



MONASH
University

MONASH
BUSINESS
SCHOOL

notes

Shuofan Zhang

4/16/2018

Goal

Hello everyone, I am Shuofan, my research project is about human and computer. Our goal in this paper is to teach the computer to read residual plots.

Because the multiple regression model for cross-sectional data is still the most widely used vehicle for empirical analysis in economics and other social sciences". And detecting possible violations of the Gauss-Markov assumptions is crucial to interpret the data properly, especially in the early stage of analysis. There are several distribution tests that are commonly used, for instance, the Breusch-Pagan test and White test for investigating heteroskedasticity. But primarily residual plots are the main diagnostic tool for these problems (refer to slides), and these rely on human evaluation. It can be difficult for beginners to learn how to recognize patterns seen could arise by chance and that the model is proper.

Why plots?

The short answer is because plots show a lot more information than a single statistics. A good example here will be Matejka and Fitzmaurice's paper, where they generated this set of interesting plots. The original plot here is the dinosaur, and then they add tiny little bit of changes to this data and only keep the ones when the statistics are still the same, keep doing this, after different number of iterations, they get this set of different plots, but all of them have the same statistics up to two decimal places.

As Anscombe said in 1973, "... make both calculations and graphs. Both sorts of output should be studied; each will contribute to understanding."

Visual inference

So the graphs are very important, and it relies on human evaluation. How to utilize this information provided by graphs and more importantly, to make valid inference?

Wickham, Di, Hofmann and Buja have developed a protocol named lineup which is validated later on using human experiments in 2013.

This is an example of lineup. Which plot do you think is the most different? (3) We reject the null hypothesis that the data there is no relationship between the residuals and the fitted values.

Visual inference

Here we use this set of plots as our test statistics, and different type of plots can be used for different hypothesis, in this case, the null hypothesis is there is no relationship left in the residuals, while the alternative hypothesis is there are some kind of relationship.

More formally, “The protocol consists of generating 19 null plots (could be other number), inserting the plot of the real data in a random location among the null plots and asking the human viewer to single out one of the 20 plots as most different from the others”. If the real plot is chosen, it means the real data is different from the null hypothesis, so we reject the null hypothesis with 5% chance to be wrong (Type I error). (then same with slides)

Deep learning

Although human vision has been proved to be powerful and reliable, it is not realistic to render human experiments for every single hypothesis test. If we can train a computer to read residual plots we can have it process a lot more data, than a human can manage. How are we gonna teach computer to read residual plots for us? The answer is deep learning. - Computer vision has advanced substantially in recent years, you may all heard about self-driving cars, robotics - Convolutional neural networks is a type of deep-learning model almost universally used in computer vision applications. For simplicity, we can think of the deep learning neural network as a super complex nonlinear model which could estimate hundreds of thousands parameters with big enough dataset. As usual regression problem, to get the estimates of unknown parameters, we need to provide the model with dependent variable and independent variables. Since we wanna train the deep learning model to read residual plots, the independent variable will be the residual plots simulated from the null distribution and the alternative distribution, and dependent variable will be the labels of that plot indicating the true relationship of the original data. Once we have these estimated parameters, we then can use them to classify unseen residual plots, eg. to perform hypothesis

Aside: Volvo admits its self-driving cars are confused by kangaroos

However, computer has its own disadvantages, and is restricted by the presetting of training. As we can see in this case, the self-driving cars fails to detect the movement of kangaroos because their unique movement. Because the model was trained to detect movement based on the objects moving directions as left or right, forward and backwards but not up and down.

Aside: Computers can't tell difference between blueberry muffins and chihuahuas

This is another example of computers confused by muffins and chihuahuas. Can you tell which one is muffin and which one is dog?

So deep learning model does have pitfalls, disadvantages compared to human visions. And we wanna know how it will perform in reading residual plots compared to human.

Experiment

- 1 simulation (the first line seems pretty simply and straightforward, isn't it? In fact, this is where I spent most of my time in the last one month) Think about it, there are countless possibilities in the residual plots generated from a non-linear model. We will talk about this in more details later.
- 2 Fit a linear model to the data, extract standardized residuals and fitted values (we standardize the residuals and fitted value cause nothing will be lost by doing so, and it makes the plots more comparable with each other)
- 3 Save residual plots as fixed-sized images to our local drive (we set all plots as same fix-sized images, it makes the comparison more reasonable and also easier for deep learning model to process)
- 4 Train a deep learning classifier to recognise the departures from assumptions
- 5 Test the model's performance on new data and compute the accuracy

Data simulation

A lot of decisions have been made in the model design process, like sample sizes for each plot, the coefficients variances and a whole lot of other parameters. All details are discussed in the paper, here I just listed out several factors as an example.

- We generated all explanatory variables from $X \sim N(0, 3)$ instead of variance = 1, because we want x and y to have more variation.
- intercept $\beta_0 = 0$, because we will fit linear model to data, the intercept does not matter in the residual plots anymore, so we simply set it to be 0.
- Sample size: randomly generated between 20-1500. When sample size is smaller than 20, we can hardly see any systematic patterns. And 1500 is big enough for the data to show all kinds of relationship clearly.
- Image size: fixed 150x150, this size is large enough for us to see patterns, and not too cumbersome to be processed.

Type of relationship

Examples of residual plots of these four relationships.

Simchoni's analysis

There are similar studies have been done, for example, Simchoni has rendered an experiment in his blog.

- Simulate data from linear relationship with $\rho = -0.9$ to $\rho = 0.9$
- Separate into two groups: significant/insignificant by t-test
- Train deep learning model with these two groups
- Test the model on lineup
- Successful in detecting linear relationship but fail in non-linear

Comparison with human subject experiments

- Majumder et al (2013) conducted a large study to compare the performance of the lineup protocol, assessed by human evaluators, in comparison to the classical test.
- They have conducted three experiments in their paper, experiment 2 examined $H_o : \beta_k = 0$ vs $H_a : \beta_k \neq 0$ assessing the importance of including variable k in the linear model, t-test was also conducted in comparison with the human experiments.
- The experiment consists 70 lineups of size 20 plots
- 351 evaluations by human subjects have been done in that experiment
-
- Trained deep learning model will be used to classify plots from this study. Accuracy will be compared with results from human subjects.

Example lineup from experiment 2

This is an example of lineup from experiment 2.

Timeline

