, but just was not able to document it. My MSE for Linear Regression was actually quite better than some of my ANN models, which was a quite surprising observation.