UCLA CS97 Homework Assignment 2 Part 1

Yizhou Sun (yzsun@cs.ucla.edu)

July 18, 2021

0 Instruction and Preparation

Due date: Tuesday, 7/20 at midnight (11:59pm PT) Instructions: Be sure to clearly label where each problem and sub-problem begins. All problems must be submitted in order.

Goal: This homework aims to help you go through main concepts covered so far with toy examples.

Notations. A list of notations that might be new to you are provided below. (Skip this part if you already know them)

1. Summation Notation: $\sum$. When summing multiple items together, this notation can help us shorten the formula. For example, $x_1 + x_2 + x_3 + x_4 + x_5 + x_6$ can be written as $\sum_{i=1}^{6} x_i$, where i is the index for ith item.
2. Product Notation: $\prod$. Similar to summation notation, product notation is to help us to shorten the formula when *multiplying* multiple items together. For example, $x_1 \times x_2 \times x_3 \times x_4 \times x_5 \times x_6$ can be written as $\prod_{i=1}^{6} x_i$.

1 (20pt) Know your data

area (100 sq.ft.)	bedroom	zipcode	price ($100K)
15	2	11111	5
20	3	22222	7
18	3	11111	6
16	3	33333	5.5

Table 1.1: House Price Training Dataset

1.1 Mean

The **mean** of a set of n observations of a variable is denoted $\bar{x}$ and is defined as:

$$\bar{x} = \frac{x_1 + x_2 + x_3 + \cdots + x_n}{n} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1}$$

(4pt) Exercise : Compute mean for *price* in Table 1.1.

$\bar{x} = \left(\frac{5+7+6+5.5}{4}\right) = 5.875$ units

1.2 Variance and Standard Deviation

The (sample) **variance** of a set of n observations of a variable is denoted as s^2 and is defined as:

$$s^2 = \frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + \cdots + (x_n - \bar{x}^2)^2}{n-1} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 \tag{2}$$

(4pt) Exercise : Compute variance for *price* in Table 1.1.

$s^2 = \frac{1}{n-1}\sum_{i=1}^{4}(x_i - \bar{x})^2 = 0.729$

The **standard deviation** (std) is the square root of variance, denoted as s.

(2pt) Exercise : Compute standard deviation for *price* in Table 1.1.

$s = \sqrt{s^2} = 0.854$

1.3 Normalization

Normalization is to transform a numerical variable by re-centering and scaling. One of the most popular normalization is z-score normalization, which is also called standardization. For each observation x_i of variable X, it will be transformed by subtracting from mean and dividing by standard deviation:

$$x_i^{(new)} = \frac{x_i - \bar{x}}{s} \tag{3}$$

This process has been seen in our Homework 1 without disclosing the details.

(4pt) Exercise: Normalize *area* in Table 1.1 via z-score normalization.

$\bar{x} = \frac{1}{n}\sum_{i=1}^{4} x_i = 17.75$

$s^2 = \frac{1}{n-1}\sum_{i=1}^{4}(x_i - \bar{x}) = 4.917$

$s = \sqrt{s^2} = 2.217$

$x_i^{(new)} = \frac{x_i - \bar{x}}{s}$

$x_1 = -1.015 \quad x_3 = 0.338$

$x_2 = 1.240 \quad x_4 = -0.564$

1.4 One-hot encoding

In order to handle categorical features, we need to create dummy variables to convert discrete values into numerical ones. A standard way of doing so is to create a binary dummy variable for each possible value for that variable (e.g., in sklearn they adopt this way).

(4pt) Exercise: Write down the one-hot encoding for every data point for variable *zipcode* in Table 1.1. Hint: Fill in the table below.

area (100 sq.ft.)	bedroom	is_11111	is_22222	is_33333	price ($100K)
15	2	1	0	0	5
20	3	0	1	0	7
18	3	1	0	0	6
16	3	0	0	1	5.5

Table 1.2: House Price Training Dataset with One-hot Encoding

1.5 The relationship between two variables

Correlation between two variables is important to decide the relationship between two variables. Right now, you are not required to know the definition of correlation. Roughly, given two variables, X and Y, if Y increases when X increases, they are positively correlated. If Y decreases when X increases, they are negatively correlated. If no obvious trend between them, they are not correlated. Scatter plot gives us a good understanding of such correlation.

(2pt) Exercise: Are *area* and *price* correlated in Table 1.1? If yes, are they positively correlated or negatively correlated?

Yes; they are positively correlated because as the area increases, the price also increases.

2 (45pt) Linear Regression

For a regression/prediction task, the goal is to use predictors/features to predict numerical response/target variable. Linear regression is one type of model that can help us solve this problem.

There are three important components:

1. Write down the form of model.
2. Estimate the parameters (unknown variables) using training dataset.
3. Evaluate the model using test dataset.

Now we use the regression problem of predicting house price (Table 1.1) as an example to illustrate.

2.1 Simple Linear Regression

A simple linear regression contains a only one predictor. Let's say we choose *area* as the predictor.

2.1.1 The form of simple linear regression

In general, if Y is something we want to predict, and X is the predictor, the form of simple linear regression is

$$Y = \beta_0 + \beta_1 \times X \tag{4}$$

In our case, it is:

$$price = \beta_0 + \beta_1 \times area \tag{5}$$

What does it mean? It means that if we increase area by 1 unit (in our case, 1 unit = 100 sq.ft.), the price will increase by β_1 units (in our case, 1 unit = \$100K). If the area (meaning living area) is 0, the price would be β_0, which is the price for a land lot. If we plot all those $(area, price)$, we will see they form a straight line with slope β_1 and intercept β_0.

(5pt) Exercise: Given $\beta_0 = 1$ and $\beta_1 = 3$, what would be our price prediction for a house with 1500 sq.ft. (i.e., area = 15)?

price = $\beta_0 + \beta_1$(area) = 1 + 3(15) = 46 units

2.1.2 Estimating parameters using training dataset

If we know β_0 and β_1, we can predict price for any house given their living areas. Unfortunately we do not know them, which are called parameters. How to get them? We can compute them using some observed data. Just remember the dataset we have observed is just one realization among all possible datasets. Thus, those parameters will be different when we are given different datasets. So we use the terminology "estimate" to reflect such uncertainty. Anyhow, it is just a terminology, in practice, we do calculation to estimate.

For the estimation task, we need to address two questions.

1. Given a specific β_0 and β_1, how do we know it is good or bad? We calculate a *loss* based on the given dataset. Note here, the dataset is given, the loss will change along with β_0 and β_1, therefore we call it a loss function. It is ok if you are not comfortable with the terminology of function. Just keep in mind, for different β's, we have different losses. This mathematical function, however, is very similar to the function we used in programming, both of which transform some input to some output.
2. Optimization: Given the loss function, we want to find the best β_0 and β_1 such that the loss is the smallest. We call it as minimize the loss function, and it is ok if you are not comfortable with the terminology minimization. In class, we introduced closed form solution and gradient descent algorithm for optimization. They can be treated as blackbox, and you don't need to know how to do optimization, as python libraries already provide the solution.

Now let's introduce some key concepts for estimation and do some calculation.

price = $\beta_0 + \beta_1$ (area)

- residual = observed value - predicted value.
- Mean Squared Error (MSE) = sum over squared residuals for all the data points.

(15pt) Exercise: Suppose we have two possible lines to fit the data in Table 1.1. For the first line, $\beta_0 = 1$ and $\beta_1 = 3$; and for the second line, $\beta_0 = 2$ and $\beta_1 = 2$. Please fill in the following table (Table 2.1). What is the MSE for each line? Which line will you choose and why?

area

price	predicted price for line 1	residual for line 1	predicted price for line 2	residual for line 2
15	1+3(15) = 46	5-46 = -41	2+2(15) = 32	5-32 = -27
20	1+3(20) = 61	7-61 = -54	2+2(20) = 42	7-42 = -35
18	1+3(18) = 55	6-55 = -49	2+2(18) = 38	6-38 = -32
16	1+3(16) = 49	5.5-49 = -43.5	2+2(16) = 34	5.5-34 = -28.5

Table 2.1: House Price Training Residual Calculation

* see code notebook for work

line 1 MSE = 3479.0625

line 2 MSE = 1347.0

I would choose line 2 because it has the more minimized loss function.

After we know how to decide whether a line is a good fit or not, we can use optimization to find the best line.

(10pt) Exercise: Find the best β_0 and β_1. You can use any approach you like, for example, calculating by hand using closed form formula provided in lecture slides or coding (e.g., code from Homework 1 Part 2). Calculate MSE for this best-fit line. Is it indeed smaller than Line 1 and 2?

* see code notebook for work

β_0 = -0.7033

β_1 = 0.3814

MSE = 0.0106

The MSE is much smaller than both Line 1 and Line 2.

2.1.3 Model Evaluation

Whether a model is good or not is measured based on test dataset, rather than the training dataset. There are several evaluation metrics, including MSE, MAE, and R^2.

area (100 sq.ft.)	price ($100K)
25	8
12	4

Table 2.2: House Price Test Dataset

(10pt) Exercise: Based on the above test dataset, evaluate your model derived from previous exercise using MSE. Note, this is called test MSE, in contrast to train MSE in the previous exercise. Please use the following table to help your calculation.

area (100 sq.ft.)	price ($100K)	predicted price	residual
25	8	8.83	-0.83
12	4	3.87	0.13

Table 2.3: House Price Test Dataset Residual Calculation

* see code notebook for work

MSE = 0.3530

2.2 Multiple Linear Regression

In many cases, multiple predictors might contribute to the prediction task, and we want to include them into the linear regression model. In general, if we have p predictors (in lecture slides, we used J), Say $X_1, X_2, \cdots, X_p$, the form of multiple linear regression is:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p = \beta_0 + \sum_{j=1}^{p} \beta_j X_j \quad (6)$$

In our house price prediction case, let's include both *area* and *bedroom* for the prediction task, and our model becomes:

$$price = \beta_0 + \beta_1 \times area + \beta_2 \times bedroom \tag{7}$$

(5pt) Exercise: Given $\beta_0 = 1$, $\beta_1 = 3$, and $\beta_2 = 2$, what would be our price prediction for a house with 1500 sq.ft. (i.e., area = 15)?

price = $1 + 3(15) + 2(2) = 50$ units

The parameter estimation and model evaluation follows similar procedure as simple linear regression.

Polynomial regression follows similar procedure as multiple linear regression, as long as we prepare new features by feature augmentation properly.

3 (35pt + 10pt bonus) Logistic Regression

Now let's consider a different task, called classification. For a classification task, the goal is to use predictors/features to predict categorical response/target variable. Logistic regression is one type of model that can help us solve this problem.

Similarly, there are three important components in logistic regression:

1. Write down the form of model.
2. Estimate the parameters (unknown variables) using training dataset.
3. Evaluate the model using test dataset.

Now we use a credit card transaction setting to illustrate, where a small and oversimplified dataset can be found in Table 3.2. We use simple logistic regression with a binary response variable as an example. Similar to linear regression case, multiple logistic regression and polynomial logistic regression will follow a very simple procedure to the simple logistic regression case.

amount ($100)	fraud
0.1	0 (No)
1	0 (No)
10	1 (Yes)
5	1 (Yes)

Table 3.1: Credit Card Transaction Dataset

3.1 Binary Responses (Binary Classification)

In binary classification, the key problem is to predict the probability of each possible outcome. For simplification, we use 1 and 0 to denote the outcome "Yes" and "No", respectively.

3.1.1 The form of simple logistic regression

In general, the prediction for the probability of $Y = 1$, denoted as $P(Y = 1)$, using one predictor X can be written as:

$$P(Y = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X)}} \tag{8}$$

where e is the natural number.

In our transaction classification case, it can be written as:

$$P(fraud = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 \times amount)}} \tag{9}$$

(5pt) Exercise: Given $\beta_0 = -2$ and $\beta_1 = 1$, what is the probability that a transaction with amount \$100 (i.e., *amount* = 1) is a fraud?

$P(fraud = 1) = \frac{1}{1 + e^{-(-2+1\times 1)}} = 0.269$

3.1.2 Estimating parameters using training dataset

Likelihood of the parameters given the dataset, i.e., the product of probabilities of observing the outcome for each data point, needs to be maximized to get the best β_0 and β_1. More concretely, if we denote the probability of the ith data point to have an outcome 1 as $p_i = P(y_i = 1)$, the probability of the ith data point becomes $L_i = p_i$ if $y_i = 1$, and $L_i = (1 - p_i)$ if $y_i = 0$. For example, if $y_i = 1$ and $p_i = 0.8$, then $L_i = 0.8$. If $y_i = 0$ and $p_i = 0.8$, then $L_i = 0.2$. Then the likelihood of the whole dataset with n observations is:

$$L_1 \times L_2 \times \cdots \times L_n = \prod_{i=1}^{n} L_i \tag{10}$$

(15pt) Exercise: Compute the likelihood of observing the dataset given $\beta_0 = -2$ and $\beta_1 = 1$, with the help of the following table. $P(\text{fraud} = 1) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 \text{amount})}}$

amount ($100)	fraud	$p_i = P(fraud = 1)$	The probability L_i
0.1	0 (No)	0.1301	0.8699
1	0 (No)	0.2678	0.7322
10	1 (Yes)	0.9997	0.9997
5	1 (Yes)	0.9526	0.9526

Table 3.2: Credit Card Transaction Dataset Likelihood Computation

$\prod_{i=1}^{4} L_i = (0.8699)(0.7322)(0.9997)(0.9526) = 0.6066$

3.1.3 Decision Boundary

Decision boundary is a set of points where $P(Y = 1) = P(Y = 0) = 0.5$. In other words, those points are the boundary of the two classes. The points in two sides of the boundary go to different classes. As we know that, for logistic function when the input is 0, the output is 0.5. In logistic regression case, that is when $\beta_0 + \beta_1 X = 0$. We can solve the equation and derive that $X = -\beta_0/\beta_1$ is the decision boundary for simple logistic regression, which is just one single point.

(10pt) Exercise: If the learned logistic model has $\beta_0 = -2$ and $\beta_1 = 1$, what is the decision boundary? Can you describe the meaning of this boundary in the transaction classification example?

$X = \frac{-(\beta_0)}{\beta_1} = \frac{-(-2)}{1} = 2$

This boundary means that data points above $200 will be classified as a fraud, and below $200 will not.

3.1.4 Regularization for Logistic Regression

Regularization is extremely important for logistic regression.

(10pt) Bonus Exercise:

1. Consider another set of coefficients, $\beta_0 = -4$ and $\beta_1 = 2$. Does this change the decision boundary? Does it change the likelihood? Compared to our old β coefficients, which one do you want to choose?

   $X = \frac{-(\beta_0)}{\beta_1} = \frac{-(-4)}{2} = 2$

   $\prod_{i=1}^{4} L_i = 0.8594$

   The decision boundary remains the same but the likelihood increased. I would choose the new set because its likelihood is higher.

2. We can see that by increasing β_0 and β_1 proportionally (you can try another set of coefficients, e.g., $\beta_0 = -6$ and $\beta_1 = 3$), the decision boundary will not change, but the likelihood is keep increasing. Which β_0 and β_1 should we use then? Is that acceptable? How regularization can be used to handle this issue?

   $X = \frac{-(\beta_0)}{\beta_1} = \frac{-(-6)}{3} = 2$

   $\prod_{i=1}^{4} L_i = 0.9493$

   We should use the new set since the original set has a lower likelihood. However, the new set is not acceptable because it overfits the data. Regularization can be used to prevent overfitting while maintaing a high likelihood, thus achieving an effective balance.

3.2 Multiple Responses (Multi-Class Classification)

In many cases, we might have multiple responses. For example, we may want to classify the risk credit card transaction into three levels: {no-risk, medium-risk, high-risk}, instead of two. When there are multiple responses, we need to model the probability for each possible outcome. To understand the modeling process in optional. When the model is learned via a training dataset, we can use the model to classify each data point into the class with the biggest probability.

(5pt) Exercise: Consider a credit card transaction risk classification task, which has three levels: {no-risk, medium-risk, high-risk}. If the prediction probability of a transaction for each class is $0.6, 0.3, 0.1$, respectively, which risk level do you want to classify this transaction into?

I want to classify this transaction into the no-risk level. since the model is learned via a training dataset as suggested above, the data point is classified into the class with the biggest probability. In this case, the highest probability is 0.6, for the no-risk level, hence why the transaction should be classified as no-risk.