CIS 7000-005: Debugging Data & Models Fall 2022

Linear Models — August 30

Prof. Eric Wong

The following themes will reoccur throughout the course:

- What are the assumptions we are making?
- What can methods tell us, and what do they not?
- Is the information usable or actionable for debugging?

1 Prelude - what is debugging?

First of all, what is a bug in ML? A bug is an *error* in computing that causes an incorrect or *unexpected* result. Suppose we have a cat vs dog classifier, what kinds of errors could occur?

- One-off mistakes: user error, freak accident. Is it a bug if a cat is wearing a collar labeled with the word dog?
- Inherent noise: obscured inputs, partial information, unclear label. If you see a four legged critter partially hidden behind a wall, there is a non-zero chance that this could actually be a dog.
- Systematic issues: Does it affect a population? You could say animals over 20lbs are probably not cats, and you would be correct most of the time. But there are some heavy breeds, like Maine Coons and Siberian Cats.

These are various kinds of errors that can occur. But what is the expected outcome that we'd like to have instead? This is not always clear-cut, and also depends on your setting. The notion of correctness or desired behavior is not always be the same everywhere. In some ML-adjacent settings, these are some notions of correctness that we might expect our models to follow:

- Supervised - predicting the right label
- Unsupervised - predicting similar results for similar inputs
- Explainable - predicting the right result for the right reason
- Fairness - predicting results that are not unfavorable to certain groups
- Legal - predicting results that adhere to existing laws

Loosely speaking, debugging is thus the act of finding these errors and correcting them to obtain an expected result.

2 Linear models

Let us consider one of the simplest machine learning models: the linear model. How can we debug in this setting?

$$f(x) = x^T \beta \tag{1}$$

We can derive the least squares estimator by setting the derivative to zero to get the following estimate for β (written in matrix form):

$$\begin{aligned} \hat{\beta} &= \min_{\beta} \|Y - X\beta\|_2^2 \\ &= (X^T X)^{-1} X^T Y \end{aligned} \tag{2}$$

Typically, in linear regression we make certain assumption including the following:

1. Linearity: $Y = X\beta + \epsilon$ where ϵ is a noise variable
2. Gaussian noise: $\epsilon \sim MVN(0, \sigma^2 I)$

3 Hypothesis testing for linear regression

Linearity is often praised as a gold standard for being understandable. After all, interpretion of the linear coefficients is a core technique taught in statistics courses, with stringent assumptions and conclusions. Hypothesis testing allows us to conclude whether the linear trends that we find are statistically significant, as follows:

1. For each variable x_i of the linear model, we can compute a p-value that tests the null hypothesis that the predicted variable has no correlation with the dependent variable, $H_0 : \beta_i = 0$.
2. If this p-value is below a significance level, then there is enough evidence to reject the null hypothesis.
3. If this p-value is above the significance level, then there is insufficient evidence to conclude that a non-zero correlation exists.

In theory, this allows us to check the significance of the correlations identified in the linear model. But how does this work exactly? Under the linearity assumption, we have

$$\hat{\beta} = (X^T X)^{-1} X^T Y = (X^T X)^{-1} X^T (X\beta + \epsilon) = \beta + (X^T X)^{-1} X^T \epsilon \tag{3}$$

Combined with the Gaussian assumption, we have

$$\hat{\beta} \sim MVN(\beta, \sigma^2 (X^T X)^{-1}) \tag{4}$$

so $\frac{\hat{\beta}_i - \beta_i}{\sigma \sqrt{(X^T X)^{-1}_{ii}}} \sim N(0,1)$. Combined with the fact that $\frac{(n-p)s^2}{\sigma^2} \sim \chi^2_{n-p}$, we have

$$\frac{\hat{\beta}_i - \beta_i}{\hat{\sigma} \sqrt{(X^T X)^{-1}_i i}} \sim t_{n-p} \tag{5}$$

This allows us to compute a p-value based on the t-distribution. Here is an example output of the `statsmodels` package on a heart disease data set that we will use later:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.463
Model:                            OLS   Adj. R-squared:                  0.431
Method:                 Least Squares   F-statistic:                     14.70
Date:                Sun, 28 Aug 2022   Prob (F-statistic):           6.47e-27
Time:                        13:36:59   Log-Likelihood:                -299.77
No. Observations:                 272   AIC:                             631.5
Df Residuals:                     256   BIC:                             689.2
Df Model:                          15
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
x1             0.0062      0.006      1.024      0.307      -0.006       0.018
x2             0.4912      0.102      4.794      0.000       0.289       0.693
x3             0.0050      0.003      1.738      0.083      -0.001       0.011
x4             0.0014      0.001      1.441      0.151      -0.001       0.003
x5             0.0483      0.136      0.355      0.723      -0.220       0.316
x6            -0.0041      0.003     -1.551      0.122      -0.009       0.001
x7             0.2172      0.116      1.877      0.062      -0.011       0.445
x8             0.1699      0.051      3.330      0.001       0.069       0.270
x9            -0.7354      0.247     -2.978      0.003      -1.222      -0.249
x10           -0.4669      0.207     -2.259      0.025      -0.874      -0.060
x11           -0.5752      0.195     -2.956      0.003      -0.958      -0.192
x12            0.0009      0.192      0.005      0.996      -0.378       0.379
x13           -0.6239      0.270     -2.313      0.022      -1.155      -0.093
x14           -0.6036      0.485     -1.246      0.214      -1.558       0.351
x15           -0.5491      0.280     -1.959      0.051      -1.101       0.003
x16           -0.6780      0.268     -2.528      0.012      -1.206      -0.150
x17           -0.3442      0.244     -1.410      0.160      -0.825       0.137
x18           -0.7545      0.289     -2.610      0.010      -1.324      -0.185
==============================================================================
Omnibus:                        2.553   Durbin-Watson:                   1.852
Prob(Omnibus):                  0.279   Jarque-Bera (JB):                2.626
Skew:                           0.217   Prob(JB):                        0.269
Kurtosis:                       2.790   Cond. No.                     4.68e+18
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.31e-30. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

Review:

- What are the assumptions we are making? Hypothesis testing assumes linearity with Gaussian noise for the underlying ground truth. These are typically very strong assumptions, and so the p-values may not even be meaningful in many cases.
- What can methods tell us, and what do they not? Hypothesis testing can tell us whether a coefficient is statistically different from zero. If the assumptions are true, then this is a statistically rigorous way to do so. It cannot tell us exactly what the coefficient is. It also does not explain any individual prediction.
- Is the information usable or actionable for debugging? Possibly. Significance testing could be used to trim or delete variables, in the hopes that this improves the model. However, there is no guarantee that this does. Debugging with significance tests may be difficult to do for a more specific purpose, i.e. trying to correct mispredictions for individual examples or finding subpopulations.

4 Score-based explanations

Linear models are simple enough that we can also directly interpret each individual prediction. Specifically, for any prediction $f(x) = x^T\beta$, we can decompose the prediction over the individual features: $s_i(x) = x_i\beta_i$ for each i. This decomposition serves as an accurate explanation of the model's prediction:

1. There is no approximation; the scores are 100% faithful to the actual model.
2. Linear models (usually) don't have too many features to interpret. So the entirety of the explanation can be feasibly understood.

Once we leave linear models, we'll see later on that these two properties (faithfulness and feasibility) almost never hold.

One potential drawback here is that scores, although intuitive by construction, may be meaningless for certain features. For example, suppose for an example, we have scores

$$s_i(x) = \begin{cases} 1 & \text{if } i \text{ is odd} \\ -1 & \text{if } i \text{ is even} \end{cases}$$

We could have have an infinite number of these scores without changing the prediction. Furthermore, the individual scores themselves could be arbitrarily large. Since all the scores are perfectly correlated, any individual feature may not actually be meaningful for the model.

Review:

- What are the assumptions we are making? Although we are not making many assumptions here (the score is by construction of the linear model), in using these scores, we are implicitly assuming that the score is correlated with importance. This may not be the case if variables are highly correlated or redundant, leading to potentially large and meaningless scores!

- What can methods tell us, and what do they not? The scores tell us how much each variable contributed to the overall prediction on a per-instance basis. It does not tell us globally how the model makes predictions. This score may or may not be related to an actual importance of the feature. It also does not tell us any information about correlations or causal effects.
- Is the information usable or actionable for debugging? Possibly. Manual inspection of the scores for misclassified examples may shed light on why the linear model was incorrect. However, if there are too many scores, this may become humanly impossible.

5 Heart disease

Armed with these two toolkits, lets try to debug a linear model. For example, let's take a heart disease model and fit a simple linear model. We'll use the UCI data set from:

`https://archive.ics.uci.edu/ml/datasets/heart+disease`

Here, we'll fit a linear model with ordinary least squares and attempt to debug it. The code for this section is in the Jupyter notebook accompanying these notes. Our expectation is simple: we want the model to predict the correct answer (heart disease or no heart disease) according to the labels. The goal is to figure out why the model makes particular predictions and figure out why it was wrong on certain predictions.

Here, we debug a prediction by showing the score and significance of each feature. Each feature is also annotated with `*` if the score exceeds a threshold (> 0.2) and `p` if the feature used is statistically significant (at significance level 0.05). For example, this is the debugging output for a test prediction that correctly predicts the presence of heart disease:

```
Ground truth: y_test=1

age: 0.33*
sex: 0.49*p
trestbps: 0.55*
chol: 0.30*
fbs: 0.00
thalach: -0.44*
exang: 0.22*
oldpeak: 0.00p
cp_1.0: -0.00p
cp_2.0: -0.00p
cp_3.0: -0.00p
cp_4.0: 0.45*
restecg_0.0: -0.00p
restecg_1.0: -0.00
restecg_2.0: 0.04
slope_1.0: -0.00p
slope_2.0: 0.25*
```

```
slope_3.0: -0.00p

(bias) -1.63 + (w^Tx): 2.18 = (total) 0.56
Heart disease predicted
```

Here is the debugging output for a test datapoint that the model gets wrong:

```
Ground truth: y_test=1

age: 0.36*
sex: 0.49*p
trestbps: 0.50*
chol: 0.34*
fbs: 0.00
thalach: -0.64*
exang: 0.00
oldpeak: 0.02p
cp_1.0: -0.00p
cp_2.0: -0.00p
cp_3.0: -0.00p
cp_4.0: 0.45*
restecg_0.0: -0.03p
restecg_1.0: -0.00
restecg_2.0: 0.00
slope_1.0: -0.09p
slope_2.0: 0.00
slope_3.0: -0.00p

(bias) -1.63 + (w^Tx): 1.40 = (total) -0.23
No lung cancer predicted
```

Can you figure out why the model made this mistake? Does the significance test or the calculated scores help you in identifying where the model went wrong? You may be worried about several aspects:

1. There is very little overlap between variables that actually contribute towards a positive heart disease prediction (high score), and variables that are statistically significant (hypothesis testing)! So even in the linear setting, our debugging tools are already pointing at different parts.
2. Many of the scores for both of these predictions are very similar, and there doesn't seem to be any meaningful difference between scores from statistically significant features.
3. If you look carefully, it may be because the score for the `thalach` variable was slightly more negative than in the correct example, indicating that the maximum heart rate achieved may have had some influence on this misclassification.

There is no clear-cut answer here: debugging even a linear model is ultimately still pretty difficult!

For reference, these are the variable definitions in this dataset.

Variable	Definition
age	age in years
sex	sex (1 = male; 0 = female)
cp	cp: chest pain type
cp_1	Value 1: typical angina
cp_2	Value 2: atypical angina
cp_3	Value 3: non-anginal pain
cp_4	Value 4: asymptomatic
trestbps	esting blood pressure (in mm Hg on admission to the hospital)
chol	serum cholestoral in mg/dl
fbs	(fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)
restecg	resting electrocardiographic results
restecg_0	Value 0: normal
restecg_1	Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV)
restecg_2	Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria
thalach	maximum heart rate achieved
exang	exercise induced angina (1 = yes; 0 = no)
oldpeak	ST depression induced by exercise relative to rest
slope	the slope of the peak exercise ST segment
slope_1	upsloping
slope_2	flat
slope_3	downsloping

6 References

Parts of these notes are pulled from Cosma Shalizi's course on Modern Regression at

`https://www.stat.cmu.edu/~cshalizi/mreg/15/`