Machine Learning Engineer Nanodegree

Capstone Proposal

Arbaaz Muslim

January 10th, 2018

Domain Background

This project relates to the field of education. It involves taking student information and using it to determine the student's final math grade. Therefore, the project performs a regression task. The student information provides input features and the student's final math grade is the target feature. The data I plan on using was featured in a report authored by Associate Professor Paulo Cortez at the University of Minho, who provided the data for the UCI Machine Learning Repository (BRITO, A. ; TEIXEIRA, J., eds. lit. – "Proceedings of 5th Annual Future Business Technology Conference, Porto, 2008". [S.l. : EUROSIS, 2008]. ISBN 978-9077381-39-7. p. 5-12.).

A model that accurately performs such a regression task can provide great insight to the factors that impact academic success. As a college student myself, I find this problem particularly interesting and possibly even applicable to my own life.

Problem Statement

This task is a regression problem. It involves taking student information gathered from surveys and using it to predict the student's final math grade. The solution to this task would be a supervised learner. The accuracy of the learner can be quantified via the mean squared error and/or the R^2 value. The data and information about it are available at https://archive.ics.uci.edu/ml/datasets/Student+Performance. In one sentence, the problem can be phrased as follows: Given information about a student, what is the best way to determine the student's final math grade?

Datasets and Inputs

The publicly available dataset used for this project is the Student Performance Data Set from the UCI Machine Learning Repository. The data comes from Associate Professor Paulo Cortez at the University of Minho. The data comes from students in secondary education in two Portuguese schools and was collected through surveys and questionnaires. There are 395 data points available. I will split the data into a training test (296 points) and a testing set (99 points). I will then further divide the training set using k-fold cross validation. Each data point has 33 attributes. These attributes include both student data as well as student grades, making the data both relevant and useful for a solution to the problem. The attribute information can be found at https://archive.ics.uci.edu/ml/datasets/Student+Performance. I intend on using 30 of the 33 features as input (excluding the three features discussed below).

Three of the attributes present are G1, G2, and G3. G1 refers to the grade for the first grading period, G2 refers to the grade for the second grading period, and so on. In this project, I will be making G3, the final grade, the target variable. I will be disregarding G1 and G2 since "G3 has a strong correlation with attributes G2 and G1" and "it is more difficult to predict G3 without G2 and G1, but such prediction is much more useful", according to the site.

Variable	**Type**
School	Categorical
Sex	Categorical
Age	Continuous
Address	Categorical
Family Size	Categorical
Parent's cohabitation status	Categorical
Mother's education	Discrete
Father's education	Discrete
Mother's job	Categorical
Father's job	Categorical
Reason to choose school	Categorical

Guardian	Categorical
Home to school travel time	Discrete
Weekly study time	Discrete
Class failures	Discrete
Educational support	Categorical
Family educational support	Categorical
Extracurricular activities	Categorical
Attended nursery school	Categorical
Wants to take higher education	Categorical
Internet access at home	Categorical
Family relationship qualities	Discrete
Free time	Discrete
Going out with friends frequency	Discrete
Workday alcohol consumption	Discrete
Weekend alcohol consumption	Discrete
Health status	Discrete
Number of absences	Continuous

In the above table, Categorical features need to be one-hot encoded. Discrete features are numerical but limited to a range, while Continuous features are numerical and are not limited to a range.

Solution Statement

The solution to this problem would be a supervised learner that is trained on a subset of the data. Post-training, this learner should be able to take in student data as input (specifically, the student data would be formatted into features just like the training examples) and output a prediction of the student's final math grade with minimal mean squared error and/or a maximal R^2 score.

Benchmark Model

Unfortunately, there is no easily available existing model for this dataset that can serve as a benchmark model. I will be using a simple linear regression model as a benchmark model.

Evaluation Metrics

One evaluation metric relevant to this problem is the mean squared error. This metric is commonly used for regression problems and provides a single number that illustrates the accuracy of predictions in comparison to actual values. The formula for mean squared error is $\frac{1}{n}\ \sum_{i=1}^{n}(\hat{X}_i - X_i)^2$, where n is the number of data points, X_i is an actual value, and X hat is the predicted value. For each input value, the formula involves finding the difference between the predicted and actual value and then squaring it to ensure that the result is positive. These error squares are then averaged.

Another evaluation metric relevant to this problem is the R^2 value, or correlation coefficient. This value illustrates the proportion of variance in an output variable that can be explained by an input variable. The formula for the R^2 value is $R^2 \equiv 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}.$, where $SS_{\text{res}} = \sum_i (y_i - f_i)^2$ and $SS_{\text{tot}} = \sum_i (y_i - \bar{y})^2;$. In these formulas, f_i is the same as y-hat (the prediction of the output value) and y-bar is the average of the actual output values.

Project Design

The general solution to this problem will involve training a number of regression models and then tuning the most accurate one.

The first step of the workflow is to preprocess the data. The first part of doing so would be to separate the features from the target. The features would then have to be scaled or one-hot encoded and the feature data would have to be normalized, most likely through a logarithmic transformation. The next step would be removing irrelevant features (such as G1 and G2) and removing Tukey outliers that are present in multiple feature categories. The next step of the workflow would be shuffling and splitting the data into training and testing sets. The training set would then undergo a k-fold cross-validation split. Then, different regression models would be trained and tested on the divided training data.

Possible models include LogisticRegression(), SGDRegressor(), SVR(), GaussianProcessRegressor(), DecisionTreeRegressor(), GradientBoostingRegressor(), MLPRegressor(), and a deep learning algorithm. The hyperparameters of the model with the best cross-validation score will be tuned using GridSearchCV or RandomizedSearchCV.