پروژه رگرسیون ۲

موضوع: برازش و بررسی مدل رگرسیون خطی چندگانه و بررسی نمونهای عملکرد رگرسیون بیز

استاد: جناب دکتر اسکندری

دانشجو: علی شکارچی

دانشکده آمار، ریاضی و رایانه

زمستان ۱۴۰۲

فهرست :

- جمع آوری داده ها
- هدف پروژه
- متغیر گزینی
- بررسی متغیر کیفی با عنوان data_1
  - بررسی آزمون همگنی واریانس
  - آزمون معناداری مدل رگرسیونی
  - آزمون معناداری ضرایب انفرادی رگرسیون
- بررسی متغیر کیفی با عنوان data_2
  - بررسی آزمون همگنی واریانس
  - آزمون معناداری مدل رگرسیونی
  - آزمون معناداری ضرایب انفرادی رگرسیون
- بررسی نمونه ای عملکرد رگرسیون بیز
- فاصله اطمینان بیزی برای مدل رگرسیونی

جمع آوری داده ها

داده های موجود در لینک زیر در پروژه مورد استفاده قرار گرفته است.

لینک دیتا :

https://www.kaggle.com/datasets/nikhil7280/student-performance-multiple-linear-regression?resource=download

شرح مجموعه داده:

مجموعه داده عملکرد دانش آموز مجموعه داده ای است که برای بررسی عوامل موثر بر عملکرد تحصیلی دانش آموزان طراحی شده است. مجموعه داده شامل ۱۰۰۰۰ رکورد دانش آموز است که هر رکورد حاوی اطلاعاتی در مورد پیش بینی کننده های مختلف و یک شاخص عملکرد است.

متغیرها(متغیر های پیش بین):

- ساعت مطالعه(Hours Studied): تعداد کل ساعات مطالعه هر دانش آموز.
- نمرات قبلی(Previous Scores): نمرات کسب شده توسط دانش آموزان در آزمون های قبلی.
- فعالیت های فوق برنامه(Extracurricular Activities): اینکه آیا دانش آموز در فعالیت های فوق برنامه شرکت می کند (بله یا خیر).
- ساعات خواب(Sleep Hours): میانگین ساعات خواب دانش آموز در روز.
- نمونه سوالات تمرین شده(Sample Question Papers Practiced): تعداد نمونه سوالاتی که دانش آموز تمرین کرده است.

متغیر هدف:

- شاخص عملکرد: معیاری از عملکرد کلی هر دانش آموز. شاخص عملکرد نشان دهنده عملکرد تحصیلی دانش آموز است و به نزدیکترین عدد صحیح گرد شده است. این شاخص از ۱۰ تا ۱۰۰ متغیر است که مقادیر بالاتر نشان دهنده عملکرد بهتر است.

هدف از انجام پروژه :

مجموعه داده ارائه بینشی در مورد رابطه بین متغیرهای پیش بینی کننده و شاخص عملکرد است. محققان و تحلیلگران داده‌ها می‌توانند از این مجموعه داده برای بررسی تأثیر ساعات مطالعه، نمرات قبلی، فعالیت‌های فوق برنامه، ساعات خواب و نمونه سوالات بر عملکرد دانش‌آموز استفاده کنند.

درواقع رگرسیون خطی چندگانه (Multiple linear regression)با نام متداول MLR که به سادگی به عنوان رگرسیون چندگانه نیز شناخته می شود، یک تکنیک آماری است که از چندین متغیر توضیحی برای پیش بینی نتیجه یک متغیر پاسخ استفاده می کند هدف رگرسیون خطی چندگانه مدل سازی رابطه خطی بین متغیرهای توضیحی (مستقل) و متغیرهای پاسخ (وابسته) است.

همچنین هدف از انجام آزمون هایی که در این پروژه انجام می شود این است که ما تا حد ممکن بهترین برازش را روی بهترین مدل انجام دهیم. به عبارتی یک رگرسیون متغیر پاسخ را روی این تعداد متغیر مستقل بردازش کنیم.

P.S: لطفاً توجه داشته باشید که این مجموعه داده مصنوعی است و برای اهداف توضیحی ایجاد شده است. روابط بین متغیرها و شاخص عملکرد ممکن است منعکس کننده سناریوهای دنیای واقعی نباشد

با استفاده از این داده ها مدل رگرسیونی تشکیل داده و بررسی های مورد نظر را روی مدل انجام خواهد داد . در ابتدا مرحله متغیرگزینی :

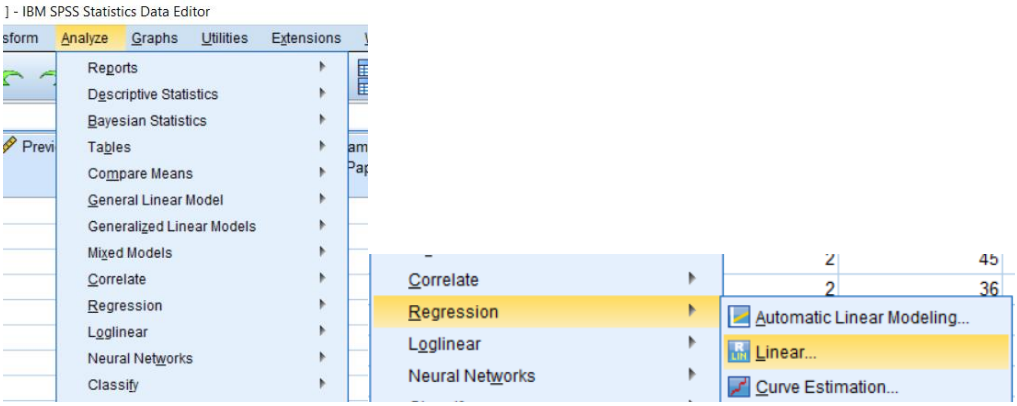
متغیرگزینی

باید بررسی شود که همه این تعداد متغیر مستقل سهم آنچنانی در مدل دارند یا خیر . ممکن است بعضی از آنها ضعیف باشند. پس از کل متغیر های مستقل یک تعدادی برای مدل مناسب است و یک تعدادی هم مناسب نیست. سپس متغیر های مستقلی که مناسب هستند را گزینش می کنیم.

داده ها از اکسل به صورت زیر در spss فراخوانی شده است.

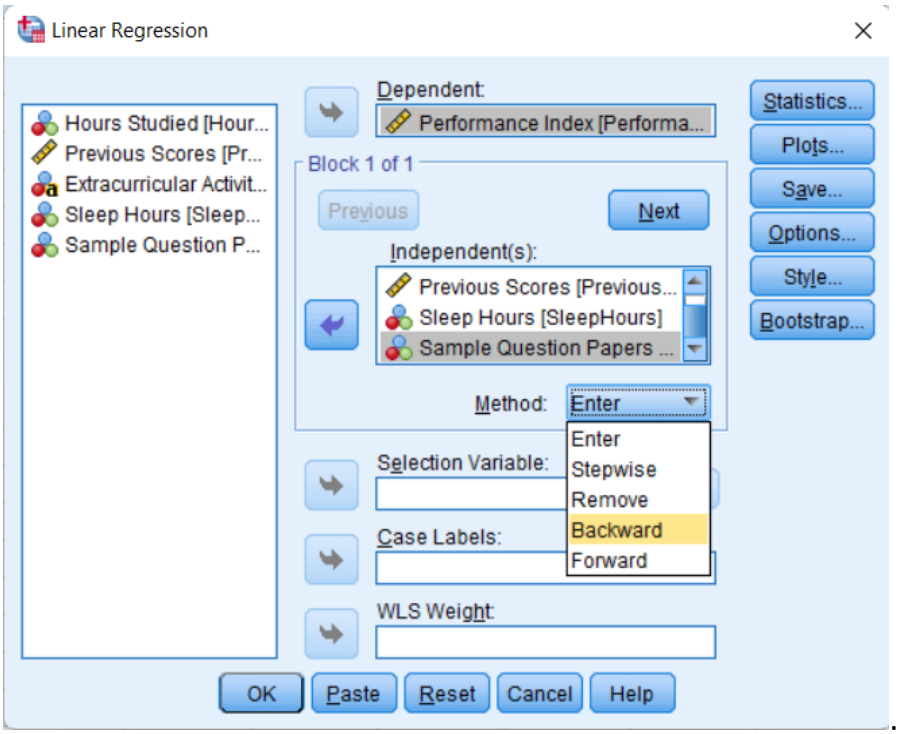
	HoursStudied	PreviousScores	Extra	SleepHours	SampleQuestionPapersPracticed	PerformanceIndex	var	var
1	7	99	Yes	9	1	91		
2	4	82	No	4	2	65		
3	8	51	Yes	7	2	45		
4	5	52	Yes	5	2	36		
5	7	75	No	8	5	66		
6	3	78	No	9	6	61		

سپس در قسمت Analyze گزینه Regression پس از آن گزینه Linear را انتخاب کرده.



متغیر پاسخ را در قسمت Dependent و متغیر های پیشبین را به جز متغیر کیفی Extracurricular Activities در قسمت Independent وارد کرده.

برای متغیر گزینی در spss از متد backward استفاده شده است.



از جدول های خروجی spss جدول زیر مورد توجه قرار میگیرد.

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients			Collinearity Statistics	
		B	Std. Error	Beta	t	Sig.	Tolerance	VIF
1	(Constant)	-33.764	.127		-266.189	.000		
	Hours Studied	2.853	.008	.385	358.403	.000	1.000	1.000
	Previous Scores	1.019	.001	.919	857.021	.000	1.000	1.000
	Sleep Hours	.476	.012	.042	39.193	.000	1.000	1.000
	Sample Question Papers Practiced	.195	.007	.029	27.152	.000	1.000	1.000

با توجه به جدول coefficients همخطی (رابطه بین متغیرهای ورودی) را بررسی می شود.

یک شاخص برای تعیین میزان همخطی: $VIF = \frac{1}{1-R^2}$

با توجه به جدول، VIF همه متغیرهای ورودی یک است و اگر مقدار VIF کمتر از ۲ و ۳ باشد ، قابل اغماض است . پس همخطی میان متغیر های ورودی (مستقل) وجود ندارد و تمام متغیر های موجود مناسب هستند و وارد مدل میشنود.

بررسی آزمون همگنی واریانس

در حالت کلی ازمون همگنی واریانس به صورت زیر مورد بررسی قرار میگیرد:

$$X_i \sim N(\theta_1, \sigma_1^2) \quad , \quad Y_j \sim N(\theta_2, \sigma_2^2)$$

$$X \sim N(\theta, \sigma^2) \ \rightarrow \ X - \theta \sim N(0, \sigma^2) \ \rightarrow \ Z = \frac{X-\theta}{\sigma} \sim N(0,1)$$

$$Z^2 \sim \chi^2_{(1)} \ \rightarrow \sum_{i=1}^{n} Z^2 \sim \chi^2_{(n)}$$

آزمون فرض آماری (همگنی یا ناهمنگی واریاس):

$$\begin{cases} H_0 : \sigma_1^2 = \sigma_2^2 \\ H_1 : \sigma_1^2 \neq \sigma_2^2 \end{cases}$$

$$A = \frac{\sum (x_i - \theta_1)^2}{\sigma^2} \sim \chi^2_{(n)} \quad , \quad B = \frac{\sum (y_j - \theta_2)^2}{\sigma^2} \sim \chi^2_{(m)}$$

$$\frac{\sum (x_i - \theta_1)^2}{\sum (y_j - \theta_2)^2} \sim F_{(n,m)}$$

هرگاه θ_1 و θ_2 نامعلوم باشند از برآوردگر آنها استفاده میشود :

$$\frac{\sum (x_i - \bar{X})^2}{\sum (y_j - \bar{Y})^2} \sim F_{(n-1 \, , \ m-1)}$$

در حالت کلی مدل رگرسیونی زیر را درنظر می گیریم :

$$y_i = \beta_0 + \beta_1 x_{11} + \cdots + \beta_p x_{iP} + e_i$$

در ادامه این موضوع را از ابتدا و برای داده های بدست آمده ، در نرم افزار R Studio مورد بررسی قرار می دهیم.

قبل از شروع کدنویسی ابتدا پکیج readxl در R را باز میکنیم.

داده ها در R studio فراخوانی میکنیم .

ماتریس data_frame متشکل از کل داده ها خواهد بود.

```
library(readxl)

data_file <- read_excel("C:\\Users\\Zahra\\Desktop\\پروژه رگرسیون\\Student_Performance.xlsx")
data_frame <- data.matrix(data_file)
```

با توجه به روش متغیر ظاهری مجموعه داده های کیفی به صورت مجزا بررسی خواهند شد و در این قسمت عدد منتسب به yes عدد ۲ و عدد منتسب به no عدد ۱ خواهد بود که به بررسی داده های no پرداخته میشود (data_1)

تعداد data_1 : n1

ستون ۶ (performance index) در data_1: y1

ستون ۳ (Extracurricular Activities) :داده های کیفی

ماتریس ضرایب توسط ستون های ۱و۲و۴و ۵ در data_1 تشکیل میشود. (همانطور که میدانیم این ماتریس ۵ ستون خواهد داشت و تمامی درایه های ستون اول ۱ است.)

ستون اول (Hours Studied) در data_1 : x1_1

ستون دوم (Previous Scores) در data_1 : x1_2

ستون چهارم (Sleep Hours) در data_1 :x_3

ستون پنجم (Sample Question Papers Practiced) در data_1 :x1_4

ماتریس ضرایب در data_1 : x1

```
data_1 <- subset(data_frame, data_frame[, 3] == 1)
n1 <- nrow(data_1)
n1
y1 <- data_1[, 6]
y1
x1_0 <- rep(1, n1)
x1_1 <- data_1[, 1]
x1_2 <- data_1[, 2]
x1_3 <- data_1[, 4]
x1_4 <- data_1[, 5]
x1 <- matrix(c(x1_0, x1_1, x1_2, x1_3, x1_4), ncol = 5)
x1
```

خروجی به صورت زیر خواهد بود:

```
> n1 <- nrow(data_1)
> n1
[1] 5052
 y1 <- data_1[, 6]
 y1
  [1] 65 66 61 61 69 84 73 27 33 68 43 63 85 57 35 66 42 68 64 45 36 54 53 75 78 91 78 38 71 54 42
 [32] 91 74 61 45 71 67 95 29 21 30 57 27 34 76 57 45 81 66 56 25 56 46 45 70 36 71 49 43 77 34 49
 [63] 69 84 41 41 58 94 40 36 47 83 36 74 42 26 42 85 33 77 72 53 16 45 49 49 73 65 72 67 73 72 42
 [94] 47 77 49 30 75 78 89 48 27 66 65 29 26 72 41 63 59 46 42 30 77 92 70 35 66 81 61 27 77 43 48
[125] 19 41 28 52 53 52 64 35 73 47 36 63 58 58 62 37 86 88 38 57 35 51 92 39 56 69 86 89 44 33 36
[156] 94 44 64 30 51 10 30 74 51 68 22 67 50 68 41 83 57 18 62 66 85 62 66 43 25 88 44 34 27 78 33
[187] 68 45 58 75 70 62 57 82 75 69 75 59 56 54 83 59 70 41 54 64 66 26 30 28 48 76 71 60 77 36 96
[218] 22 79 82 37 42 72 56 43 24 64 53 84 53 67 58 41 89 85 79 67 80 26 57 91 29 84 33 48 66 89 49
[249] 57 34 76 22 30 67 71 46 76 64 68 58 64 70 77 60 48 81 29 32 57 58 22 30 59 43 62 63 73 76 34
[280] 57 72 49 77 56 89 65 48 47 48 57 45 82 90 42 74 60 73 32 67 65 69 62 43 84 15 39 26 73 34 85
[311] 18 44 72 40 31 25 27 34 70 81 30 64 66 32 51 67 57 77 24 60 39 19 68 49 74 26 73 61 57 37 54
[342] 22 68 52 64 45 75 47 45 43 32 64 33 36 45 58 85 78 63 76 51 36 74 46 47 46 34 27 84 81 41 69
[373] 34 52 77 59 35 97 37 18 84 60 78 31 68 37 69 55 12 88 69 88 41 42 73 23 85 93 79 57 28 85 41
[404] 57 43 46 61 87 78 18 36 58 69 58 74 74 60 70 17 34 51 27 67 84 67 67 77 59 88 60 14 79 20 42
[435] 31 75 22 46 44 90 47 56 66 72 24 45 56 29 28 84 42 51 74 50 62 90 71 42 85 67 32 62 48 33 43
[466] 29 79 41 61 69 38 54 93 76 54 68 33 40 87 70 38 71 58 38 57 63 84 84 15 43 48 66 53 51 43 68
[497] 80 57 56 53 52 47 78 40 60 29 68 76 57 61 34 44 18 82 18 88 78 35 79 81 87 20 75 38 78 91 49
[528] 45 65 53 57 50 75 61 78 36 49 71 77 52 25 76 40 74 50 79 31 33 65 45 55 28 58 67 79 45 70 58
[559] 50 77 75 40 71 63 37 72 68 59 97 83 47 51 32 63 70 32 32 21 81 11 65 37 75 61 17 87 52 42 91
[590] 64 78 24 36 87 18 46 62 58 37 40 38 39 47 45 56 30 80 27 81 83 52 50 46 51 31 29 51 79 52 81
[621] 55 68 50 78 26 52 57 57 75 54 82 43 37 23 42 65 57 59 46 84 73 74 94 89 62 59 57 80 50 18 68
[652] 39 67 84 50 65 74 35 64 66 71 13 59 67 35 59 57 51 22 71 73 40 25 44 26 31 75 64 43 54 56 66
[683] 25 20 86 46 73 42 27 64 71 43 42 60 76 28 45 48 57 61 86 18 24 32 66 22 42 56 72 31 44 96 51
[714] 58 23 50 44 73 76 50 30 51 76 66 50 40 42 29 49 74 42 30 38 75 75 26 49 42 14 64 16 69 62 43
[745] 70 72 46 44 45 41 62 70 67 69 35 55 56 58 40 68 43 61 36 67 29 74 37 79 65 72 93 41 36 47 57
[776] 22 27 49 66 36 27 71 76 31 26 79 33 40 58 28 36 69 84 58 36 40 37 61 72 57 42 90 89 33 68 33
[807] 73 50 93 47 77 87 28 28 37 66 44 33 67 59 75 19 16 45 44 50 15 71 49 41 40 25 62 69 82 82 49
[838] 82 40 48 28 60 43 59 71 52 28 27 55 39 47 59 46 47 26 45 88 67 75 91 30 48 59 65 73 50 36 65
[869] 49 32 69 60 88 61 69 73 22 74 83 82 29 50 80 21 39 74 76 48 48 40 48 60 73 22 74 58 76 55 37
[900] 25 57 39 45 72 63 36 49 43 60 53 53 63 31 57 76 38 49 74 32 52 41 87 32 89 36 25 24 35 58 76
[931] 34 67 36 66 78 48 50 40 94 39 49 89 79 74 83 51 41 53 31 58 62 61 39 40 35 55 60 45 76 33 33
[962] 25 76 38 63 60 49 66 71 67 46 86 37 77 32 34 59 52 84 57 30 47 15 49 45 68 30 33 28 64 88 30
[993] 52 45 49 43 52 53 37 77
```

ماتریس ضرایب x1 در این قالب اجرا میشود که در این بخش قسمتی از آن برای نمونه قرار داده شده است.

```
> x1_0 <- rep(1, n1)
> x1_1 <- data_1[, 1]
> x1_2 <- data_1[, 2]
> x1_3 <- data_1[, 4]
> x1_4 <- data_1[, 5]
> x1 <- matrix(c(x1_0, x1_1, x1_2, x1_3, x1_4), ncol = 5)
> x1
      [,1] [,2] [,3] [,4] [,5]
 [1,]    1    4   82    4    2
 [2,]    1    7   75    8    5
 [3,]    1    3   78    9    6
 [4,]    1    5   77    8    2
 [5,]    1    4   89    4    0
 [6,]    1    8   91    4    5
 [7,]    1    8   79    6    2
 [8,]    1    3   47    9    2
 [9,]    1    6   47    4    2
[10,]    1    5   79    7    8
[11,]    1    2   72    4    3
[12,]    1    5   75    7    0
[13,]    1    6   96    9    0
[14,]    1    1   85    5    6
[15,]    1    3   61    6    3
[16,]    1    4   79    8    9
[17,]    1    4   59    8    3
[18,]    1    9   72    8    2
[19,]    1    9   68    5    3
[20,]    1    5   62    7    4
```

مدل خطی ساده تشکیل میشود به این معنا که از طریق فرمول زیر ماتریس betahat1 تشکیل شده و با ضرب x1در آن مقدار برآورد یعنی y1hat به دست می آید.

مقدار مانده ها از تفاضل y1 و y1hat محاسبه میشود.

$$\hat{\beta}_1 = (X'X)^{-1}(X'Y) \quad , \quad \hat{y} = X\hat{\beta}_1 \quad , \quad \hat{e} = y - \hat{y}$$

```
betahat1 <- solve(t(x1)%*%x1)%*%(t(x1)%*%y1)
betahat1
y1hat <- x1%*%betahat1
y1hat
e1 <- y1-y1hat
e1
qqnorm(e1)
```

خروجی به صورت زیر خواهد بود:

```
> betahat1 <- solve(t(x1)%*%x1)%*%(t(x1)%*%y1)
> betahat1
             [,1]
[1,] -34.0452900
[2,]   2.8411857
[3,]   1.0183656
[4,]   0.4832404
[5,]   0.1972391
```

از ضرایب فوق می توانیم بفهمیم که کدام متغیر اثر مستقیم و کدام متغیر اثر غیر مستقیم دارد.

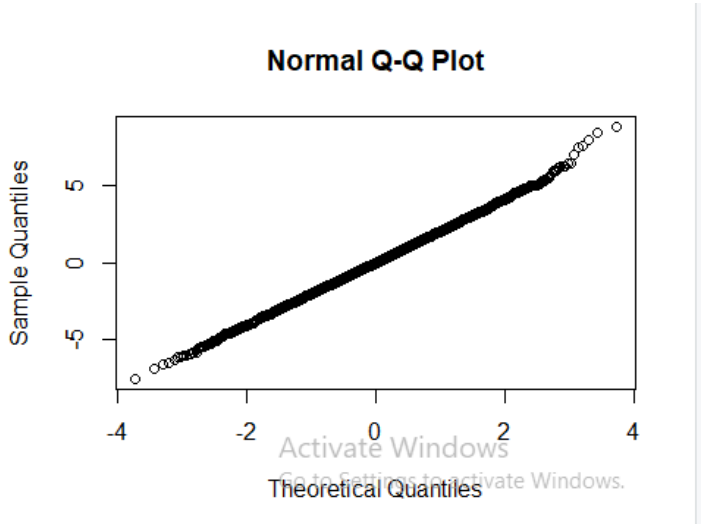
متغیری که دارای ضریب مثبت است اثر مستقیم و متغیری که دارای ضریب منفی است اثر غیر مستقیم دارد.

```
 [2,] 67.07255
 [3,] 59.44338
 [4,] 62.83519
 [5,] 69.88695
 [6,] 84.27462
 [7,] 72.42900
 [8,] 27.08509
 [9,] 33.19245
[10,] 65.57212
[11,] 47.48408
[12,] 59.92074
[13,] 85.11409
[14,] 58.95661
[15,] 40.08973
[16,] 63.41141
[17,] 41.86066
[18,] 69.10811
[19,] 63.78216
[20,] 47.47095
```

مقدار مانده ها را مشاهده می کنیم که مقدار های بسیار کوچکی بدست آمده اند و این موضوع نشان می دهد که مدل رگرسیونی ما مدل بسیار خوبی است.

```
 [2,] -1.072549e+00
 [3,]  1.556617e+00
 [4,] -1.835192e+00
 [5,] -8.869538e-01
 [6,] -2.746233e-01
 [7,]  5.710007e-01
 [8,] -8.509238e-02
 [9,] -1.924476e-01
[10,]  2.427883e+00
[11,] -4.484084e+00
[12,]  3.079258e+00
[13,] -1.140863e-01
[14,] -1.956609e+00
[15,] -5.089729e+00
[16,]  2.588589e+00
[17,]  1.393358e-01
[18,] -1.108106e+00
[19,]  2.178380e-01
[20,] -2.470945e+00
[21,] -2.023187e+00
[22,] -7.765386e-01
[23,] -1.581659e+00
[24,]  3.386246e+00
[25,]  1.514905e+00
[26,]  3.866345e+00
[27,] -2.018725e+00
[28,] -2.465187e+00
[29,] -1.587492e+00
[30,]  9.002325e-01
```

سپس با دستور qqnorm نمودار مانده ها مورد بررسی قرار میگیرد.



طبق نمودار مشاهده شده توزیع مانده ها به صورت نرمال می باشد و به دنبال آن توزیع متغیر پاسخ و پارامتر مدل نرمال است.

در این مرحله آزمون همگنی واریانس روی data1 انجام میشود.

ابتدا e1 از کوچک به بزرگ مرتب شده.

e1به سه قسمت تقسیم میشود (قسمت اول و سوم مد نظر است که به ترتیب ek1_1 , ek3_1 نامیده میشود.)

سپس آماره آزمون یعنی Q1 بدست آمده و با آزمون فیشربا درجه آزادی k3_1 و k1_1 در سطح معنا دار ۰.۰۵ مقایسه میشود.

اگر مقدارآماره بین احتمال ۰.۰۵ و ۰.۹۵ باشد پذیرش H_0 است و در غیر این صورت رد H_0 و شاهد نا همگنی واریانس بوده و رگرسیون وزنی خواهد بود.

تحت فرض H_0 :
$$Q = \frac{\sum_{i=1}^{n_1}(U_i-\bar{U})^2/K_1}{\sum_{i=1}^{n_3}(U_i-\bar{U})^2/K_3} \sim F_{(K_1,K_3)}$$

اما در اینجا آماره بین $F_{0.05}$ و $F_{0,95}$ است پس دلیلی برای رد H_0 نیست و ناهمگنی واریانس رد میشود بنابراین رگرسیون غیر وزنی است و مدل رگرسیونی همانند قبل میماند.

```
e1sort <- sort(e1)
k1_1 <- k3_1 <- floor(n1/3)
ek1_1 <- e1sort[1:k1_1]
ek3_1 <- e1sort[-(1:k3_1)]
Q1 <- (sum((ek1_1-mean(ek1_1))^2))/((sum((ek3_1-mean(ek3_1))^2)))
Q1
Falpha01_1 <- qf(0.05, k1_1, k3_1)
Falpha01_1
Falpha02_1 <- qf(0.95, k1_1, k3_1)
Falpha02_1
```

خروجی به صورت زیر چاپ میشد:

```
> e1sort <- sort(e1)
> k1_1 <- k3_1 <- floor(n1/3)
> ek1_1 <- e1sort[1:k1_1]
> ek3_1 <- e1sort[-(1:k3_1)]
> Q1 <- (sum((ek1_1-mean(ek1_1))^2))/((sum((ek3_1-mean(ek3_1))^2)))
> Q1
[1] 0.9844443
> Falpha01_1 <- qf(0.05, k1_1, k3_1)
> Falpha01_1
[1] 0.9229429
> Falpha02_1 <- qf(0.95, k1_1, k3_1)
> Falpha02_1
[1] 1.083491
>
```

آزمون معنا داری مدل رگرسیونی:

مطابق جدول تجزیه واریانس مقادیر SSR1,MSR1,SSE1,MSE1 همچون تصویر زیر با کد نویسی در RStudioبه دست میاوریم.

ANOVA					
	df	**SS**	**MS**	***F***	***Significance F***
Regression	*k*	SSR	MSR=SSR/*k*	MSR/MSE	*P*-value of the *F* Test
Residuals	*n−k−1*	SSE	MSE= SSE/(n−k−1)		
Total	*n−1*	SST			

$$\begin{cases} H_0: & \beta_1 = \beta_2 = \cdots = \beta_j = 0 \\ H_1: & \beta_1 \neq \beta_2 \neq \ldots \neq \beta_j \neq 0 \end{cases}$$

$$F_0|_{H_0}, \tilde{X} = \tilde{x} \sim F_{(k\,,n-k-1)}$$

ناحیه رد در سطح معنی دار α : $F_0 > F_{(\alpha\,,\ k\ ,\ n-k-1)}$

تعداد داده های برآورد شده: p1

آماره آزمون: F0_1

اگر F0_1 از Falpha1 بزرگ تر باشد ،فرض بی معنا بودن آزمون رد میشود.

```
p1 <- ncol(x1)-1
SSR1 <- t(betahat1)%*%t(x1)%*%y1 - (sum(y1)^2/n1)
SSR1
MSR1 <- SSR1/p1
MSR1
SSE1 <- t(y1)%*%y1 - t(betahat1)%*%t(x1)%*%y1
SSE1
MSE1 <- SSE1/(n1-p1-1)
MSE1
F0_1 <- MSR1/MSE1
F0_1
Falpha1 <- qf(0.95, p1, n1-p1)
Falpha1
```

و نتایج خروجی به شکل زیر خواهد شد:

حال با مقایسه F0_1 و Falpha1 درسطح α=۰.۰۵ ، با توجه به اینکه مقدار F0_1 از Falpha1 بزرگ تر است،فرض بی معنا بودن مدل با قاطعیت رد شده و مدل معنا دار خواهد بود.(۱۰۹۱۹۶.۱>۲.۳۷۳۶۹۲)

```
> p1 <- ncol(x1)-1
> SSR1 <- t(betahat1)%*%t(x1)%*%y1 - (sum(y1)^2/n1)
> SSR1
         [,1]
[1,] 1831552
> MSR1 <- SSR1/p1
> MSR1
        [,1]
[1,] 457888
> SSE1 <- t(y1)%*%y1 - t(betahat1)%*%t(x1)%*%y1
> SSE1
         [,1]
[1,] 21163.4
> MSE1 <- SSE1/(n1-p1-1)
> MSE1
          [,1]
[1,] 4.193263
> F0_1 <- MSR1/MSE1
> F0_1
          [,1]
[1,] 109196.1
> Falpha1 <- qf(0.95, p1, n1-p1)
> Falpha1
[1] 2.373692
> ++
```

آزمون معنا داری ضرایب انفرادی رگرسیون:

$$\begin{cases} H_0: \beta_j = 0 \\ H_1: \beta_j \neq 0 \end{cases}$$

$$T=\frac{\hat{\beta}_j}{S\cdot E(\hat{\beta}_j)} \quad , \quad S\cdot E(\hat{\beta}_j) = \hat{\sigma}\sqrt{c_{jj}}$$

$$T|_{H_0}, \tilde{X} = \tilde{x} \sim t_{(n-p)}$$

ناحیه رد در سطح معنی دار :

$$|T| \geq t_{1-\frac{\alpha}{2}}(n-p)$$

در نرم افزار R :

آماره آزمون: T0_1

در ماتریس $(X'X)^{-1}$ درایه های قطر اصلی نمایانگر واریانس β ها خواهد بود و هدف از تشکیل ماتریس همین مورد است.

```
varbeta1 <- solve(t(x1)%*%x1)
varbeta1
```

```
> varbeta1 <- solve(t(x1)%*%x1)
> varbeta1
              [,1]          [,2]          [,3]          [,4]          [,5]
[1,]  7.412623e-03 -1.489528e-04 -4.495161e-05 -4.392282e-04 -1.037428e-04
[2,] -1.489528e-04  2.921495e-05  9.534290e-08 -1.918138e-07 -4.337585e-07
[3,] -4.495161e-05  9.534290e-08  6.603075e-07 -1.536461e-07 -6.043409e-08
[4,] -4.392282e-04 -1.918138e-07 -1.536461e-07  6.846396e-05  2.300843e-07
[5,] -1.037428e-04 -4.337585e-07 -6.043409e-08  2.300843e-07  2.388429e-05
```

در این قسمت با مقایسه Tbeta0_1 , Tbeta1_1 , Tbeta2_1, Tbeta3_1 , Tbeta4_1 با آماره آزمون در سطح α=۰.۰۵ ، معنا داری ضرایب به صورت انفرادی مورد بررسی قرار میگیرد.

```
Tbeta0_1 <- abs(betahat1[1]/(varbeta1[1,1]^0.5))
Tbeta0_1
Tbeta1_1 <- abs(betahat1[2]/(varbeta1[2,2]^0.5))
Tbeta1_1
Tbeta2_1 <- abs(betahat1[3]/(varbeta1[3,3]^0.5))
Tbeta2_1
Tbeta3_1 <- abs(betahat1[4]/(varbeta1[4,4]^0.5))
Tbeta3_1
Tbeta4_1 <- abs(betahat1[5]/(varbeta1[5,5]^0.5))
Tbeta4_1
T0_1 <- qt(0.95, n1-p1)
T0_1
```

با توجه به اینکه تمامی مقادیر Tbeta0_1 , Tbeta1_1 , Tbeta2_1, Tbeta3_1 , Tbeta4_1 از آماره آزمون بیشتر است پس معنا دار نبودن همه ضرایب رد شده و ضرایب در حد بسیار خوبی معنا دار هستند.

```
> Tbeta0_1
[1] 395.4313
> Tbeta1_1
[1] 525.6505
> Tbeta2_1
[1] 1253.23
> Tbeta3_1
[1] 58.4026
> Tbeta4_1
[1] 40.35868
> T0_1
[1] 1.645156
```

تحلیل جدول **Model Summary :**

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.994[a]	.988	.988	2.061

a. Predictors: (Constant), Sample Question Papers Practiced, Sleep Hours, Previous Scores, Hours Studied

ضریب تعیین (Coefficient of Determination) :

$$R^2 = \frac{SSR}{SST} = \frac{SST - SSE}{SST} = 1 - \frac{SSE}{SST}$$

R Square = ۰.۹۸۸ یا ٪۹۸.۸

مقدارهای نزدیک به یک، برازش بهتر و همچنین سهم بیشتر در بیان تغییرات متغیر وابسته را نشان می دهد می توان گفت ٪۹۹.۸ عملکرد تحصیلی دانش آموزان بستگی به این ۵ متغیر مستقل دارد که ما در این مدل در نظر گرفتیم و ٪۰.۲ بستگی به سایر متغیر ها که ما در مدل خود نیاورده ایم.

ضریب تعیین تعدیل شده :

$$R^2_{adjusted} = R^2_{adj} = 1 - \left(\frac{n-1}{n-p}\right)(1 - R^2_P) = 1 - \frac{(1-R^2)(n-1)}{n-p-1}$$

$$R^2_{adjusted} = R^2_{adj} = 0.988 \quad \text{یا} \quad 98.8\%$$

$$R^2_P = \frac{SSR(P)}{SST}$$

ضریب تعیین اصلاح یا تعدیل شده است. همانطور که متغیرهای مستقل یا پیش‌گو به مدل اضافه می‌شوند، ضریب تعیین افزایش یافته و به نظر مدل بهتری حاصل می‌شود. می‌توان با اضافه کردن متغیرهای مستقل به مدل ادامه داد تا جایی که توانایی مدل در توصیف متغیر وابسته بهبود یابد. البته این امر به پیچیده شدن مدل رگرسیونی منجر می‌شود. اگر چه افزودن متغیر مستقل به مدل باعث افزایش مقدار ضریب تعیین می‌شود ولی ممکن است این امر به علت تغییرات تصادفی یا شانسی حاصل از نمونه‌ها رخ داده باشد.

نزدیکی این دو مقدار به هم نشانگر آن است که متغیرهای به کار رفته در مدل، توانسته‌اند به خوبی به کار آیند و برازش مناسبی ارائه دهند.

با اضافه کردن متغیر به مدل : $SSE \downarrow \quad \Longrightarrow \quad SSR \uparrow$

ضریب تعیین زیاد می شود. $\Longrightarrow \quad R^2 = \frac{SSR}{SST}$

$R^2 \uparrow$

ضریب تعیین تعدیل شده $\downarrow$ $\bar{R}^{\,2} = R^2_{adj} = 1 - \frac{\frac{SSE}{n-p}}{\frac{SST}{n-1}}$ ، $p \uparrow \Longrightarrow n-p \downarrow \Longrightarrow \frac{SSE}{n-p} \uparrow$

کاهش می یابد.

ضریب همبستگی پیرسون : 99.4% یا $R = 0.994$

همبستگی خطی بین متغیر های مستقل X و متغیر وابسته Y در واقع یک عدد زیادی است و میزان همبستگی زیاد بین آنها را بیان می کند.

خطای استاندارد براورد (Std. Error of the Estimate): برابر ۲،۰۶۱ که به آن میانگین ریشه مربع خطا نیز می‌گویند. در حقیقت این مقدار، انحراف معیار اصطلاح خطا است و ریشه مربعات باقیمانده (یا خطا) را نشان می‌دهد. از این مقدار برای برآورد واریانس متغیر وابسته نیز می‌توان استفاده کرد. در مورد ارزیابی دو مدل، با ضرایب تعیین تقریبا یکسان، مدلی انتخاب می‌شود که خطای استاندارد مقادیر خطا (باقیمانده) کمتری داشته باشد.

در ادامه برای داده های کیفی yes با عنوان data_2 تمام مراحلی که برای داده های data_1 انجام دادیم را انجام می دهیم.

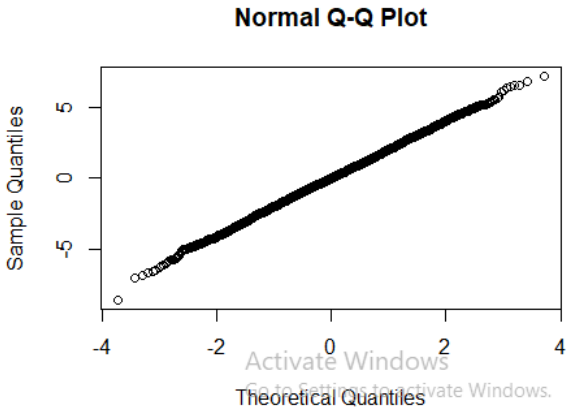
```
data_2 <- subset(data_frame, data_frame[, 3] == 2)
n2 <- nrow(data_2)
n2
y2 <- data_2[, 6]
y2
x2_0 <- rep(1, n2)
x2_1 <- data_2[, 1]
x2_2 <- data_2[, 2]
x2_3 <- data_2[, 4]
x2_4 <- data_2[, 5]
x2 <- matrix(c(x2_0, x2_1, x2_2, x2_3, x2_4), ncol = 5)
x2
```

```
> n2 <- nrow(data_2)
> n2
[1] 4948
> y2 <- data_2[, 6]
> y2
   [1]  91  45  36  63  42  67  70  30  71  73  49  83  74  74  39  36  58  47  60  74  32  39  58
  [24]  71  54  17  58  27  65  52  33  47  70  98  87  49  41  61  54  81  52  65  36  35  15  88
  [47]  49  33  60  81  58  38  60  76  69  81  36  25  61  76  83  50  38  82  23  56  43  30  92
  [70]  82  71  86  68  44  68  47 100  23  60  33  47  31  58  18  36  58  45  60  56  42  51  57
  [93]  67  27  74  38  28  54  56  32  29  57  38  18  27  33  37  77  45  73  43  39  54  62  63
 [116]  89  62  88  35  60  38  56  47  43  43  46  60  47  60  53  23  81  71  75  89  60  54  41
 [139]  47  71  42  21  48  38  49  81  52  77  33  67  76  37  76  82  41  75  34  69  40  63  94
 [162]  79  50  22  60  58  78  51  29  22  45  70  38  82  87  77  72  40  72  18  64  66  44  90
 [185]  64  62  62  26  30  77  62  82  73  40  39  73  46  80  56  37  88  32  53  41  63  17  24
 [208]  47  43  48  17  54  20  34  63  72  21  83  51  83  49  85  95  56  50  23  26  46  73  45
 [231]  62  21  55  22  70  64  42  55  34  78  71  33  46  86  34  26  44  61  40  33  20  66  38
 [254]  25  43  16  48  82  81  24  81  54  25  53  40  29  73  27  40  83  69  72  77  32  44  29
 [277]  74  67  34  46  41  84  60  65  21  77  63  68  66  36  29  31  45  65  38  48  59  58  62
 [300]  56  40  65  48  24  51  81  43  43  47  52  41  49  37  61  77  75  40  46  42  75  40  42
 [323]  34  37  34  31  92  44  66  56  60  74  52  84  47  71  86  72  55  54  83  72  86  60  54
 [346]  45  68  72  17  73  47  63  63  38  60  39  67  48  44  37  41  54  74  50  41  65  65  79
 [369]  42  52  17  53  39  44  36  30  80  77  64  78  74  76  53  46  61  46  86  24  59  29  84
 [392]  46  25  62  50  54  83  62  45  65  61  41  35  92  49  82  80  41  73  61  43  75  72  92
 [415]  71  72  61  27  49  81  45  50  70  27  27  54  32  44  33  58  68  66  60  88  78  43  69
 [438]  47  72  51  73  28  60  66  21  32  37  96  51  82  64  50  73  20  49  70  28  82  42  68
 [461]  67  39  39  38  69  88  64  70  72  40  77  60  82  77  61  73  64  33  78  34  60  26  70
 [484]  85  30  48  79  75  59  49  48  63  71  54  32  23  52  26  26  74  52  40  54  51  45  56
 [507]  65  38  17  35  26  23  67  31  57  22  35  79  58  17  70  36  71  63  78  39  70  34  34
 [530]  67  68  85  56  62  99  46  86  73  37  65  53  55  47  63  27  53  56  55  67  32  73  43
 [553]  32  42  66  46  83  63  63  43  47  22  72  73  65  60  29  74  71  46  99  40  57  58  55
 [576]  46  45  65  35  45  35  63  35  63  74  78  55  84  60  34  54  56  82  45  43  57  56  75
 [599]  33  26  41  34  68  44  75  56  37  67  31  82  58  40  36  43  81  84  64  41  53  65  31
```

```
> x2_0 <- rep(1, n2)
> x2_1 <- data_2[, 1]
> x2_2 <- data_2[, 2]
> x2_3 <- data_2[, 4]
> x2_4 <- data_2[, 5]
> x2 <- matrix(c(x2_0, x2_1, x2_2, x2_3, x2_4), ncol = 5)
> x2
      [,1] [,2] [,3] [,4] [,5]
 [1,]    1    7   99    9    1
 [2,]    1    8   51    7    2
 [3,]    1    5   52    5    2
 [4,]    1    7   73    5    6
 [5,]    1    8   45    4    6
 [6,]    1    8   73    8    4
 [7,]    1    6   83    7    2
 [8,]    1    2   54    4    9
 [9,]    1    1   99    4    3
[10,]    1    9   74    7    6
[11,]    1    7   62    7    4
[12,]    1    9   84    6    6
[13,]    1    3   94    6    5
[14,]    1    5   90    4    3
[15,]    1    3   61    7    3
[16,]    1    7   44    9    1
[17,]    1    5   70    6    9
[18,]    1    9   52    8    1
[19,]    1    7   67    9    3
[20,]    1    2   97    9    4
[21,]    1    2   55    4    1
[22,]    1    2   63    6    0
[23,]    1    4   73    7    0
[24,]    1    8   77    6    4
[25,]    1    3   76    4    3
```

```
betahat2 <- solve(t(x2)%*%x2)%*%(t(x2)%*%y2)
betahat2
y2hat <- x2%*%betahat2
y2hat
e2 <- y2-y2hat
e2
qqnorm(e2)
```

```
> betahat2 <- solve(t(x2)%*%x2)%*%(t(x2)%*%y2)
> betahat2
             [,1]
[1,] -33.4927053
[2,]   2.8653001
[3,]   1.0184543
[4,]   0.4780133
[5,]   0.1902356
```




```
 [2,] -0.0974274047
 [3,] -0.5639549808
 [4,] -1.4430373435
 [5,]  3.6863956845
 [6,] -1.3619060135
 [7,] -1.9573642227
 [8,] -0.8585993148
 [9,] -1.6823282045
[10,]  0.8518817271
[11,] -4.8155956212
[12,]  1.1453522121
[13,] -0.6571546627
[14,] -0.9774399035
[15,] -2.1457055332
[16,]  0.1312616536
[17,]  1.2942055214
[18,] -2.2689594079
[19,]  0.3263420391
[20,] -2.0910216658
```

```
 [2,] 45.09743
 [3,] 36.56395
 [4,] 64.44304
 [5,] 38.31360
 [6,] 68.36191
 [7,] 71.95736
 [8,] 30.85860
 [9,] 72.68233
[10,] 72.14812
[11,] 53.81560
[12,] 81.85465
[13,] 74.65715
[14,] 74.97744
[15,] 41.14571
[16,] 35.86874
[17,] 56.70579
[18,] 49.26896
[19,] 59.67366
[20,] 76.09102
```

```
e2sort <- sort(e2)
k1_2 <- k3_2 <- floor(n2/3)
ek1_2 <- e1sort[1:k1_2]
ek3_2 <- e1sort[-(1:k3_2)]
Q2 <- (sum((ek1_2-mean(ek1_2))^2))/((sum((ek3_2-mean(ek3_2))^2)))
Q2
Falpha01_2 <- qf(0.1, k1_2, k3_2)
Falpha01_2
Falpha02_2 <- qf(0.9, k1_2, k3_2)
Falpha02_2
```

```
> e2sort <- sort(e2)
> k1_2 <- k3_2 <- floor(n2/3)
> ek1_2 <- e1sort[1:k1_2]
> ek3_2 <- e1sort[-(1:k3_2)]
> Q2 <- (sum((ek1_2-mean(ek1_2))^2))/((sum((ek3_2-mean(ek3_2))^2)))
> Q2
[1] 0.9698233
> Falpha01_2 <- qf(0.1, k1_2, k3_2)
> Falpha01_2
[1] 0.9388184
> Falpha02_2 <- qf(0.9, k1_2, k3_2)
> Falpha02_2
[1] 1.065169
```

آزمون معنا داری مدل رگرسیون:

```
p2 <- ncol(x2)-1
SSR2 <- t(betahat2)%*%t(x2)%*%y2 - (sum(y2)^2/n2)
SSR2
MSR2 <- SSR2/p2
MSR2
SSE2 <- t(y2)%*%y2 - t(betahat2)%*%t(x2)%*%y2
SSE2
MSE2 <- SSE2/(n2-p2-1)
MSE2
F0_2 <- MSR2/MSE2
F0_2
Falpha2 <- qf(0.95, p2, n2-p2)
Falpha2
```

```
> p2 <- ncol(x2)-1
> SSR2 <- t(betahat2)%*%t(x2)%*%y2 - (sum(y2)^2/n2)
> SSR2
         [,1]
[1,] 1815580
> MSR2 <- SSR2/p2
> MSR2
       [,1]
[1,] 453895
> SSE2 <- t(y2)%*%y2 - t(betahat2)%*%t(x2)%*%y2
> SSE2
          [,1]
[1,] 20339.28
> MSE2 <- SSE2/(n2-p2-1)
> MSE2
          [,1]
[1,] 4.114764
> F0_2 <- MSR2/MSE2
> F0_2
          [,1]
[1,] 110308.9
> Falpha2 <- qf(0.95, p2, n2-p2)
> Falpha2
[1] 2.373729
```

آزمون معنا داری ضرایب انفرادی رگرسیون:

```
varbeta2 <- solve(t(x2)%*%x2)
varbeta2
```

```
> varbeta2 <- solve(t(x2)%*%x2)
> varbeta2
              [,1]          [,2]          [,3]          [,4]          [,5]
[1,]  7.747749e-03 -1.517200e-04 -4.722115e-05 -4.623851e-04 -1.080118e-04
[2,] -1.517200e-04  3.051483e-05  1.458967e-08  6.670160e-08 -5.182402e-07
[3,] -4.722115e-05  1.458967e-08  6.707486e-07  7.471220e-08 -3.407959e-09
[4,] -4.623851e-04  6.670160e-08  7.471220e-08  7.081138e-05 -5.980252e-07
[5,] -1.080118e-04 -5.182402e-07 -3.407959e-09 -5.980252e-07  2.482513e-05
```

```
T0_2 <- qt(0.95, n2-p2)
T0_2
Tbeta0_2 <- abs(betahat2[1]/(varbeta2[1,1]^0.5))
Tbeta0_2
Tbeta1_2 <- abs(betahat2[2]/(varbeta2[2,2]^0.5))
Tbeta1_2
Tbeta2_2 <- abs(betahat2[3]/(varbeta2[3,3]^0.5))
Tbeta2_2
Tbeta3_2 <- abs(betahat2[4]/(varbeta2[4,4]^0.5))
Tbeta3_2
Tbeta4_2 <- abs(betahat2[5]/(varbeta2[5,5]^0.5))
Tbeta4_2
```

```
> T0_2 <- qt(0.95, n2-p2)
> T0_2
[1] 1.645162
> Tbeta0_2 <- abs(betahat2[1]/(varbeta2[1,1]^0.5))
> Tbeta0_2
[1] 380.5068
> Tbeta1_2 <- abs(betahat2[2]/(varbeta2[2,2]^0.5))
> Tbeta1_2
[1] 518.6981
> Tbeta2_2 <- abs(betahat2[3]/(varbeta2[3,3]^0.5))
> Tbeta2_2
[1] 1243.545
> Tbeta3_2 <- abs(betahat2[4]/(varbeta2[4,4]^0.5))
> Tbeta3_2
[1] 56.80524
> Tbeta4_2 <- abs(betahat2[5]/(varbeta2[5,5]^0.5))
> Tbeta4_2
[1] 38.18089
```

بررسی عملکرد رگرسیون بیزی (در شیب رگرسیون خطی ساده) بصورت نمونه‌ای:

برای بررسی تاثیر عملکرد بیزی با فرض آگاهی از اطلاعات جامعه داده‌های Student_Performance به عنوان جامعه مورد بررسی پس از انتخاب بهترین متغیر پیشگو برای مدل خطی ساده با ایجاد یک اطلاع پیشین و یک نمونه تاثیر اطلاع پیشن در دقت برآورد پارامتر شیب خط مدل رگرسیونی در مقایسه با مقدار این پارامتر در جامعه بررسی میشود.

در این قسمت نحوه محاسبه شیب خط مدل رگرسیونی (β) مطابق با عملکرد رگرسیون بیز آورده میشود:

$$\beta = \left(\frac{n_{.}\beta_{.} + n\hat{\beta}}{n_{.} + n}\right), \qquad n_{.} = \frac{(\frac{\sigma^{2}}{\sigma^{2}{}_{x}})}{\sigma^{2}{}_{.}}, \qquad Var(\beta) = \frac{(\frac{\sigma^{2}}{\sigma^{2}{}_{x}})}{n_{.} + n}$$

که در آن:

β شیب خط مدل رگرسیونی پسین است.

$\beta_{.}$ شیب خط مدل رگرسیونی مطابق با اطلاع پیشین است.

$\hat{\beta}$ شیب خط مدل رگرسیونی مطابق با داده‌های نمونه‌ای است (درستنمایی)

$n_{.}$ تعداد شبه مشاهدات یا همان اطلاع موجود پیشین است.

n تعداد مشاهدات نمونه‌ای است (درستنمایی)

σ^{2} واریانس مقادیر متغیر پاسخ است که چون فرض میشود اطلاعات جامعه موجود است از مقدار آن در جامعه استفاده میشود.

$\sigma^{2}{}_{x}$ واریانس مقادیر متغیر مستقل است که چون فرض میشود اطلاعات جامعه موجود است از مقدار آن در جامعه استفاده میشود.

شایان ذکر است با توجه به متغیر مستقل کیفی Extracurricular Activities که دو وضعیتی است از روند متغیر ظاهری استفاده میشود که در بررسی زیر تنها یکی از وضعیت‌های متغیر فوق یعنی No پس از تبدیل به مقدار کمی ۱ در کل روند استفاده میشود.

در مرحله اول در راستای پیاده سازی مدل خطی ساده پس از بررسی مقادیر ضریب همبستگی بین متغیرهای پیشگو و متغیر پاسخ، متغیر با بیشترین همبستگی برای مدل انتخاب میشود. که روند پیدا کردن آن در قطعه کد زیر (که نوشته شده با زبان برنامه نویسی python است) قابل مشاهده است.

```
import csv
import numpy as np

y = Performance_Index = []
x1 = Hours_Studied = []
x2 = Previous_Scores = []
x3 = Sleep_Hours = []
x4 = Sample_Question_Papers_Practiced =[]
with open("C:\\Users\\Zahra\\Desktop\\پروژه رگرسیون\\Student_Performance.csv", newline='') as csvfile:
    reader = csv.DictReader(csvfile)
    for row in reader:
        if row['Extracurricular Activities'] == "No":
            y.append(float(row['Performance Index']))
            x1.append(float(row['Hours Studied']))
            x2.append(float(row['Previous Scores']))
            x3.append(float(row['Sleep Hours']))
            x4.append(float(row['Sample Question Papers Practiced']))

independent_variables = [x1, x2, x3, x4]
corrcoeff = 0
index = 0
for i in independent_variables:
    tmp = abs(np.corrcoef(y, i)[0, 1])
    if tmp > corrcoeff:
        corrcoeff = tmp
        index = independent_variables.index(i)+1
corrcoeff_result = (round(corrcoeff,2), index)
```

(توضیح: پس از فرخوانی داده‌های فایل و انتصاب مقادیر پیشگو و پاسخ به لیست y و لیست‌های x و تفکیک متغیرظاهری، متغیر مستقلی که بالاترین همبستگی با متغیر پاسخ را دارد در corrcoeff_result ذخیره میشود.)

که خروجی آن مطابق زیر است:

```
In [6]: corrcoeff_result
Out[6]: (0.91, 2)
```

(که مبین آن است که متغیر پیشگو Previous Scores یعنی(Previous Scores) بالاترین همبستگی(۰/۹۱) را دارد)

و در مرحله بعد آن در نرم افزار R:

پس از فراخوانی اطلاعات داده‌های فایل Student_Performance با استفاده از کتابخانه readxl و تفکیک داده‌ها برحسب متغیر کیفی Extracurricular Activities که در قطعه کد زیر قابل مشاهده است:

```
library(readxl)

data_file <- read_excel("C:\\Users\\Zahra\\Desktop\\پروژه رگرسیون\\Student_Performance.xlsx")
data_frame <- data.matrix(data_file)
data_1 <- subset(data_frame, data_frame[, 3] == 1)
```

(تابع matrix مقادیر کیفی No را به ۱ و مقادیر کیفی Yes را به ۲ میبرد)

در **گام اول** جامعه و پارامترهای مدل رگرسیون خطی ساده برحسب متغیر مسقل Previous Scores مورد بررسی واقع میشود. (حرف T که قبل از نام پارامترها و آمارهها آمده مخفف total به منظور بررسی کل جامعه میباشد.)

```
Ty <- data_1[,6]
Ty
```

```
> Ty
   [1] 65 66 61 61 69 84 73 27 33 68 43 63 85 57 35 66 42 68 64 45 36 54 53 75 78 91 78 38 71
  [30] 54 42 91 74 61 45 71 67 95 29 21 30 57 27 34 76 57 45 81 66 56 25 56 46 45 70 36 71 49
  [59] 43 77 34 49 69 84 41 41 58 94 40 36 47 83 36 74 42 26 42 85 33 77 72 53 16 45 49 49 73
  [88] 65 72 67 73 72 42 47 77 49 30 75 78 89 48 27 66 65 29 26 72 41 63 59 46 42 30 77 92 70
 [117] 35 66 81 61 27 77 43 48 19 41 28 52 53 52 64 35 73 47 36 63 58 58 62 37 86 88 38 57 35
```

```
Tx <- data_1[, 2]
Tx
```

```
> Tx
   [1] 82 75 78 77 89 91 79 47 47 79 72 75 96 85 61 79 59 72 68 62 46 73 61 81 93 99 98 48 88
  [30] 60 48 94 77 68 69 80 75 99 52 46 48 64 50 51 99 70 53 89 92 74 40 68 67 60 84 56 79 52
  [59] 71 76 54 51 88 96 54 56 87 96 48 62 72 95 56 96 58 52 66 98 49 90 90 62 43 59 58 66 96
  [88] 73 95 87 89 91 65 65 93 58 48 84 89 93 52 51 71 69 57 41 87 42 75 65 73 47 49 89 94 92
 [117] 57 77 85 78 54 92 63 55 45 52 48 68 76 82 87 54 72 52 62 77 84 74 84 44 96 93 44 71 60
```

```
TSxy <- sum((Tx-mean(Tx))*(Ty-mean(Ty)))
TSxy
```

```
> TSxy
[1] 1532648
```

```
TSxx <- sum((Tx-mean(Tx))^2)
TSxx
```

```
> TSxx
[1] 1516275
```

```
Tbeta <- TSxy/TSxx
Tbeta
```

```
> Tbeta
[1] 1.010798
```

که این عدد مقدار پارامتر شیب برای مدل رگرسیون خطی ساده در جامعه است و هر میزان برآورد به این مقدار نزدیک‌تر باشد دقیق‌تر است.

```
sigma2 <- var(Ty)
sigma2
```

```
> sigma2
[1] 366.8017
```

این مقدار مبین واریانس متغیر پاسخ در جامعه که همان σ^2 است میباشد.

```
sigma2x <- var(Tx)
sigma2x
```

```
> sigma2x
[1] 300.193
```

این مقدار نیز مبین واریانس متغیر مستقل که همان σ_x^2 است میباشد.

در **گام دوم** زیرمجموعه ای از جامعه جدا شده و از پارامترهای مدل رگرسیون خطی ساده آن بعنوان اطلاع پیشین استفاده میشود. (به نحوی که پس از این گام فرض میشود داده‌های این زیرمجموعه در دسترس نیستند و تنها پارامترهای آن بعنوان اطلاع پیشین مورد استفاده قرار میگیرند.)

(حروف PR که قبل از نام پارامترها و آماره‌ها آمده مخفف prior به منظور اطلاع پیشین میباشد.)

```
PRdata_1 <- data_1[1:(nrow(data_1)/2), ]
```

در این بخش اطلاعات نیمه اول داده‌های جامعه که بطور یکنواخت و بدون مرتب سازی در متغیر data_1 پراکنده شده‌اند جدا میشود و اطلاع پیشین برحسب آن بدست می‌آید.

```
PRy <- PRdata_1[, 6]
PRy
```

```
> PRy
  [1] 65 66 61 61 69 84 73 27 33 68 43 63 85 57 35 66 42 68 64 45 36 54 53 75 78 91 78 38 71
 [30] 54 42 91 74 61 45 71 67 95 29 21 30 57 27 34 76 57 45 81 66 56 25 56 46 45 70 36 71 49
 [59] 43 77 34 49 69 84 41 41 58 94 40 36 47 83 36 74 42 26 42 85 33 77 72 53 16 45 49 49 73
 [88] 65 72 67 73 72 42 47 77 49 30 75 78 89 48 27 66 65 29 26 72 41 63 59 46 42 30 77 92 70
[117] 35 66 81 61 27 77 43 48 19 41 28 52 53 52 64 35 73 47 36 63 58 58 62 37 86 88 38 57 35
```

```
PRx <- PRdata_1[, 2]
PRx
```

```
> PRx
  [1] 82 75 78 77 89 91 79 47 47 79 72 75 96 85 61 79 59 72 68 62 46 73 61 81 93 99 98 48 88
 [30] 60 48 94 77 68 69 80 75 99 52 46 48 64 50 51 99 70 53 89 92 74 40 68 67 60 84 56 79 52
 [59] 71 76 54 51 88 96 54 56 87 96 48 62 72 95 56 96 58 52 66 98 49 90 90 62 43 59 58 66 96
 [88] 73 95 87 89 91 65 65 93 58 48 84 89 93 52 51 71 69 57 41 87 42 75 65 73 47 49 89 94 92
[117] 57 77 85 78 54 92 63 55 45 52 48 68 76 82 87 54 72 52 62 77 84 74 84 44 96 93 44 71 60
```

```
PRSxy <- sum((PRx-mean(PRx))*(PRy-mean(PRy)))
PRSxy
```

```
> PRSxy
[1] 760104.5
```

```
PRSxx <- sum((PRx-mean(PRx))^2)
PRSxx
```

```
> PRSxx
[1] 754380.6
```

```
PRbeta <- PRSxy/PRSxx
PRbeta
```

```
> PRbeta
[1] 1.007588
```

این همان شیب خط مدل رگرسیونی این داده‌ها است که مقدار آن مقدار پیشین β. است.

```
PRconstant <- mean(PRy)-PRbeta*mean(PRx)
PRyhat <- PRx*PRbeta+PRconstant
PRe <- PRy-PRyhat
PRsigma2 <- var(PRe)/PRSxx
PRsigma2
```

```
> PRsigma2
[1] 7.935851e-05
```

و این مقدار نیز همان واریانس شیب خط مدل رگرسیونی است که مقدار پیشین σ^{2}. است.

در **گام سوم** از زیرمجموعه دوم جامعه که در اطلاع پیشین نیست نمونه ای ۲۰ تایی به روش تصادفی ساده انتخاب میشود که پارامترهای مدل رگرسیون خطی ساده آن مقادیر اطلاع درستنمایی هستند.

(حروف LL که قبل از نام پارامترها و آماره‌ها آمده مخفف likelihood به منظور اطلاع درستنمایی میباشد.)

```
SecondHalfdata_1 <- data_1[(nrow(data_1)/2):nrow(data_1),]
SecondHalfdata_1
sample_numbers <- sample(1:nrow(POSdata_1), 20)
sample_numbers
samples <- matrix(nrow = 0, ncol = 6)
for (i in sample_numbers){
  samples <- rbind(samples,SHdata_1[i,])
}
LLdata <- samples
```

در این بخش اطلاعات نیمه دوم داده‌های جامعه که بطور یکنواخت و بدون مرتب سازی در متغیر data_1 پراکنده شده‌اند جدا میشود و داده‌های ۲۰ سطر آن بصورت تصادفی ساده انتخاب میشوند.)

```
LLy <- POSdata[, 6]
LLy
```

```
> LLy
 [1] 61 62 54 86 51 62 66 56 32 75 36 39 27 46 70 53 45 72 78 74
```

```
LLx <- POSdata[, 2]
LLx
```

```
> LLx
 [1] 70 69 80 95 68 73 68 72 51 94 58 53 52 45 82 75 70 94 81 93
```

```
LLSxy <- sum((POSx-mean(POSx))*(POSy-mean(POSy)))
LLSxy
```

```
> LLSxy
[1] 4084.25
```

```
LLSxx <- sum((POSx-mean(POSx))^2)
LLSxx
```

```
> LLSxx
[1] 4368.55
```

```
LLbeta <- POSSxy/POSSxx
LLbeta
```

```
> LLbeta
[1] 0.9349212
```

این مقدار همان شیب خط مدل رگرسیونی این داده‌های نمونه‌ای است که مقدار آن مقدار درستنمایی $\hat{\beta}$ است.

در گام آخر n. اطلاع موجود در پیشین (تعداد شبه مشاهدات) و پس از آن میانگین شیب خط مدل رگرسیونی β. پیشین و $\hat{\beta}$ درستنمایی محاسبه و مقداری برای واریانس این براورد لحاظ یعنی β پسین از ترکیب موزون میشود.

(حروف POS که قبل از نام پارامترها آمده مخفف posterior به منظور اطلاع پسین میباشد.)

```
n0 <- (sigma2/sigma2x)/PRsigma2
n0
```

```
> n0
[1] 15397.04
```

```
POSbetaav <- (n0*PRbeta+20*LLbeta)/(n0+20)
POSbetaav
```

```
> POSbetaav
[1] 1.007493
```

این مقدار همان شیب خط مدل رگرسیونی پسین داده‌ها یعنی β است که مقدار آن بسیار نزدیک به مقدار شیب خط مدل برای کل داده‌های جامعه فرض شده است

```
POSbetavar <- (sigma2/sigma2x)/(n0+20)
POSbetavar
```

```
> POSbetavar
[1] 7.925556e-05
```

این مقدار نیز همان واریانس شیب خط مدل رگرسیونی پسین داده‌ها است که مقداری بسیار کوچک و قابل اغماض است.

```
POSbetaSE <- POSbetavar^0.5
POSbetaSE
```

```
> POSbetaSE
[1] 0.008902559
```

در این قسمت آخر یک فاصله اطمینان در سطح معنی دار ۹۵ درصد برای شیب خط پسین مدل رگرسیونی یعنی β محاسبه میکنیم که دربردارنده مقدار این شیب در جامعه مورد بررسی است.

نحوه محاسبه آن شایان ذکر است که در زیر آورده شده است:

$$\beta \in \left(\frac{n_.\beta_. + n\hat{\beta}}{n_. + n}\right) \pm \frac{\frac{\sigma}{\sigma_x}}{\sqrt{n_. + n}}$$

```
LPOSbeta <- POSbetaav-1.96*POSbetaSE
LPOSbeta
```

```
> LPOSbeta
[1] 0.9900443
```

```
UPOSbeta <- POSbetaav+1.96*POSbetaSE
UPOSbeta
```

```
> UPOSbeta
[1] 1.024942
```

که این دو مقدار کران‌های پایین و بالا برای این فاصله اطمینان‌اند، بعبارت دیگر داریم:

$$\beta \in (۰.۹۹۰۰۴۴, ۱.۰۲۴۹۴۲)$$