NANYANG TECHNOLOGICAL UNIVERSITY

BC2406 Business Analytics I: Predictive Techniques

Seminars 7
Linear Regression Model II

Instructor: Prof. Lee Gun-woong

Nanyang Business School

Review

- Deterministic vs. Probabilistic Models
- Linear Relationship
- Correlation vs. Regression
- Ordinary Least Square (OLS)
- Model Evaluation
  - Model fit: R^2 vs. adjusted R^2
  - Model Specification: F-test, H_0: $\beta_1 = \beta_2 = \ldots = 0$
  - Model Predictive Performance: MSE = SSE / #obs.

Interpretation and Prediction

NANYANG TECHNOLOGICAL UNIVERSITY

How to interpret log-transformed coefficients

Dependent Variable	Independent Variable	Estimated Price ($)	Example
Sales	Price	- 10.232	A one-unit increase in Price decreases Sales by $10.232
Sales	Log(Price)	- 9.389	A one-percent increase in Price decreases Sales by $0.09389 (=coefficient/100)
Log(Sales)	Price	- 0.045	A one-unit increase in Price decreases Sales by 4.5% (=coefficient*100)
Log(Sales)	Log(Price)	- 0.032	A one-percent increase in Price decreases Sales by 0.032%

Check the video clips: http://www.cazaar.com/ta/econ113/interpreting-beta

Multiple Liner Regression Model on *mpg*

- **Regression Model**: $mpg = \beta_0 + \beta_1 hp + \beta_2 cyl + \beta_3 am + \varepsilon$

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 30.88834    2.78422  11.094 9.27e-12 ***
hp          -0.03688    0.01452  -2.540  0.01693 *  
cyl         -1.12721    0.63417  -1.777  0.08636 .  
am           3.90428    1.29659   3.011  0.00546 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

- **Predicted Model**: $\widehat{mpg} = 30.88834 - 0.03688*hp - 1.12721cyl + 3.90428**am$
- **Expected mpg**
  - E[*mpg* | *hp* = 0, *cyl* = 0, *am* = 0] = 30.88834
  - E[*mpg* | *hp* = 0, *cyl* = 0, *am* = 1] = 30.88834 + 3.90428
    - ✓ E[*mpg* | *hp* = 0, *cyl* = 0, *am* = 1] - E[*mpg* | *hp* = 0, *cyl* = 0, *am* = 0] = 3.90428
  - E[*mpg* | *hp* = 100, *cyl* = 0, *am* = 0] = 30.88834 – 0.03688 * 100
  - E[*mpg* | *hp* = 110, *cyl* = 0, *am* = 0] = 30.88834 – 0.03688 * 110
    - ✓ E[*mpg* | *hp* = 110] - E[*mpg* | *hp* = 100] = (– 0.03688 * 110) – (– 0.03688* 100) = - 0.03688 * 10
  - E[*mpg* | *hp* = 100, *cyl* = 0, *am* = 0] = 30.88834 – 0.03688 * 100
  - E[*mpg* | *hp* = 110, *cyl* = 0, *am* = 1] = 30.88834 – 0.03688 * 110 + 3.90428
    - ✓ E[*mpg* | *hp* = 110, *am* = 1]] - E[*mpg* | *hp* = 100, *am* = 0]] = - 0.03688 * 10 + 3.90428

Calculate a car’s expected *mpg* when *hp* = 150, *cyl* = 8, and *am*= 1

NANYANG TECHNOLOGICAL UNIVERSITY

Multiple Liner Regression Model on *Mobile App Sales*

- Regression Model

$$-\log(Rank) = \beta_0 + \beta_1 Price + \beta_2 Screenshots$$
$$+ \beta_3 Rating\_Score + \beta_4 \log(Rating\_Num)$$
$$+ \beta_5 Business + \beta_6 Finance + \beta_7 Health + \beta_8 Utlilities$$
$$+ \varepsilon$$

- Estimation Output

```
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    -6.556413   0.160590 -40.827  < 2e-16 ***
Price          -0.008649   0.006326  -1.367   0.1717
Screenshots     0.084122   0.025914   3.246   0.0012 **
Rating_Score    0.037921   0.025746   1.473   0.1410
Log_Rating_Num  0.152256   0.011985  12.704  < 2e-16 ***
Business        0.677768   0.086081   7.874 6.58e-15 ***
Finance         0.638288   0.084796   7.527 8.92e-14 ***
Health          0.491402   0.081207   6.051 1.81e-09 ***
Utilities       0.518035   0.080418   6.442 1.59e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **Predicted Model**

$$-\widehat{\ln(Rank)} = -6.556^{***} - 0.009 Price + 0.084^{**} Screenshots + 0.038 Rating\_Score + 0.152^{***} \log(Rating\_Num) + 0.678^{***} Business + 0.638^{***} Finance + 0.491^{***} Health + 0.518^{***} Utilities$$

$^{***}=p<0.001$, $^{**}=p<0.01$, $^{**}=p<0.05$

- **Expected App Sales**
  - E[*Sales* | *Price* = *Rating_Score* = … = *Utilities* = 0] = -6.556 (expected sales for apps in Games)
  - E[*Sales* | *Price* = *Rating_Score* = … = 0, *Utilities* = 1] = -6.556 + 0.518
    - ✓ E[*Sales* | *Utilities* = 1] - E[*Sales* | *Utilities* = 0] = 0.518
      - ✓ Apps in Utilities improved sales by an average of 51.8% as compared to Apps in Games (e.g., 152 vs. 100 copies)

  - E[*Sales* | *Rating_Score* = 3.0] = - 6.556 + 0.038 * 3.0
  - E[*Sales* | *Rating_Score* = 4.0] = - 6.556 + 0.038 * 4.0
    - ✓ E[*Sales* | *Rating_Score* = 4.0] - E[*Sales* | *Rating_Score* = 3.0] = 0.038 * 1.0

  - E[*Sales* | *Rating_Score* = 3.0, *Finance* = 1] = -6.556 + 0.038*3.0 + 0.638
  - E[*Sales* | *Rating_Score* = 4.0, *Business* = 1] = -6.556 + 0.038*4.0 + 0.518
    - ✓ E[*Sales* | *Rating_Score* = 4.0, *Business* = 1] - E[*Sales* | *Rating_Score* = 3.0, *Finance* = 1] = 0.038*1.0 + (0.638-0.491)

  - E[*Sales* | *Price* = $0.99, *Rating_Num* = 100] = -6.556 - 0.009*0.99 + 0.152*log(100)
  - E[*Sales* | *Price* = $1.99, *Rating_Num* = 1000] =-6.556 - 0.009*1.99 + 0.152*log(1000)
    - ✓ E[*Sales* | *Price* = $1.99, *Rating_Num* = 1000]- E[*Sales* | *Price* = $0.99, *Rating_Num* = 100] = -0.009*1.0 + 0.152*log(1000/100)
      - ✓ $1.99-priced Apps having 1,000 ratings improved sales by an average of 34.1% as compared to $0.99-priced Apps having 100 ratings.