Introduction to Machine Learning

ECE 580
Spring 2024

HW #1

Submission Instructions

Submit your work to the corresponding assignment in Gradescope. Although Gradescope accepts multiple file formats, they strongly recommend submitting your assignment as a single PDF file.

It is your responsibility to ensure the uploaded file is: 1) the correct file, 2) complete (includes all pages), 3) legible, and 4) submitted on-time as determined by the Gradescope server system clock.

It is your responsibility to submit a multi-page PDF file and tag the pages that correspond to each question. Pages may be tagged after submission, even if the submission deadline has passed. If you are submitting close to the submission deadline, submit your assignment first then immediately return to tag pages.

When code is requested, submit a PDF print-out of your code. Submitting a URL for a cloud-based repository is insufficient.

Late Submissions

Late submissions will be accepted up to 5 days after the submission deadline, with the following point penalty applied if its late submission is not excused: [1]

1 day (0^+ to 24 hours) late: 2 point deduction ($\frac{1}{5}$ letter grade)
2 days (24^+ to 48 hours) late: 5 point deduction ($\frac{1}{2}$ letter grade)
3 days late: 10 point deduction (1 letter grade)
4 days late: 20 point deduction (2 letter grades)
5 days late: 30 point deduction (3 letter grades)
6 or more days late: score = 0 (not accepted for credit)

The late policy is designed to be minimally punitive for submissions up to 3 days late, yet encourage staying current with the coursework for our course by not allowing one assignment's late submission to overlap with the next assignment's submission.

A homework score will not drop below 0 as a result of applying the late penalty point deduction.

[1] One day = one 24-hour period or fraction thereof.

Structuring and Organizing Your Code

We will use Python this semester.[2] You may choose to use packages/toolboxes authored by others. If you use packages/toolboxes authored by others, you are expected to reference the packages/toolboxes so we know what external code supported your completion of the assignment. You are also responsible for knowing how to use the packages/toolboxes to achieve what you are asked to do. In past semesters, many students have commented toward the end of the semester that, in hindsight, they spent more time looking for functions and figuring out how to make the functions do what they wanted them to do than they would have spent writing their own functions. You may wish to keep this in mind as you decide when to write your own functions and when to leverage existing packages/toolboxes.

The majority of homework assignments this semester will involve coding. I suggest you think about how you can structure and organize your code so it can easily be extended to additional use cases. For example, think about how you can design extensible code so input/output argument lists for your functions do not become unwieldy.

I also suggest thinking about how to modularize your code. For example, if a quantity could be calculated in more than one context, consider making the calculation of that quantity a separate function rather than embedding (and replicating!) the code to calculate that quantity within several functions. When a code block exists once, extending it or correcting it can be achieved by revising that single code block. When a code block is replicated within several functions, it must be revised every place it exists in order to extend it or correct it across all instances.

Data Preparation and Exploration

Download the Automobile Data Set from the UCI Machine Learning Repository.[3] Although there are several potential uses of this data set, we are going to use this data to predict a car's price from its characteristics. For the time being, we are going to **restrict ourselves to the 13 continuous predictor variables** (in the order in which they appear in the data base): wheel-base, length, width, height, curb-weight, engine-size, bore, stroke, compression-ratio, horsepower, peak-rpm, city-mpg, and highway-mpg.

(5) 1. (a) Document (list) the steps you take to clean this data set, including removing the (non-continuous) features that are not of interest, removing observations for which the target variable (price) is unknown, and removing observations for which any of the 13 continuous features are missing. The documentation should include the number of observations that are removed at each step (do not list the removed observations themselves), and conclude with a statement as to the number of observations that remain after cleaning the data set.

(5) (b) In a future assignment, you will explore systematic feature (predictor variable) selection. For now, you are going to select features you believe are most promising via data exploration.

For each of the 13 continuous predictor variables (features), plot the target variable (price) as a function of the predictor variable (feature) using a scatter plot. This will produce 13 scatter plots, one for each feature.

(I recommend writing, or leveraging, a function that accepts a matrix of features and a vector of target variables, and produces a feature vs. target scatter plot for each feature in the data matrix.)

[2] I am working on integrating some auto-grading into Gradescope for "Field Tests" of classifiers later in the semester. This auto-grading is likely to depend on your submitting Python code to Gradescope, so I recommend taking advantage of the opportunities to acclimate yourself to Python in the early homeworks even though the early homeworks do not use auto-grading.

[3] `https://archive.ics.uci.edu/ml/datasets/Automobile`

(5) (c) For each of the 13 features, explain, based on your scatter plots, why you believe that feature holds promise for predicting a car's price, or why you believe it does not hold promise for predicting a car's price. You may notice that for some features there appears to be a nonlinear relationship between the feature and the car's price. For example, $price = feature^2$, or $price = 1/feature$. Keep this in mind when you propose candidate models for predicting a car's price. (There should be 13 explanations, one for each feature.)

(5) (d) When performing regression, it is preferable to have features that are as independent as possible, as strongly related (correlated) features do not provide much additional information and may lead to computational challenges. For example, if there were two additional continuous features, "km per gallon city" and "km per gallon highway", these features would be highly (perfectly?) correlated with the existing features "city-mpg" and "highway-mpg," respectively, because 1 km = 0.6241 miles. For this reason, we would want to include only one of "km per gallon city" and "city-mpg" in our model, and only one of "km per gallon highway" and "highway-mpg" in our model.

Plot each pair-wise combination of features using scatter plots to aid in (visually) identifying features that are related (correlated).

This type of visualization is often referred to as a "scatterplot matrix" or "pairwise scatterplots." (Examples of this type of visualization are available in Figure 3.6 in *Introduction to Statistical Learning* and Figure 1.1 in *Elements of Statistical Learning*.) There will be **a lot** of subplots. Write code (or leverage a package) to do the repetitive heavy lifting for you! (I recommend writing, or leveraging, a function that accepts a matrix of features (and, optionally, a vector of target variables), and produces a set of pair-wise feature scatter plots, perhaps with symbols that may be color-coded by target variable.) Since you are using these subplots to identify correlation trends the subplots do not need to be high-resolution; it is ok if the subplots are "small".

(5) (e) Identify, based on your pair-wise scatter plots, variables that are related and preferably would not both (or all) be used in the model simultaneously.

Explain how you arrived at your conclusions.

(5) (f) Submit a PDF print-out of your code.
(Submitting a URL for a cloud-based repository is insufficient.)

Unregularized Regression

Continuing with the **13 continuous predictor variables** from the Automobile Data Set from the UCI Machine Learning Repository to predict a car's price from its characteristics:

(I *strongly* encourage you to write a single code block to learn and evaluate a model that you then leverage for each of your three models, rather than replicating code for each model individually.)

(5) 2. (a) Based on your data exploration, propose 3 unique linear models (3 unique subsets of the 13 continuous features) for predicting a car's price from its characteristics. The subset of features used in each model must be unique; individual features may be utilized in more than one model. If you noticed that for some features there appears to be a nonlinear relationship between the feature and the car's price, such as $price = feature^2$ or $price = 1/feature$, you may propose a model that uses a transformed feature that you expect to capture the nonlinear relationship you observed. Constrain each of your proposed models to use exactly three features.

Explain why you proposed each of these models as a candidate model to test. (There should be 3 explanations, one for each model.)

(15) (b) For your proposed model #1,

i. Perform unregularized linear regression to find the model parameters, and provide the specific model (*i.e.*, write down the equation $\hat{price} = f(features, \hat{\mathbf{w}})$ with the values for each element of $\hat{\mathbf{w}}$ specified.

(I recommend writing, or leveraging, a function that accepts a matrix of features (or transformed features) and a vector of target variables, and performs linear regression. This function may (optionally) return R^2 and/or other diagnostic information.)

ii. Determine the coefficient of determination, R^2, for this model.

iii. Scatter plot the predicted price (on the vertical axis) as a function of the true price (on the horizontal axis). Also plot the line $\hat{price} = price$ (the line representing perfect prediction) as a reference.

This visualization aids in (visually) evaluating the model quality, by facilitating identification of trends in the estimation errors as a function of the target variable (true price).

iv. Describe your impression of this model. How do the predicted prices compare to the true prices? Do the errors appear to be systematic, or random? Are there price ranges where the model is particularly good? Are there price ranges where the model is particularly bad?

(15) (c) For your proposed model #2,

i. Perform unregularized linear regression to find the model parameters, and provide the specific model (*i.e.*, write down the equation $\hat{price} = f(\text{features}, \hat{\mathbf{w}})$ with the values for each element of $\hat{\mathbf{w}}$ specified.

ii. Determine the coefficient of determination, R^2, for this model.

iii. Scatter plot the predicted price as a function of the true price. Also plot the line $\hat{price} = price$ (the line representing perfect prediction) as a reference.

iv. Describe your impression of this model. How do the predicted prices compare to the true prices? Do the errors appear to be systematic, or random? Are there price ranges where the model is particularly good? Are there price ranges where the model is particularly bad?

(15) (d) For your proposed model #3,

i. Perform unregularized linear regression to find the model parameters, and provide the specific model (*i.e.*, write down the equation $\hat{price} = f(\text{features}, \hat{\mathbf{w}})$ with the values for each element of $\hat{\mathbf{w}}$ specified.

ii. Determine the coefficient of determination, R^2, for this model.

iii. Scatter plot the predicted price as a function of the true price. Also plot the line $\hat{price} = price$ (the line representing perfect prediction) as a reference.

iv. Describe your impression of this model. How do the predicted prices compare to the true prices? Do the errors appear to be systematic, or random? Are there price ranges where the model is particularly good? Are there price ranges where the model is particularly bad?

(5) (e) Which of your three proposed models would you select? Explain your reasoning.

(If coefficients of determination for models are similar, other factors, such as the nature of the errors, may influence the model choice. For example, a model with relatively small errors for all target variables may be preferable to a model with no error for a large fraction of the target variables and large errors for the remaining target variables, even if its coefficient of determination is slightly lower. In other words, quantitative measures of performance are not necessarily the sole determinants of the final model.)

(5) (f) Submit a PDF print-out of your code.
(Submitting a URL for a cloud-based repository is insufficient.)

3. In the case of simple linear regression of t onto x, both the coefficient of determination, R^2, and the sample correlation, r, between x and t are measures of the linear relationship between x and t. It can be shown in this case (simple linear regression) that the coefficient of determination is equal to the square of the sample correlation between x and t, $R^2 = r^2$. (This question is **not** asking you to prove this relationship.)

(*Introduction to Statistical Learning* provides a review of correlation (r) and the R^2 statistic within the contexts of linear regression (section 3.1) and multiple linear regression (section 3.2) that may be helpful when contemplating this question.)

(5) (a) Does the equivalence between the coefficient of determination and the square of the sample correlation extend to multiple linear regression (regression with more than one predictor, or feature, variable)? Why, or why not?

(5) (b) What does R^2 provide that r (or r^2) does not?

Total weight = 100