Problem Set 1

Audrey Glaser

1/15/2020

Problem Set 1: Learning and Regression

Statistical and Machine Learning

1. Describe in 500-800 words the difference between supervised and unsupervised learning.

To answer this question, it's helpful to first define what machine learning is. In contrast to traditional software engineering, which combines human-created rules (programs) with data to discover answers to a given problem, machine learning uses data to discover the rules behind a problem. More specifically, a computer algorithm is given a set of training data along with certain mathematical constraints, which it uses to build a mathematical model which performs a given task related to the data. To quote Professor Tom Mitchell, a machine "is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

Supervised learning and unsupervised learning are the two primary categories of machine learning tasks, though there are other, more niche categories, such as semi-supervised and reinforcement learning.

The goal of supervised learning is to learn a mathematical function that, given a sample of data which contains both the inputs and outputs of interest, best approximates the relationship between inputs and outputs observable in the data. This approximated function, or model, can then be used to make predictions about the (unknown) output values of data from the same population set.

Supervised learning can be further divided into two main categories of tasks: regression and classification. In both regression and classification, the goal is to find the specific relationships or structure in the training data that allow us to make the best possible predictions about output values. (Note that the accuracy of the learned model will be constrained by the quality of the training data.) In regression tasks, the output variable of interest is continuous, and the goal of the model to predict its values for new observations in the data, with minimal error. In classification tasks, the output variable of interest is categorical, and the goal of the model is to assign new observations in the data to those categories (classify the data) with maximal accuracy.

The goal of unsupervised learning is to learn about undiscovered patterns and structure in existing data. Unlike cases of supervised learning, the output values of the learned model are not yet known.

Two of the most common tasks within unsupervised learning are clustering analysis and dimensionality reduction. Clustering tasks involve finding natural clusters, or groups, in the input data based on certain shared attributes. It is possible to set constraints on how many clusters your unsupervised learner should identify, depending on the desired granularity of results. Dimensionality reduction tasks focus on simplifyng the categories of attributes of data by grouping attributes with certain similarities.

In real-world applications, the usefulness of supervised vs. unsupervised learning approaches will depend entirely on the nature of one's problem. For example, real-world sample data often comes to us unlabelled and disorganized, and unsupervised learning tools are a powerful way of discovering inherent structure to the data. However, if the problem at hand isn't understanding a dataset's structure, but forecasting some feature of its population in future scenarios, the problem at hand probably requires a supervised learning approach.

Linear Regression Regression

1. Using the mtcars dataset in R (e.g., run names(mtcars)), answer the following questions:

1a. Predict miles per gallon (mpg) as a function of cylinders (cyl). What is the output and parameter values for your model?

```
mpg_cyl_regress <- mtcars %>% lm(formula = mpg ~ cyl)
summary(mpg_cyl_regress)
```

```
##
## Call:
## lm(formula = mpg ~ cyl, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.9814 -2.1185  0.2217  1.0717  7.5186
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.8846     2.0738   18.27  < 2e-16 ***
## cyl          -2.8758     0.3224   -8.92 6.11e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.206 on 30 degrees of freedom
## Multiple R-squared:  0.7262, Adjusted R-squared:  0.7171
## F-statistic: 79.56 on 1 and 30 DF,  p-value: 6.113e-10
```

1b. Write the statistical form of the simple model in the previous question (i.e., what is the population regression function?).

Y_i = -2.8758X_i + 37.8846 + E_i, where:

Y = Miles per gallon X = Number of cylinders E = Stochastic error term

1c. Add vehicle weight (wt) to the specification. Report the results and talk about differences in coefficient size, effects, etc.

```
mpg_cyl_wt_regress <- mtcars %>% lm(formula = mpg ~ cyl + wt)
summary(mpg_cyl_wt_regress)
```

```
##
## Call:
## lm(formula = mpg ~ cyl + wt, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2893 -1.5512 -0.4684  1.5743  6.1004
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  39.6863     1.7150  23.141  < 2e-16 ***
## cyl          -1.5078     0.4147  -3.636 0.001064 **
```

```
## wt           -3.1910     0.7569  -4.216 0.000222 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.568 on 29 degrees of freedom
## Multiple R-squared:  0.8302, Adjusted R-squared:  0.8185
## F-statistic: 70.91 on 2 and 29 DF,  p-value: 6.809e-12
```

The estimated coefficient size for the new independent variable (vehicle weight) is -3.19, implying that a one-unit change in the weight of a vehicle will lead to a -3.19-unit change in the vehicle's miles per gallon.

When we add the vehicle weight variable to our initial model, the estimated coefficient size for the weight of cylinders variable shrank from -2.8758 to -1.5078. In other words, the predicted effect of a one-unit increase in a vehicle's number of cylinders on a vehicle's MPG is a -1.51 unit decrease when the vehicle's weight is held constant.

Additionally, the addition of the weight variable increased the estimated intercept coefficient slightly, from 37.88 to 39.69.

1d. Interact weight and cylinders and report the results. What is the same or different? What are we theoretically asserting by including a multiplicative interaction term in the function?

```
mpg_cyl_wt_regress_2 <- mtcars %>% lm(formula = mpg ~ cyl*wt)
summary(mpg_cyl_wt_regress_2)
```

```
##
## Call:
## lm(formula = mpg ~ cyl * wt, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2288 -1.3495 -0.5042  1.4647  5.2344
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  54.3068     6.1275   8.863 1.29e-09 ***
## cyl          -3.8032     1.0050  -3.784 0.000747 ***
## wt           -8.6556     2.3201  -3.731 0.000861 ***
## cyl:wt        0.8084     0.3273   2.470 0.019882 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.368 on 28 degrees of freedom
## Multiple R-squared:  0.8606, Adjusted R-squared:  0.8457
## F-statistic: 57.62 on 3 and 28 DF,  p-value: 4.231e-12
```

Adding an interaction term for vehicle weight and vehicle cylinders increased the estimated coefficient size of both the cylinder and weight terms (in a negative direction) to -3.8 and -8.66, respectively, while also shifting the estimated intercept to 54.31.

The theoretical implication of adding the interaction term is that the effects of a vehicle's weight and a vehicle's number of cylinders on its MPG vary in relation to one another. In other words, a vehicle's weight is estimated to not only have a direct effect on its miles per gallon, but also an effect on the effect size of the vehicle's number of cylinders, thereby indirectly affecting the vehicle's miles per gallon.

Non-linear Regression

1. Using the wage_data file, answer the following questions:

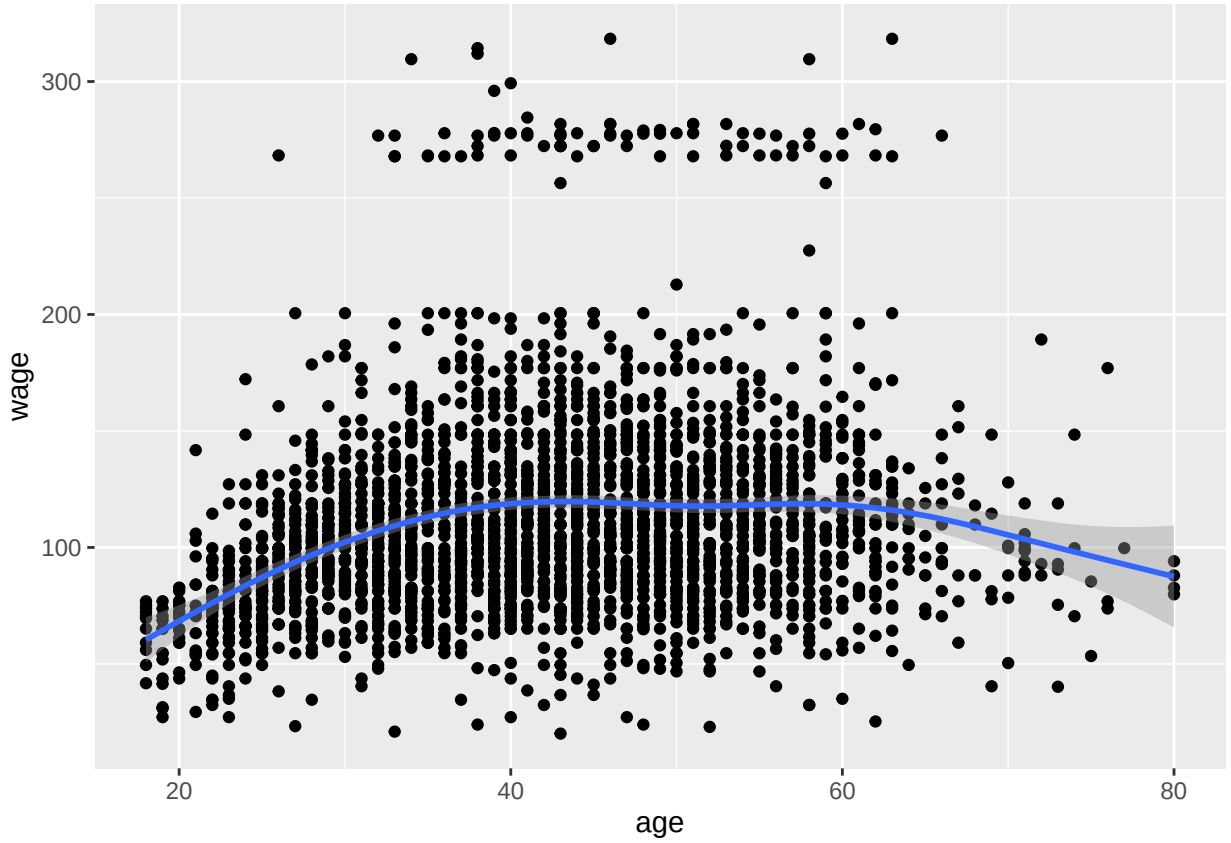
1a. Fit a polynomial regression, predicting wage as a function of a second order polynomial for age. Report the results and discuss the output (hint: there are many ways to fit polynomials in R, e.g., I, ^, poly(), etc.).

```
wage_regress <- wage_data %>% lm(formula = wage ~ age + I(age^2))
```

1b. Plot the function with 95% confidence interval bounds.

```
wage_plot <- wage_data %>% ggplot(aes(age,wage))+
  geom_point()+
  stat_smooth(formula = wage_regress$formula, level = .95)
wage_plot
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



1c. Describe the output. What do you see substantively? What are we asserting by fitting a polynomial regression?

The polynomial regression model appears to fit the data well. The scatterplot points are evenly clustered along a slightly concave slope, and the regression model fits that slope and runs through the middle of the points. By fitting a polynomial regression model to the data, we are asserting that an individual's wage has

a systematic, non-linear relationship to an individual's age, which predicts that wage will increase until one's forties, plateau until their sixties, and then decrease.

1d. How does a polynomial regression differ both statistically and substantively from a linear regression (feel free to also generalize to discuss broad differences between non-linear and linear regression)?

A polynomial regression differs statistically from a linear regression because the polynomial does not show a linearly proportional relationship between unit changes in independent variables and a dependent variable. Polynomial regression models the dependent variable 'y' in terms of an Nth degree polynomial in 'x', and are plotted as curves.

In linear regressions, the model's individual parameters can be understood in terms effect on the outcome variable, both in direction and magnitude. In polynomial and other non-linear regression forms, the estimated co-efficients are difficult to interpret individually, and it is more informative to analyze the function's global shape, direction, and fit in relation to the data.