ML and Numerical Software Development

Statistics-II

Organon Analytics

November 26, 2019

Statistics-II Agenda

- Point estimation, Interval estimation, confidence intervals
- Hypothesis Testing
- Multivariate Linear Regression: Specification, estimation, hypothesis testing, feature selection, goodness-of-fit, prediction

Estimation, Estimators, Point Estimation, Interval Estimation

- Assume DGP is parameterized, i.e. DGP $\sim p(X;\theta)$ where $p(\cdot)$ is density(or probability-mass) function
- DGP produces the data: $\{x_1, x_2, \cdots, x_N\}$
- θ: Population value
- $\hat{\theta}_N = g(x_1, x_2, \cdots x_N)$
- Remember: θ is fixed, but $\hat{\theta}_N$ is a random variable

Two important goodness criteria for the estimator $\hat{\theta}_N$

1. **Unbiased** $E[\hat{\theta}_N] = \theta \ \forall N$. Good but not enough! The expectation is true but what about its variance? The estimator might be unbiased but may have a lot of variance
2. **Consistency** $\hat{\theta}_N \underset{\longrightarrow}{P} \theta$: Hence, $\hat{\theta}_N$ becomes indistinguishable from θ as the sample size increases. But how fast?

Standard Error&Mean Square Error of a point estimator

Standard Error: Standard deviation of the estimator

$$\begin{aligned} \text{bias} &= \bar{\theta}_N = E[\hat{\theta}_N] \\ \textbf{s.e.} &= E[(\hat{\theta}_N - bias)^2] \end{aligned}$$

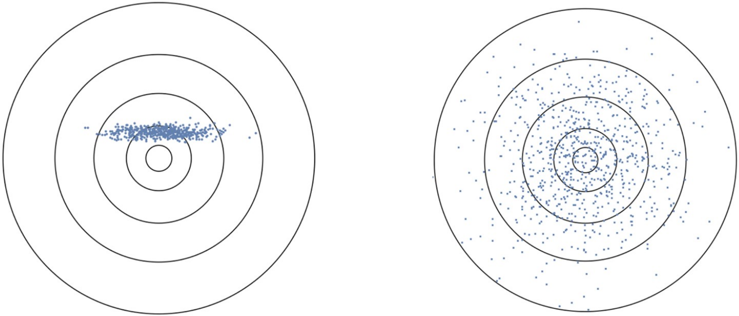
$$MSE(\hat{\theta}_N) = E[(\hat{\theta}_N - \theta)^2]$$

Note: Do not confuse the MSE with the variance of the estimator $\hat{\theta}_N$. They are different quantities.

Bias-Variance Trade-off in Statistical Estimation

$$\mathbf{MSE}(\hat{\theta}_N) = \text{bias}^2(\hat{\theta}_N) + \text{Var}(\hat{\theta}_N)$$

The truth is at the center; The dots are its estimates



Non-zero bias BUT low-variance ➔ LOW error

Zero bias BUT **high**-variance ➔ **HIGH** error

Confidence Intervals (Reliability of point estimates)

- A (scalar) estimator $\hat{\theta}_N$ is asymptotically Normal if

$$\frac{\hat{\theta}_N - \theta}{se} \to \mathcal{N}(0,1)$$

  Note: asymptotically means ”as N becomes large”

- Most estimators are asymptotically normal because of the **Central Limit Theorem**
- We can use this fact to derive interval estimates for θ starting from a point estimate $\hat{\theta}_N$
- A $(1-\alpha)$ *confidence interval* for a parameter θ is an interval (L, U) such that $P(\theta \in (L, U)) > (1-\alpha)$
- Note that L (lower) and U (for upper) are functions of the data and hence random variables as well

Confidence Intervals (Reliability of point estimates)

- α: significance level
- $(1-\alpha)$: confidence level
- Usually, $\alpha = 0.05$, and CI $= 95\%$
- If $\hat{\theta}_N$ is asymptotically-Normal (it usually is due to CLT), the CI for $\hat{\theta}_N$ is given as follows:

$$\begin{aligned}
C_n &= \left(\hat{\theta}_N - z_{\alpha/2}\hat{se}, \hat{\theta}_N + z_{\alpha/2}\hat{se}\right) \\
z_{\alpha/2} &= \Phi^{-1}(1-\alpha/2) \\
\Phi(x) &= \text{CDF for } \mathcal{N}(0,1) \\
P(\mathcal{N}(0,1) \in C_n) &= P(\mathcal{N}(0,1) <= z_{\alpha/2}) + P(\mathcal{N}(0,1) >= z_{\alpha/2}) \\
P(\mathcal{N}(0,1) \in C_n) &= \alpha/2 + \alpha/2 = \alpha
\end{aligned}$$

Confidence Intervals: A recipe

1. Calculate $\hat{\theta}_N$ from data
2. If population standard deviation σ is known, use the following table for the values $z_{\alpha/2}$

$(1-\alpha/2)$	$z_{\alpha/2}$
0.99	2.576
0.98	2.326
0.95	1.96
0.90	1.645

Then

$$C_n = \left(\hat{\theta}_N - z_{\alpha/2}\frac{\sigma}{\sqrt{n}}, \hat{\theta}_N + z_{\alpha/2}\frac{\sigma}{\sqrt{n}}\right)$$

Confidence Intervals: A recipe

3. If population standard deviation σ is unknown, use the sample standard error *se*. Note that in this case $\frac{\hat{\theta}_N - \theta}{se}$ follows a Student's-t distribution. Find the critical values $t_{\alpha/2}$ from a table.

   Then

   $$C_n = \left(\hat{\theta}_N - t_{\alpha/2}\frac{s}{\sqrt{n}}, \hat{\theta}_N + t_{\alpha/2}\frac{s}{\sqrt{n}}\right)$$

4. Generalization to other distributions is straight forward. For the critical values for common distributions you can look at statistical tables (or numerical libraries).

Examples

$\{X_1, X_2, \cdots, X_N\} \sim Poisson(\lambda)$. Find the bias, se, and MSE of the estimator $\hat{\lambda} = n^{-1} \sum X_i$

- bias $= E[\hat{\lambda}] - \lambda = (n^{-1} \sum \lambda) - \lambda = 0$
- se

$$\begin{aligned} Var(\hat{\lambda}) &= E\left[(\hat{\lambda} - E[\hat{\lambda}])^2\right] = E\left[(\hat{\lambda} - \lambda)^2\right] \\ Var(\hat{\lambda}) &= n^{-2} \sum E\left[(X_i - \lambda)^2\right] = \frac{\lambda^2}{n^2} \\ se &= \sqrt{Var(\hat{\lambda})} = \frac{\lambda}{n} \end{aligned}$$

- MSE

$$\begin{aligned} \mathbf{MSE}(\hat{\lambda}_N) &= \text{bias}^2(\hat{\lambda}_N) + \text{Var}(\hat{\lambda}_N) \\ &= 0 + n^{-2}\lambda^2 = \frac{\lambda^2}{n^2} \end{aligned}$$

MLE: Episode-II

- The **likelihood function**

$$\mathcal{L}(\theta) = \prod f(X_i;\theta) \ f \text{ density or mass}$$

- The log-likelihood:

$$\log \mathcal{L}(\theta) = \sum_i \log f(X_i;\theta)$$

- MLE estimate

$$\theta_{MLE} = \arg\max_{\theta} \log \mathcal{L}(\theta)$$

- MLE estimates are nice:
  - θ_{MLE} is consistent
  - θ_{MLE} is asymptotically normal
  - $g(\theta_{MLE})$ is the MLE for $g(\theta)$

Hypothesis Testing

1. Election poll. The result is 52% yes. Can you trust it? How close it is to the truth (population yes rate)
2. You want to know whether smoking causes cancer. How many samples you need to work with in your controlled experiment to conclude that smoking causes lung cancer with 99% confidence
3. The response rate for a product campaign is 2%. You built a mathematical model to improve it. The response rate over the test campaing for 1000 customers is 3%. Did you actually improve the benchmark rate)? Should we deploy it?

Hypothesis Testing

We formulate hypotheses about an unknown population value (scalar or vector)
Typical hypotheses:

- $\beta = 0$
- $\beta = \beta_0$
- $\beta > 0$
- $\beta < 0$
- $\beta_1 = \beta_2$
- $\sum_i \beta_i^2 = 0$

Hypothesis Testing

1. Formulate your hypothesis. This is called null-hypothesis H_0, and refers to currently accepted truth
2. The alternative-hypothesis is H_1. $H_1 = NOT(H_0)$. It is the anti-thesis
3. You fix a confidence level for the test. Usually a 95% confidence level is enough
4. You should be able to generate statistics $T(\beta)$ which is a statistics of the parameter β that is the subjects of the hypothesis
5. You should be able to know the distribution of $T(\beta)$
6. Suppose you measure the value $T\hat{(}\beta)$ as the sample value
7. If you obtain a large value of $T\hat{(}\beta)$, you reject the null hypothesis H_0, and the alternative H_1 becomes your new truth

Hypothesis Testing: Example

Let $\{X_1, X_2, \cdots, X_m\}$, and $\{Y_1, Y_2, \cdots, Y_n\}$ be two independent samples from populations with means μ_1 and μ_2 respectively. Let's test the null hypothesis $\mu_1 = \mu_2$.
Write this as

$$\begin{aligned} H_0 &: (\mu_1 - \mu_2) = 0 \\ H_1 &: (\mu_1 - \mu_2) \neq 0 \end{aligned}$$

The test statistics is $\delta = \bar{X} - \bar{Y}$.

$$\begin{aligned} \hat{se} &= \sqrt{\frac{s_X^2}{m} + \frac{s_Y^2}{n}} \\ W &= \frac{\bar{X} - \bar{Y}}{\sqrt{\frac{s_X^2}{m} + \frac{s_Y^2}{n}}} \\ W &\sim t(m + n - 2) \end{aligned}$$

If $|W_{sample}| > t_{\alpha/2}(m + n - 2)$ reject H_0

Multivariate Linear Regression

The goal of modelling in both Statistics and Machine Learning is to understand the functional relation between an output Y and a set of inputs $\{X_1, X_2, \cdots, X_d\}$ from a finite sample of data $\{(x_1, y_1), (x_2, y_2), \cdots, (x_2, y_2)\}$:

- Y: output, dependent, response variable
- X_i: input, independent, predictor, feature variable

Model-fitting for continuous outputs

Data generating process is assumed to be as follows:

$$\begin{aligned} Y &= f(X_1, X_2, \cdots, X_d) + \epsilon \\ E[\epsilon|X] &= 0 \\ Var[\epsilon|X] &= \sigma^2 \end{aligned}$$

1. First assumption: A non-linear function $f(\cdot)$ might generate the signal, and output is the sum of this signal and additive noise
2. Second assumption puts restriction on the relationship between the noise and the inputs: At a certain point X, the mean of noise is zero (A stronger constraint than lack of correlation between ϵ and X, but weaker than independence
3. Third assumption states that conditional variance is constant. Hence the first and second moments of the noise is the same everywhere in the input domain.

Model-fitting for continuous outputs

1. Multivariate Linear Regression assumes:

$$f(X) = \beta_0 + \sum_i \beta_i X_i$$

2. Generalized Additive Modelling assumes:

$$f(X) = \beta_0 + \sum_i \sum_j \beta_{ij} B_j(X_i)$$

where $B_j(X_i)$ are local univariate spline functions

3. A multilayer feed-forward ANN of n-layer assumes:

$$\begin{array}{rcl} z^\ell & = & W^\ell \cdot h^{\ell-1} \\ h^\ell & = & \sigma z^\ell \\ Y & = & W^n \cdot z^{n-1} \end{array}$$

where $z^0 = X$ and $\sigma(\cdot)$ is the activation function

Model-fitting for continuous outputs: cont'd

We need the following components to build the model

1. A specification on the signal: The mathematical form of $f(\cdot)$
2. A specification on the noise: Usually an additive noise with some further constraints
3. A loss(objective) function that measures the distance between the data and the model predictions
4. An algorithm that produces the parameters minimizing the loss

For regression problems where the output variable is continuous,

$$\begin{aligned} \text{Loss}(Y, g(X)) &\equiv E[(Y - g(X)))^2] \\ f_{Best}(X) &= \arg\min_g E[(Y - g(X)))^2] \end{aligned}$$

This is called L_2-loss. Other losses exist, but this one is good for computations.

Back to linear regression

$$\begin{aligned}
Y &= \beta_0 + \beta \cdot X + \epsilon \\
E[\epsilon|X] &= 0 \\
Var[\epsilon|X] &= \sigma^2 \\
Loss(Y, g(X)) &= E[(Y - g(X)))^2] \\
g(X) &= \hat{\beta}_0 + \hat{\beta} \cdot X \\
f_{Best}(X) &= \arg\min_g E[(Y - g(X)))^2] = E[Y|X]
\end{aligned}$$

Hence model is represented by $\{\beta_0, \beta\}$, its estimate is given by $\{\hat{\beta}_0, \hat{\beta}\}$

Linear Regression: What we are after?

1. Estimate(Train): $\{\hat{\beta}_0, \hat{\beta}\}$ from data $(x_1, y_1), (x_2, y_2), \cdots (x_2, y_N)$
2. Hypothesis Testing(Validate): Can we rely on $\{\hat{\beta}_0, \hat{\beta}\}$. Are the assumptions of regression met?
3. Goodness-of-fit(Test): Measure the performance of the model

Let's start small: Regression with only one input:

$$\begin{aligned} Y &= \beta_0 + \beta_1 X + \epsilon \\ E[\epsilon|X] &= 0 \\ Var[\epsilon|X] &= \sigma^2 \end{aligned}$$

Linear Regression: What we are after?

1. Estimate(Train): $\{\hat{\beta}_0, \hat{\beta}\}$ from data $(x_1, y_1), (x_2, y_2), \cdots (x_2, y_N)$
2. Hypothesis Testing(Validate): Can we rely on $\{\hat{\beta}_0, \hat{\beta}\}$. Are the assumptions of regression met?
3. Goodness-of-fit(Test): Measure the performance of the model

Let's start small: Regression with only one input:

$$\begin{aligned} Y &= \beta_0 + \beta_1 X + \epsilon \\ E[\epsilon|X] &= 0 \\ Var[\epsilon|X] &= \sigma^2 \end{aligned}$$

We need to estimate three scalars: $\{\hat{\beta}_0, \hat{\beta}_1, \sigma^2\}$

Bivariate regression

- $Y_i = \beta_0 + \beta_1 X_i + \epsilon, i = 1, \cdots, N$
- $\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_i, i = 1, \cdots, N$

Residuals between what is observed and what is estimated:

$$e_i = Y_i - \hat{Y}_i = Y_i - (\hat{\beta}_0 + \hat{\beta}_1 X_i)$$

Residual-sum-of-squares is defined as

$$RSS = \sum_i e_i^2$$

Minimizing RSS gives the solution $\{\hat{\beta}_0, \hat{\beta}_1\}$. These solutions are called **least squares estimates**

Bivariate regression

The last squares estimates are given by

$$\begin{aligned}\hat{\beta}_1 &= \sum_i \frac{(X_i - \bar{X}_n)(Y_i - \bar{Y}_n)}{\sum_i (X_i - \bar{X}_n)^2} = r\frac{s_y}{s_x} \\ \hat{\beta}_0 &= \hat{Y} - \hat{\beta}_1 \hat{X}\end{aligned}$$

An un-biased estimate of σ^2 is given by

$$\hat{\sigma}^2 = \left(\frac{1}{n-2}\right)\sum_i e_i^2$$

Bivariate regression: Ex-1

Data is generated by

$$\begin{aligned} X &\sim U(0,1) \\ \beta_0 &= 3.5 \\ \beta_1 &= 0.2 \\ \epsilon &\sim \mathcal{N}(0, 0.1) \\ N &= 100 \end{aligned}$$

Estimates, standard errors, test stats, p-values, confidence intervals

Variable	Est.	S.E.	t-Stat	P-value	Lower 95%	Upper 95%
Intercept	**3.473**	0.018	192.590	3.353E-128	3.437	3.509
X_1	**0.276**	0.032	8.479	2.378E-13	0.212	0.341

Bivariate regression: Variations on Ex-1

Same as Ex-1 but $N = 1000$ (Increase the sample size)

Variable	Est.	S.E.	t-Stat	P-value	Lower 95%	Upper 95%
Intercept	**3.506**	0.006	559.025	0	3.494	3.518
X_1	**0.200**	0.011	18.1252	1.084E-63	0.178	0.221

Same as Ex-1 but $\sigma^2 = 1$ (Increase the noise)

Variable	Est.	S.E.	t-Stat	P-value	Lower 95%	Upper 95%
Intercept	**3.339**	0.188	17.675	3.053E-32	2.964	3.714
X_1	**0.367**	0.341	1.075	0.284	-0.311	1.046

Same as Ex-1 but $\epsilon \sim U(-0.25, 0.25)$ (ϵ not normal)

Variable	Est.	S.E.	t-Stat	P-value	Lower 95%	Upper 95%
Intercept	**3.519**	0.028	124.005	1.52E-109	3.462	3.575
X_1	**0.170**	0.0513	3.311	0.001	0.068	0.272

Bivariate regression: Analysis of Variance

A decomposition of variance:

$$\begin{aligned} \sum_i (Y_i - \bar{Y})^2 &= \sum_i (\hat{Y}_i - \bar{Y})^2 + \sum_i e_i^2 \\ \text{TSS} &= \text{ESS} + \text{RSS} \end{aligned}$$

TSS: Total sum of squared deviations in Y
ESS: Explained sum of squares from the regression of Y on X
RSS: Residual (unexplained) sum of squares the regression of Y on X

$$r^2 = 1 - \frac{RSS}{TSS} = \frac{ESS}{TSS}$$

r^2: proportion of the Y variation *explained* by the inputs X
r^2 represents the quality of model-fit

Multivariate Linear Regression

Define the *design matrix* X as follows

$$X_{i,j} = \begin{pmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,d} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N,1} & X_{N,2} & \cdots & X_{N,d} \end{pmatrix}$$

Rows refers to the samples, columns refer to the variables
Assuming that the matrix $(X^T X)$ is invertible, then:

$$\begin{aligned} \hat{\beta} &= (X^T X)^{-1}(X^T Y) \\ Var(\hat{\beta}|X) &= \sigma^2 (X^T X)^{-1} \\ \hat{\beta} &\sim \mathcal{N}(\beta, \sigma^2 (X^T X)^{-1}) \end{aligned}$$

Multivariate Linear Regression

$$\begin{aligned} e_i &= Y_i - X_i\hat{\beta} \text{ (residuals)} \\ \hat{\sigma}^2 &= \left(\frac{1}{N-k}\right) e_i^2 = \frac{RSS}{N-k} \end{aligned}$$

Where k is the number of variables in the model. An approximate $(1-\alpha)$ CI for β_j is given by

$$\left(\hat{\beta}_j - t_{\alpha/2}\hat{se}(\hat{\beta}_j), \hat{\beta}_j + t_{\alpha/2}\hat{se}(\hat{\beta}_j)\right)$$

where $\hat{se}(\hat{\beta}_j)$ is the j-th diagonal of $\hat{\sigma}^2(X^TX)^{-1}$

MLR: Examination of an output (Assume a confidence level of 95%. Hence p-values for the estimated values should be ≤ 0.05)

Variable	$\hat{\beta}_j$	$\hat{se}(\hat{\beta}_j)$	t-Stat	P-value
(Intercept)	-589.39	167.59	-3.51	0.001
X_1	1.04	0.45	2.33	0.025
X_2	11.29	13.24	0.85	0.399
X_3	1.18	0.68	1.7	0.093
X_4	0.96	0.25	3.86	0.000
X_5	0.11	0.15	0.69	0.493
X_6	0.30	0.22	1.36	0.181
X_7	0.09	0.14	0.65	0.518
X_8	-0.68	0.48	-1.4	0.165
X_9	2.15	0.95	2.26	0.030
$X_1 0$	-0.08	0.09	-0.91	0.367

- Not all variables are important in affecting Y
- Some of the variables are heavily correlated with each other. Once you include one of them, the other are not important

HOW TO SELECT MODEL VARIABLES?

MLR: Model Selection

First, randomly divide the sample into two datasets: *Training* and *Validation* sets

1. Best subset selection: There are 2^k possible models with k variables. Find $\hat{\beta}$ for each selection. Compute the R^2 on the validation data-set. Pick the best $\hat{\beta}$ giving the maximum performance. NOT FEASIBLE for $k > 30$ (A real data-set consists of hundreds probably thousands variables)
2. Forward selection
   - 2.1 Start with the variable that has the highest correlation with the output. Include it in the model. Check the p-values for the given confidence level. If ok CONTINUE. If not stop.
   - 2.2 Compute the partial correlations of the variables outside the model (given the current model variables). Include the variable with the highest partial correlation. If any of the p-values inside the model drops below the threshold STOP. If the delta-performance over the validation set is below the threshold STOP. If not include the variable into the model and CONTINUE.

MLR: Model Selection

3. Backward selection: Start with all of the variables. Choose the variable with the highest p-value. If it is below the threshold STOP. If not, exclude this variable. Recompute all the variables. Repeat.
4. Use lasso regularization: Beyond the scope.

Industrial practice:

- Whether it is linear regression or GAM, you have to employ variable selection
- You have to validate the selection strategy for each step on validation data set: p-values must be obeyed, signs of the variables should not change, performance increase should be above a certain threshold, etc.
- Stepwise selection strategy (a variation of Forward Selection) is used the most
- Selection strategies are greedy strategies: It does not give you the optimal subset. But that is OK
- Use LASSO if available. See the *ESL, Chapter-3*

Multivariate Linear Regression: Problems

- Very sensitive to outliers in input(s) or the output: Even, with a single data point (X_i, Y_i), you can change the estimates $\hat{\beta}$ *radically*. Not robust. You have to pre-process outliers BEFORE modelling, and at the PREDICTION stage
- It is a linear function in its inputs. Does NOT account for non-linear effects. Non-linear effects are ABUNDANT in real-life data. You have to explicitly create non-linear features for it to work
- Could not handle nominal variables automatically. You have to pre-process them: One-hot-encoding and/or simultaneous grouping if large cardinality
- Could nor handle missing values automatically. You have to pre-process them

Multivariate Linear Regression: Problems

- Its quality does not increase monotonically with increasing data: Its learning capacity is low. Use it only when you have minimal amount of data (in the order of hundreds of samples)
- Overall judgement: Do not use it unless sample size is low enough so that complex models are not warranted
- A natural extension is Generalized Additive Modelling with pre-processing. We will present it soon as an explainable and accurate learning algorithm

A Short Recipe for MLR Model Building

1. Handle the outliers for both inputs and the outputs. Try not to lose data. Outliers in the output are more problematic. Delete extreme samples if the sparsity is low
2. Handle the missing values. Do NOT lose data. Use imputation (simple or complex)
3. Handle the categorical variables: If the cardinality is low convert int into many-scalar variables. If it is high, group the sparce categories into one big category and then convert the resulting variable into many-scalar variables (This is called one-hot-encoding)
4. Detect the bivariate non-linear relationships: Use scatter-plots to detect possible non-linearities. You can use suitable functional transformations (taking logarithms, using polynomial smoothing, etc.) create new non-linear variables. Use this new variables as extra inputs

A Short Recipe for MLR Model Building

5. Use lasso regression if you have it at your disposal. If not use Stepwise Variable Selection. Repeat that significances, signs, etc. are preserved at ach stage on validation set as well
6. Analyze the distribution of the resulting error vector. Validate your initial assumptions: constant conditional variance, normality, etc. Go to steps 1,2,3,4 and/or create new variables if the distribution of the error is not validated. Build a conditional variance model if you detect heterogeneous variance. Use the conditional variance in generating confidence intervals for your (out-of-sample) predictions.