Problem Set 1: Learning and Regression

Jinglin Yang

Jan 2020

Statistical and Machine Learning

Many learning problems fall naturally into one of the two categories: supervised learning or unsupervised learning.

In the supervised learning, for each observation of the predictor feature measurement(s) $X_i, i = 1, 2, 3 \ldots$, there is an associated response measurement, Y_i. And our goal is to fit a model that relates the response measurement to the predictor feature measurements, with the aim of providing the (maximally) accurate predictions of the response for future observations. The predictions are based on the training set $(x_1, y_1), (x_2, y_2), \ldots(x_N, y_N)$. The training set is the data which is already tagged with the correct answer. In supervised learning, we train the machine using the training sample, which is well "labeled". The process is also usually called "learning with a teacher". Specifically, we will make assumptions about the true data generating process, usually represented by a model between the response and the predictors. These models can be parametric and also nonparametric. Parametric models include linear regression, nonlinear regression, etc. And nonparametric models include KNN, decision trees, etc. For example, if we assume the model as $Y = f(X) + \epsilon$, then in the supervised learning, we attempt to learn $f(x)$. And the learning process can be broken into the following steps. At first, we need to gather data on both the response and the predictors and assemble a training set of observations $(x_1, y_1), (x_2, y_2), \ldots(x_N, y_N)$. Then we put these observed observations into a learning algorithm (usually a computer program), which produces outputs $f(\hat{xi})$ in response to the predictors. The learning algorithm has the property that it can modify its relationship $f(X)$ in response to diffrences $y_i - f(\hat{xi})$ between the original and predicted responses. At last, we evaluate the model's performance. And we often use the closeness between the original and predicted responses as the measure of success in supervised learning. After the learning process, we hope that the predicted and real response measurement will be close enough so that we could use the model to predict the response for future predictors likely to be encountered in practice. The typical examples of supervised learning are regression and classification.

In contrast, in the unsupervised learning, for each observation of the predictor feature measurements X_i, there is no associated response Y_i. Therefore, it is not suitable to fit a regression or classification model any more, since there is no response measurement to predict now. The target of unsupervised learning is to better understand the hidden patterns or underlying structure among the set of the predictor feature measurements. Also, unlike in the supervised learning, there is a clear measure of success, it is more challenging to evaluate the models because the goal of unsupervised learning is more complicated. It is not as simple as accurate predictions. For example, in the clustering, our task is to put a set of objects into different groups. Therefore, we need to find a way to that objects in the same cluster are more similar (in some sense) to each other than to those in other clusters. And clustering actually is a multi-objective optimization problem, because there are different criteria when we are evaluating the algorithms. And we need to tailor the algorithm and parameter settings for different data sets and different intended use of the results. Another application is in the density function. Unlike in the supervised learning, we are estimating the conditional probability density $p(Y|X)$, in the unsupervised learning, we are estimating the prior density $p(X)$.

Linear Regression

Using the *mtcars* dataset in R,

```
> attach(mtcars)
> names(mtcars)
```

(a) Predict miles per gallon (mpg) as a function of cylinders (cyl).

```
> lm.fit1 = lm(mpg~cyl)
> summary(lm.fit1)

Call:
lm(formula = mpg ~ cyl)

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
> coef(lm.fit1)
(Intercept)         cyl 
   37.88458    -2.87579 
```

The output is shown above. In this model, the intercept is 37.8845765 and the slope is -2.8757901.

(b) Write the statistical form of the simple model in the previous question.

The population regression function is $mpg = \beta_0 + \beta_1 \times cyl + \epsilon$.

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

```
> lm.fit2 = lm(mpg~cyl+wt)
> summary(lm.fit2)

Call:
lm(formula = mpg ~ cyl + wt)

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
> coef(lm.fit2)
(Intercept)         cyl          wt
  39.686261   -1.507795   -3.190972
```

The results are shown above. We know that after adding vehicle weight to the specification, the intercept is larger, and the size of the coefficient of **cyl** is smaller. Also the significance level of the coefficient of **cyl** is lower after adding **wt**. The coefficient of **wt** is highly significant. And the R^2 is larger after adding **wt**.

(d) 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?

```
> lm.fit3 = lm(mpg~cyl*wt)
> summary(lm.fit3)

Call:
lm(formula = mpg ~ cyl * wt)

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
```

After including a multiplicative interaction term in the function, the coefficients of **cyl** and **wt** are still negative and significant. But the significance level of the coefficient of **cyl** is higher after adding **cyl**·**wt**. And the R^2 is large.

By including a multiplicative interaction term, we relax the additive assumption: the effect of the change in the response **mpg** due to a one-unit change in **cyl**(or **wt**) is constant, regardless of the value of **wt**(or **cyl**). It means that we theoretically assert that adjusting **cyl**(or **wt**) will change the impact of **wt**(or **cyl**) on **mpg**.

Nonlinear Regression

Using the *wage_data* file, answer the following questions:

```
> wage_data<- read.csv("C:/Users/lenovo/Desktop/ps1/wage_data.csv",header=T)
> attach(wage_data)
```

(a) Fit a polynomial regression, predict *wage* as a function of a second order polynomial for *age*. Report the results and discuss the output)

```
> lm.fit4 = lm(wage~age+I(age^2))
> summary(lm.fit4)

Call:
lm(formula = wage ~ age + I(age^2))

Residuals:
    Min      1Q  Median      3Q     Max 
-99.126 -24.309  -5.017  15.494 205.621 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -10.425224   8.189780  -1.273    0.203    
age           5.294030   0.388689  13.620   <2e-16 ***
I(age^2)     -0.053005   0.004432 -11.960   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 39.99 on 2997 degrees of freedom
Multiple R-squared:  0.08209,	Adjusted R-squared:  0.08147 
F-statistic:   134 on 2 and 2997 DF,  p-value: < 2.2e-16
```

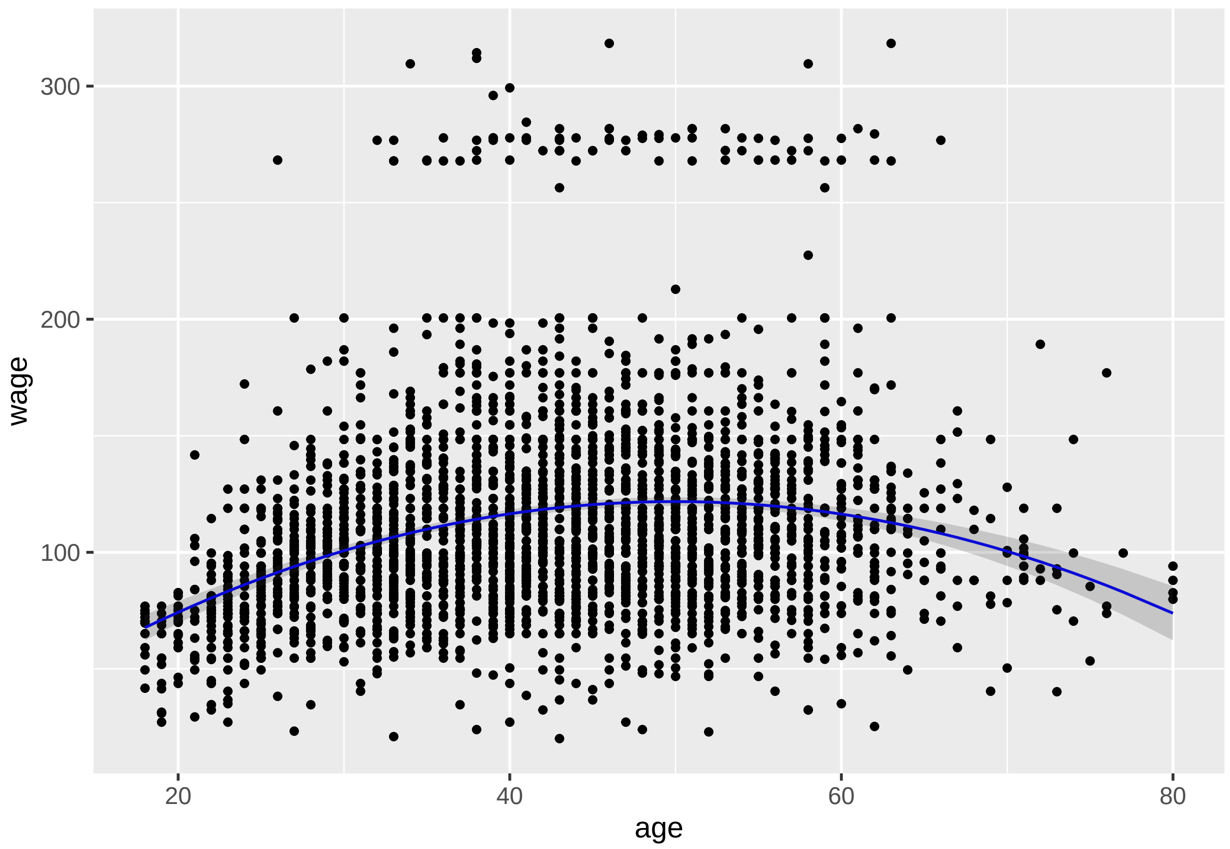
The results are shown above. From the output, we know that the intercept is -10.4252243, the coefficient of age is 5.29403, and the coefficient of age^2 is -0.0530051. And all of the three estimated parameters are statistically significant at any level. Therefore, it implies that there exists a non-linear relationship between wage and age.

(b) Plot the function with 95% confidence interval bounds

```
> CIs = cbind(age,wage, predict(lm.fit4,data.frame(age),interval = "confidence",level=0.95))
> library(ggplot2)

Attaching package: 'ggplot2'
The following object is masked from 'mtcars':

    mpg
> ggplot(data=data.frame(CIs),mapping=aes(age)) +
+   geom_point(aes(y=wage),size=1) +
+   geom_line(aes(y=fit), colour="blue") +
+   geom_ribbon(aes(ymin=lwr, ymax=upr), alpha=0.2) +
+   labs(y="wage")
```



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

From the graph, we can see that the points seem to have a quadratic shape. And the output shows that, as the age increases, the wage would at first increase, and then, when the age increases to around 50, the wage begins to decrease. And the confidence interval band is pretty thin along the line, and larger at two tails. This is because the data at two tails are much sparser, which lower the accuracy of our estimation.

And by fitting a polynomial regression, we assert that there is a quadratic relationship between wage and age.

(d) How does a polynomial regression differ both *statistically* and *substantively* from a linear regression?

```
> lm.fit5 = lm(wage~age)
> summary(lm.fit5)

Call:
lm(formula = wage ~ age)

Residuals:
     Min       1Q   Median       3Q      Max
-100.265  -25.115   -6.063   16.601  205.748

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 81.70474    2.84624   28.71   <2e-16 ***
```

```
age          0.70728    0.06475   10.92   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 40.93 on 2998 degrees of freedom
Multiple R-squared:  0.03827,   Adjusted R-squared:  0.03795
F-statistic: 119.3 on 1 and 2998 DF,  p-value: < 2.2e-16
```

From the results above, by comparing the second order polynomial regression model with the linear regression model, we know that,

- Statistically, although the coefficients in both models are significant, the quadratic fit appears to be better than the fit obtained when just the linear term is included, because the R^2 of the quadratic fit is 0.082 and the adjusted R^2 is 0.081, which are both larger than 0.038 and 0.038 for the linear fit.
- Substantively, a linear relationship between wage and age seems unrealistic. Because as for young people, it is likely that their wage will increase due to increased working proficiency and maybe the job-hopping. And as people reach at a certain age, they will become less productive. Especially their wage will substantially decrease after they retire. Therefore, a quadratic relationship between wage and age seems more realistic.