Distance moduli to the galaxies with machine learning regression methods

N.M. Diachenko, A.A.Elyiv, D.V.Dobrycheva, M.Yu.Vasylenko, I.B.Vavilova

Main Astronomical Observatory of the NAS of Ukraine

Motivation

- Determining distances with greater accuracy is a fundamental problem of cosmology
- The amount of data from extragalactic observations is rapidly increasing
-  The Large Synoptic Survey Telescope (LSST), which is expected to be launched in 2021, will detect ≈ 20 million galaxies every night (≈ 30 TB of data)
- Computational power is growing rapidly
- New data driven approaches are being created to identify the hidden patterns of big data as an alternative to physical methods

Astronomy & Astrophysics manuscript no. ML˙elyiv
March 4, 2020

Machine-learning computation of distance modulus for local galaxies

A.A. Elyiv[1], O.V. Melnyk[1], I.B. Vavilova[1], D.V. Dobrycheva[1,2], and V.E. Karachentseva[1]

[1] Main Astronomical Observatory, National Academy of Sciences of Ukraine, 27 Akademika Zabolotnoho St., 04103, Kyiv, Ukraine
e-mail: andrii.elyiv@gmail.com
[2] Bogolyubov Institute for Theoretical Physics of the NAS of Ukraine, 14-b Metrolohichna St., Kyiv, 03143, Ukraine

Received September 30, 2019; accepted

ABSTRACT

Context. Quickly growing computing facilities and an increasing number of extragalactic observations encourage the application of data-driven approaches to uncover hidden relations from astronomical data. In this work we raise the problem of distance reconstruction for a large number of galaxies from available extensive observations.
Aims. We propose a new data-driven approach for computing distance moduli for local galaxies based on the machine-learning regression as an alternative to physically oriented methods. We use key observable parameters for a large number of galaxies as input explanatory variables for training: magnitudes in U, B, I, and K bands, corresponding colour indices, surface brightness, angular size, radial velocity, and coordinates.
Methods. We performed detailed tests of the five machine-learning regression techniques for inference of $m - M$: linear, polynomial, k-nearest neighbours, gradient boosting, and artificial neural network regression. As a test set we selected 91 760 galaxies at $z < 0.2$ from the NASA/IPAC extragalactic database with distance moduli measured by different independent redshift methods.
Results. We find that the most effective and precise is the neural network regression model with two hidden layers. The obtained root-mean-square error of 0.35 mag, which corresponds to a relative error of 16%, does not depend on the distance to galaxy and is comparable with methods based on the Tully-Fisher and Fundamental Plane relations. The proposed model shows a 0.44 mag (20%) error in the case of spectroscopic redshift absence and is complementary to existing photometric redshift methodologies. Our approach has great potential for obtaining distance moduli for around 250 000 galaxies at $z < 0.2$ for which the above-mentioned parameters are already observed.

Key words. Galaxies: statistics, distances and redshifts, photometry – Methods: data analysis

Input parameters and target variable

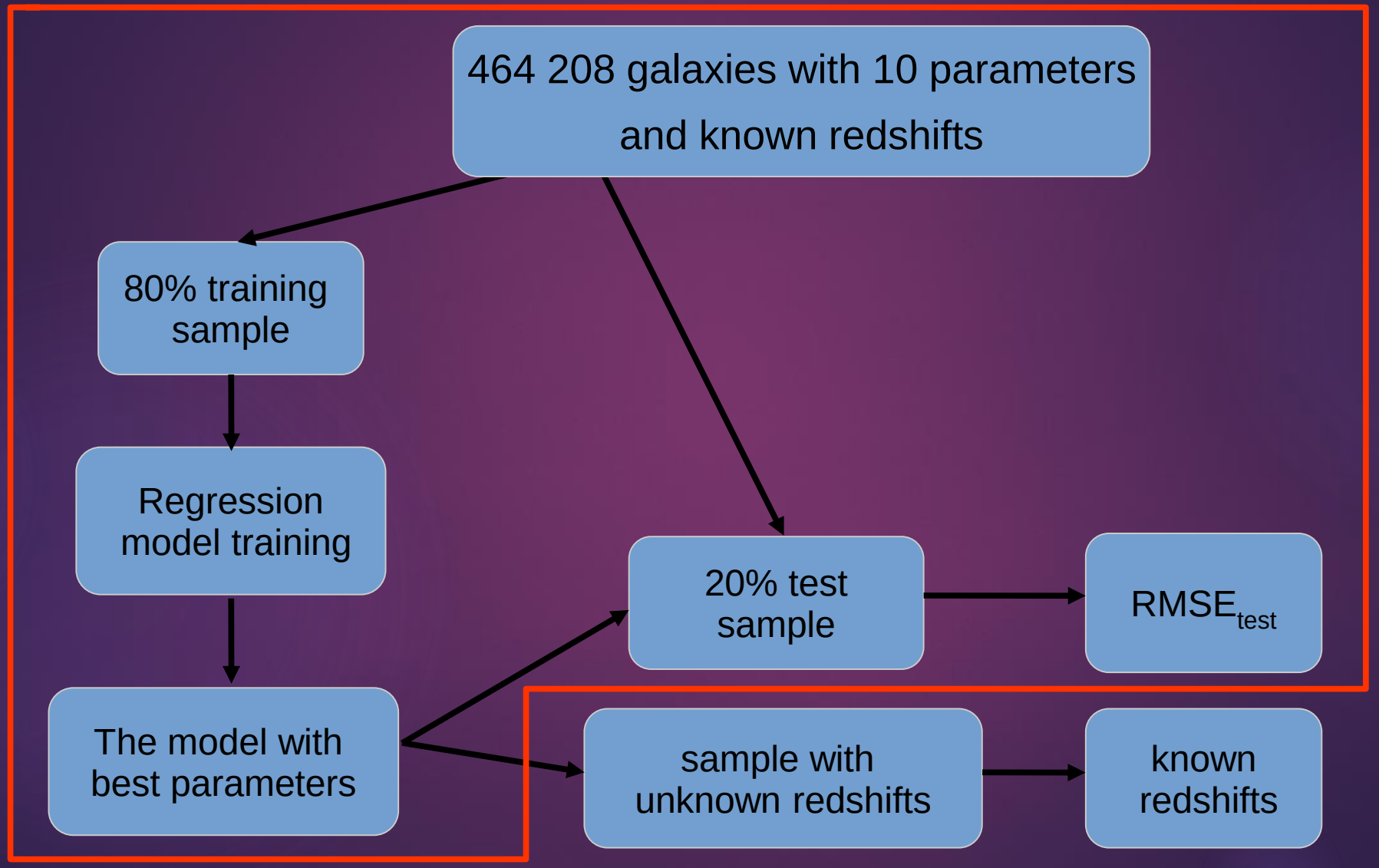
- The sample of galaxies consists of 464 208 objects from SDSS DR14
- The following easily observable parameters were used for training:
  - Visible magnitudes in g-, r-, i-, z- bands
  - Angular radii of galaxies in the same bands
  - Celestial coordinates
  - Redshifts $0.2 < z < 1.0$
- The input parameters were centered and normalized:

$$X = \frac{Xo - \mu}{\sigma},$$

μ – mean, σ – standard deviation

- The target variable – redshift z

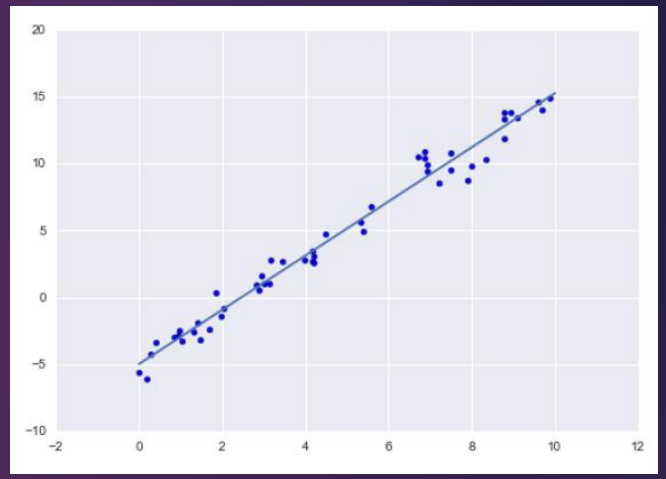
Scheme of "supervised learning"



Regression models

- Linear regression:
  - This is a basic regression model, which deals with linear combinations of input variables (also called as features or attributes)
  - Multidimensional linear regression is a system that takes a *n*-size vector of input explanatory variables $x \in R^n$ and predicts a scalar $y \in R$ with some approximation $\tilde{y} \in R$:

$$\tilde{y} = \boldsymbol{w}^T \boldsymbol{x} + b = w_1 x_1 + \ldots + w_n x_n + b \ ,$$



$\boldsymbol{w}$ – weights of feature's contribution to composed output value

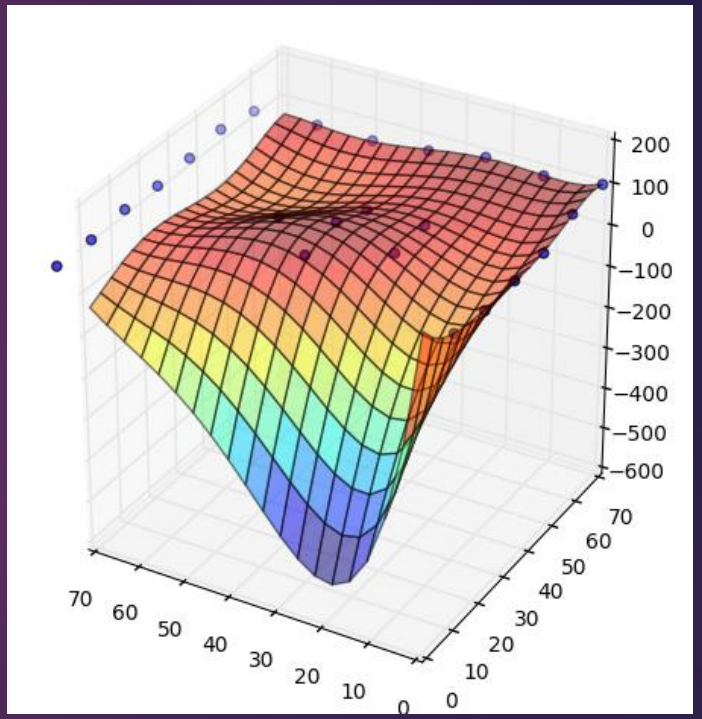
  - The mean squared error (MSE) of predicted output values:

$$MSE = \frac{1}{m} \sum_i^m (\tilde{y}_i - y_i)^2 \ ,$$

m – size of the sample

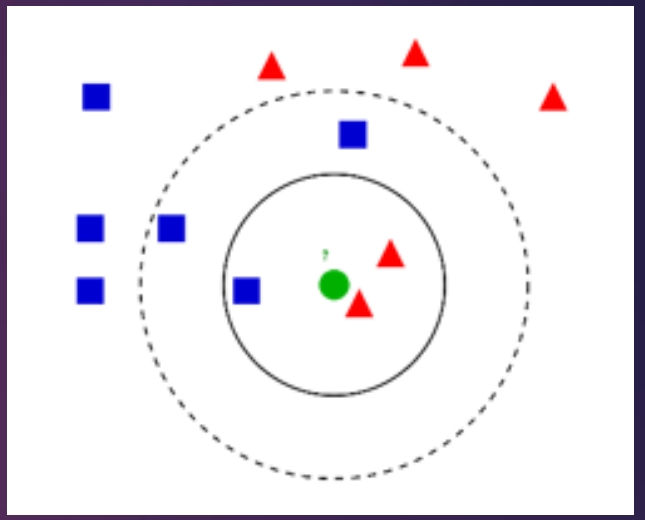
Regression models

- Polynomial regression of *k* = 2, 3 degree:
  - This is an extension of linear one, where between the input explanatory variables **x** and the dependent variable **y** is an k^{th} degree polynomial relation
  - New features consist of all possible combinations of the original features **x** as product $x_i^l \cdot x_j^m$, where *i, j* ∈ *(1, n)* and *l, m* ∈ *(0, k)*
  - Next, the simple linear regression is applied to new features
  - For example, in case of two input features a and b, and k = 2 degrees, the polynomial features are (a, b, a^2, ab, b^2)



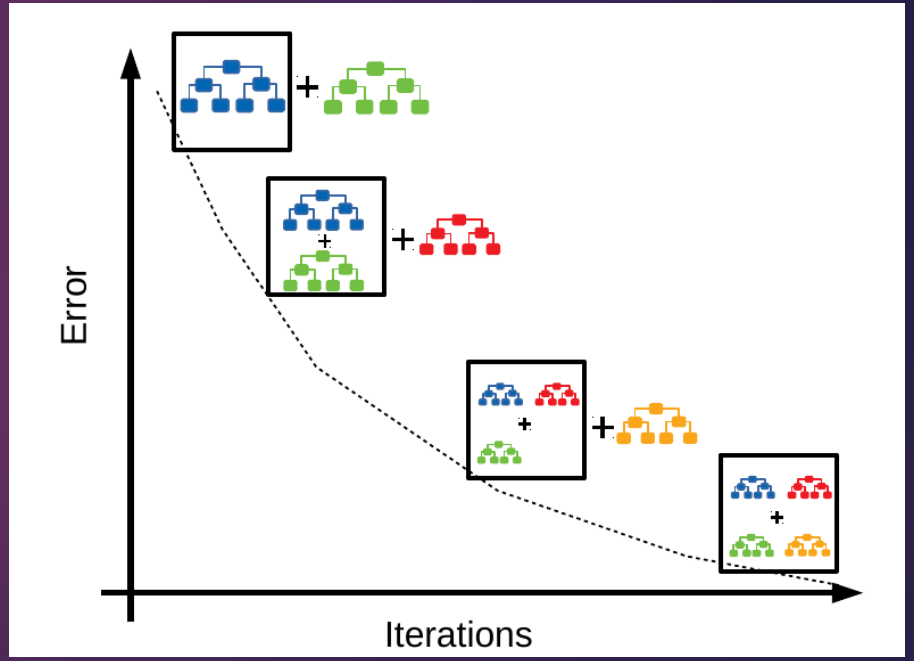
Regression models

- *k*-Nearest Neighbors:
  - This regression uses the *k* closest training points around the test point in the space of features
  - *k*-NN algorithm computes distances between new instance and the training instances to make a decision
  - Predicted variable $\tilde{\boldsymbol{y}}$ is the weighted average of the values $\boldsymbol{y}$ of *k* nearest neighbors
  - *k*-NN regression has one hyperparameter *k*, that is the number of near neighbors taken into account, also it uses 2 types of weights for neighbors: uniform and distance metric



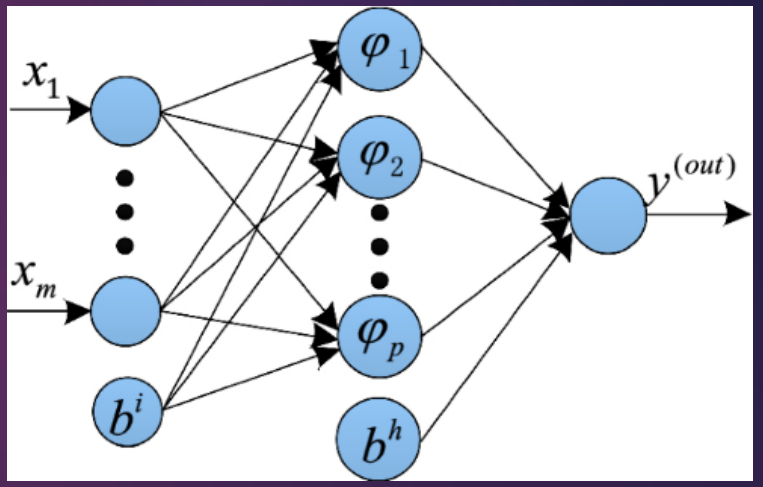
Regression models

- Gradient boosting:
  - This is a stack of simple prediction models like decision tree, linear regression etc, which are joined together to make a final prediction
  - Superposition of these models could show better result than any single predictor alone
  - 2 main approaches of ensembling: bagging and boosting
  - Bagging: all simple models are independent and final result is averaged over each model output
  - Boosting: predictors are lined up sequentially and subsequent predictor learns from the errors of the previous one, reducing these errors



Regression models

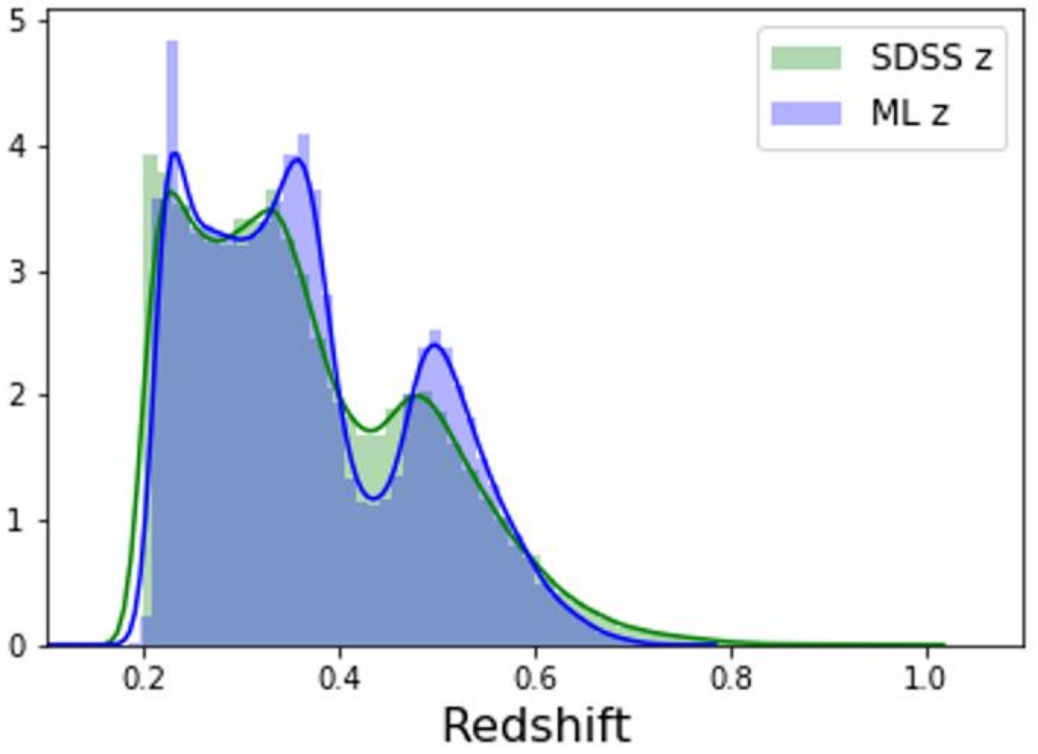
- ANN-regression:
  - Artificial neural network consists of neurons grouped by parallel layers: an input layer, a hidden layer(s) and an output one
  - The main characteristic is an ability to transmit a numerical signal from one artificial neuron to another in feedforward direction from input to output layer
  - The connection between neurons has a weight that corresponds to the importance of signal at its transmission
  - The best model performance was reached by shallow ANN with two hidden layers with 24 and 228 neurons each, respectively



Results

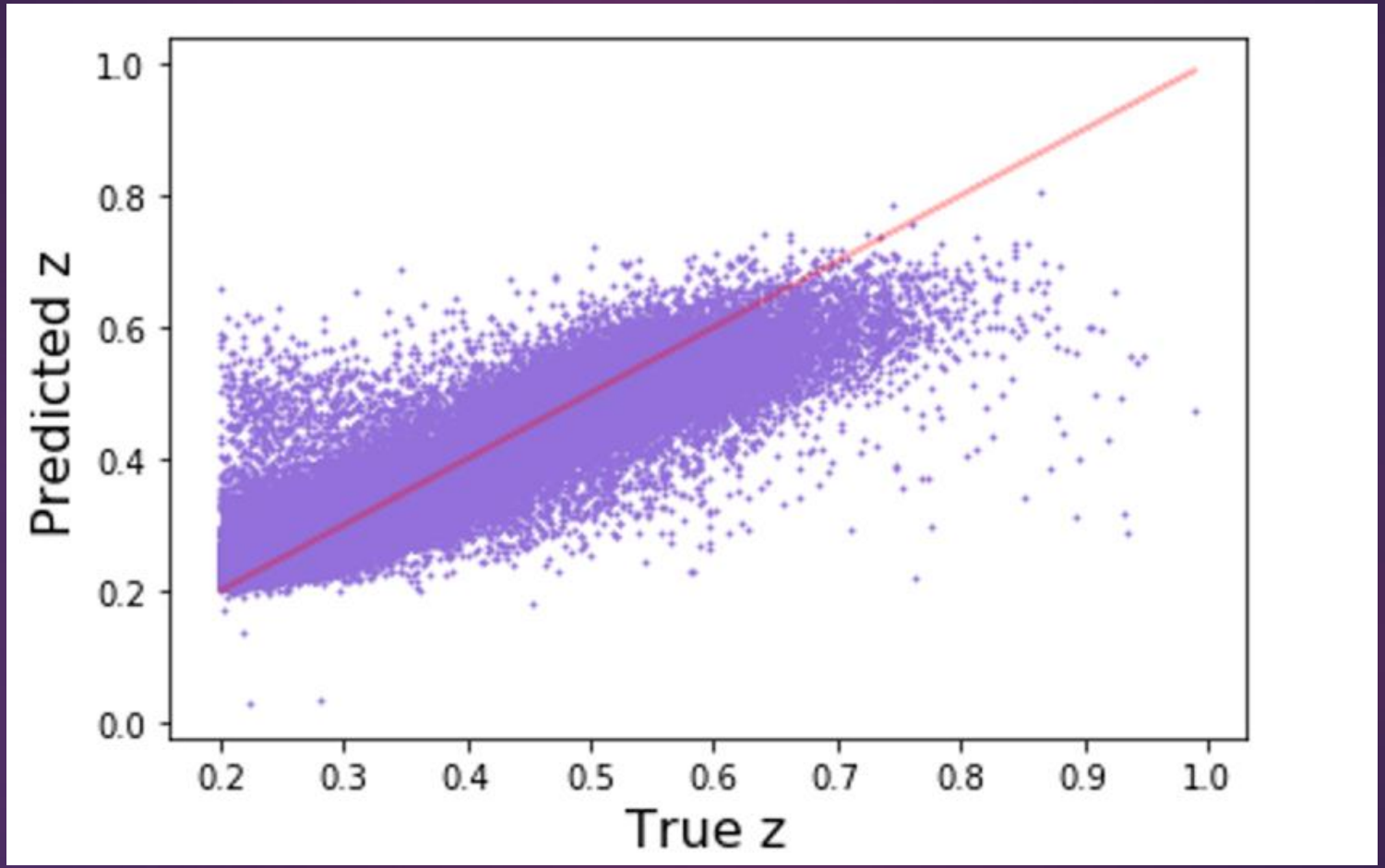
Method	Test RMSE
Linear	0.057
Polynomial	0.054
KNN	0.051
Gradient Boosting	0.050
ANN	0.048

Test root-mean-square errors for each algorithm



Comparison of known and predicted redshifts for ANN

Results



Comparison of known and predicted redshifts for ANN