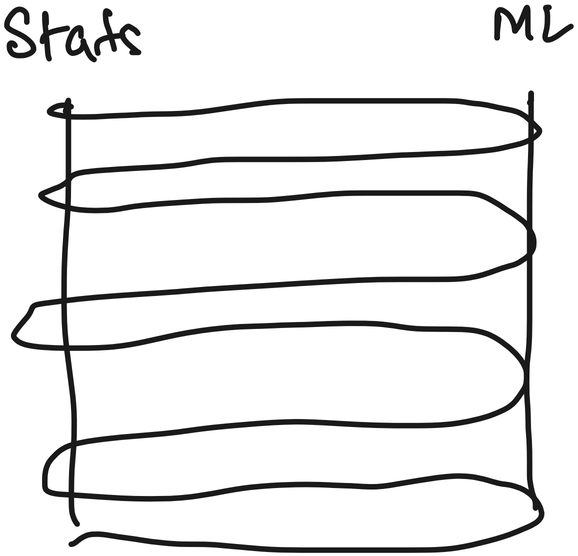
Simple Linear Rgression (SLR)

Multiple Linear Regression (MLR)

Polynomial Regression

Regression Outcomes & Interpretation



Understanding ML requires good understan-ding of Stats

Levels of Measurement.

NOMINAL
- Red, Blue, Green
- Gender

ORDINAL
- Grades

INTERVAL.
- Temp.

RATIO — ht, wt

height, weight
temperature
Gender
Grades.
Salary
Price
Qty.

Level of Measurement	Description	Typical Statistics/Parameters	Supported Operations
Nominal	Categorical data without any order	- Frequency counts - Mode	None
Ordinal	Categorical data with a meaningful order	- Frequency counts - Mode - Median	- Comparison of order (e.g., >, <, =)
Interval	Numeric data with equal intervals but no true zero	- Frequency counts - Mode - Median - Mean - Standard deviation - Correlation (e.g., Pearson's)	- Addition - Subtraction
Ratio	Numeric data with equal intervals and a true zero	- Frequency counts - Mode - Median - Mean - Standard deviation - Range - Ratio comparisons - Geometric mean - Harmonic mean	- Addition - Subtraction - Multiplication - Division

Level of Measurement	~~ML~~ Class of ML Probs.
Nominal, Ordinal (DS)	Classification
Interval, Ratio	Regression

Logistic Regression.

Linear Regression.

ML algorithms. can be used for both

- Regression
- classification

...

$Y = f(X)$

If y is available $\rightarrow$ Supervised ML.

Regression

classification

If y is unaavailable $\rightarrow$ Unsupervised ML.

More discovery necessary.

clustering

REGRESSION.

ML

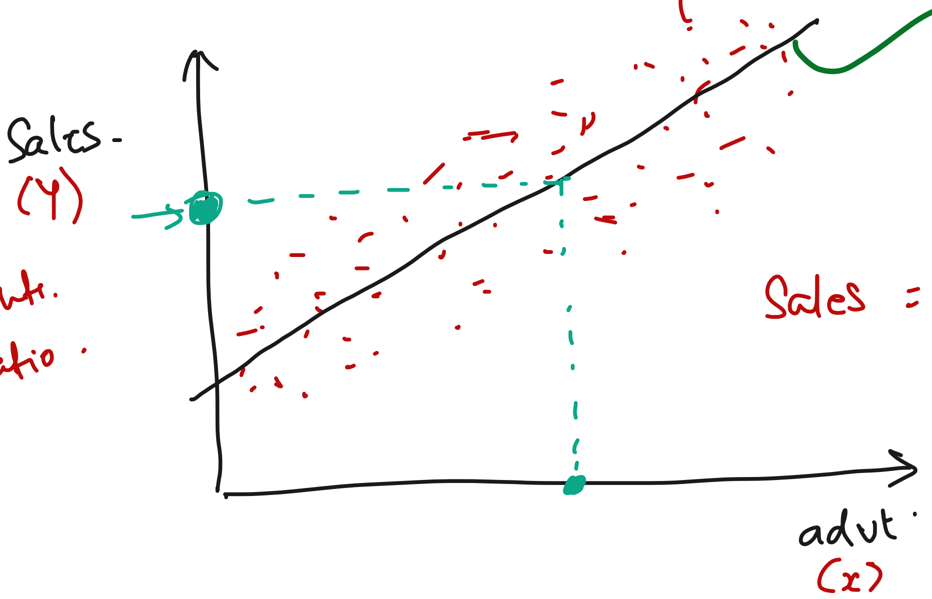
Predicted | Response / output

Inputs | features | predictors

$-\ Y = f(X)$

Prediction 'line' / curve.

Scatter plot.



Cont.
Ratio.

Cont / Ratio

Sales $= f($advt, pop, earnings, edu, weather, connectivity$)$.

$x_1 \quad x_2 \quad x_3 \quad x_4$

$[x_1\ x_2\ x_3\ x_4 \dots] \rightarrow [X]$,

CLASSIFICATION

Scatter Plot

Exercise - 1

- In your spreadsheet explore how to enable the 'Data Analysis' toolpak, or equivalent, to perform statistical calculations!
- Use the uploaded data set data-set-for-SLR-2025.csv to perform Simple Linear Regression and generate the output (for this exercise, use the entire data for training the model)
- Analyze the output generated
- Now, using the **LinearRegression** (sklearn) function of Python create a regression model and calculate metrics like R2, MAE, RMSE and analyze the results
- Further, use the **OLS** function from 'statsmodels' package to perform regression, print the results and review them.

These steps need to be completed before proceeding!

	A	B	C	D	E	F	G	H	I	J	K	L
1	y	x				SUMMARY OUTPUT						
2	7.238462	0.025641										
3	6.310256	0.051282				*Regression Statistics*						
4	8.315385	0.076923				Multiple R	0.906270151					
5	4.787179	0.102564				R Square	0.821325586					
6	5.592308	0.128205				Adjusted R Square	0.819483582					
7	7.830769	0.153846				Standard Error	1.882513522					
8	9.902564	0.179487				Observations	99					
9	5.607692	0.205128										
10	5.146154	0.230769				ANOVA						
11	4.784615	0.25641					*df*	*SS*	*MS*	*F*	*Significance F*	
12	7.05641	0.282051				Regression	1	1580.159507	1580.16	445.8869	4.72338E-38	
13	9.394872	0.307692				Residual	97	343.7541446	3.543857			
14	6.8	0.333333				Total	98	1923.913652				
15	4.871795	0.358974										
16	5.376923	0.384615					*Coefficients*	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
17	10.71538	0.410256			'b'	Intercept	5.922586409	0.381284372	15.53325	4.65E-28	5.165842474	6.679330348
18	11.55385	0.435897			'a'	x	5.452241187	0.258203842	21.11603	4.72E-38	4.939778036	5.964704338
19	9.258974	0.461538										
20	8.097436	0.487179										
21	12.10256	0.512821										

OUTPUTS CREATED BY REGRESSION TOOLS / FUNCTIONS

EXCEL (or any other spreadsheet)

(R^2)

What do these numbers indicate?

What is a good model?

– One that explains most of the variations in the data.

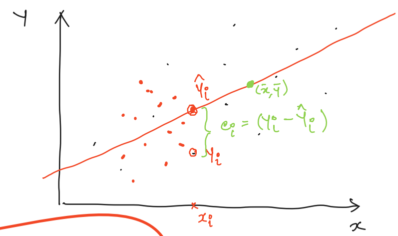
$\sum (y_i - \bar{y})^2 = SST$ (SST = measure of total variation in the given dataset)

$\sum (y_i - \hat{y}_i + \hat{y}_i - \bar{y})^2$

$\sum \left[(y_i - \hat{y}_i)^2 + (\hat{y}_i - \bar{y})^2 + 2 (y_i - \hat{y}_i) \cdot (\hat{y}_i - \bar{y}) \right]$

$\sum (y_i - \hat{y}_i)^2 + \sum (\hat{y}_i - \bar{y})^2 + 2 \sum (\ )(\ )$

$SST = SSE + SSR + \underbrace{2\sum(\ )(\ )}_{'0'}$



Work this out and confirm for yourself using the data data set already with you

SSR => total variation explained by the regression model

SSE => variation NOT explained by the model, attributed to random errors

$SST = SSR + SSE$ + ZERO

$1 = \frac{SSR}{SST} + \frac{SSE}{SST}$

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

R^2 = COEFFICIENT OF DETERMINATION (C.O.D).

= Square of the correlation coefficient 'r' between x & y.

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

R^2 is the square of the correlation Coefficient 'r' } S.L.R

SELF STUDY

– CORRELATION

– CORRELATION COEFF

$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{(\ \cdots \ )}$ ~ Correlation.

We usually calculate SSE, MSE, RMSE, MAE, R2 as metrics reflecting the quality of Linear Regression. However, when we use built-in LR functionality, in tools like Excel, many more numbers are generated .. as shown below. What are they and how to interpret / use them?

ANOVA

	df	SS	MS	F	Significance F
Regression	1	1580.159507	1580.16	445.8869	4.72338E-38
Residual	97	343.7541446	3.543857		
Total	98	1923.913652			

Regression Statistics	
Multiple R	0.906270151
R Square	0.821325586
Adjusted R Square	0.819483582
Standard Error	1.882513522
Observations	99

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%
Intercept (b)	5.922586409	0.381284372	15.53325	4.65E-28	5.165842474	6.679330343
x (a)	5.452241187	0.258203842	21.11603	4.72E-38	4.939778036	5.964704338

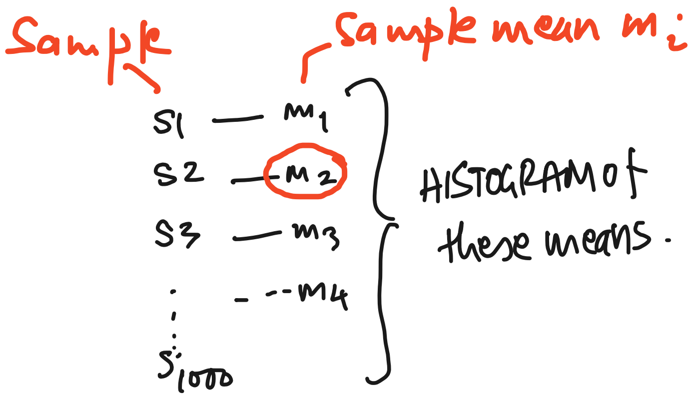
$y = b + ax$ or

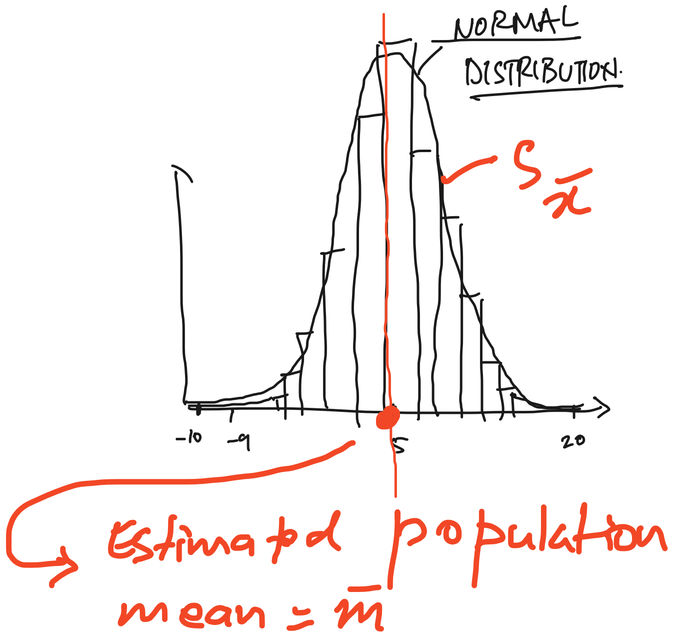
$y = \beta_0 + \beta_1 x$

- To understand these numbers we have to go back to the basics of statistics.
- We need to start with the fact that the (y,x) data that we have is essentially a **sample** (in this case 1 sample of 99 observations)
- We have fitted an LR model using this sample. Therefore the calculated values of **a** and **b** are only an estimate of the population's **actual** a and b.
- Our aim, really, is to predict the value of **y** for an **x** that is not a part of the sample. That is, we need a model that is '**general** and which reflects the reality of the population, and not limited to the sample that we have.
- So we really need to know **how good** an estimate these calculated values (a, b) are. Are they really usable? How much confidence should we have on our calculations?
- This is where we need to understand the concepts, from statistics, of **sampling distributions** and **confidence intervals**

We conduct some 'thought' experiments, related to estimating the population mean from the sample mean:

- Assume that from a population we can take multiple **good**, **representative** samples, let's say **k** samples, each of size **n.** Let's call each sample as s_i
- Using each s_i, we calculate its mean and call it m_i
- Since our samples are **good, representative** samples of the population, they will result in means m_i that are close to each other (why ? try to reason this out)
- If we collect all the m_i and create a frequency table and a histogram, it's shape will be as shown below.





- We will observe that such a histogram indicates that the calculated means m_i tend to have Normal Distribution (as per the **Central Limit Theorem** - see next slide)
- This distribution is known as the **Sampling Distribution of the mean** or **Sampling Distribution of the sample mean** and it has the following properties:
  - The **Expected Value** (ie. mean) of such a distribution is very close to the population mean
  - The Standard Deviation of this distribution - known as the **Standard Error**, and denoted by **S_xbar** - is related to **sigma**, the population's standard deviation in the following way:

$$S_{\bar{x}} = \frac{\sigma}{\sqrt{n}}$$ where n = Size of the sample.

  - Implication of this formula: For a given population, with a given sigma, S_xbar reduces with increase in the sample size **n**. This, in turn, indicates less uncertainty in estimating the true value of the population mean.
  - This appeals to our common sense that **as the sample sizes increase, our analysis becomes more accurate** or, conversely, **smaller sample sizes result in more uncertainty or inaccuracy in our predicted results**

So - given 100 observations, does it make sense to treat it as 1 sample of size 100, or 10 samples of size 10?

The Central Limit Theorem

The Central Limit Theorem (CLT) is a fundamental concept in statistics that describes the distribution of sample means for a sufficiently large sample, regardless of the shape of the original population distribution.

Central Limit Theorem:

For a random sample of size n drawn from any population with a finite mean μ and a finite standard deviation σ, the distribution of the sample means will approach a normal distribution as n becomes sufficiently large. Specifically, as n approaches infinity, the distribution of the sample means will have a mean equal to the population mean ($\mu_{\bar{X}} = \mu$) and a standard deviation equal to the population standard deviation divided by the square root of the sample size ($\sigma_{\bar{X}} = \frac{\sigma}{\sqrt{n}}$).

The Central Limit Theorem is particularly powerful because it allows statisticians to make inferences about population parameters based on the distribution of sample means, **even when the original population distribution is unknown or not normally distributed**. This theorem forms the basis for many statistical techniques and hypothesis tests that rely on the normal distribution.

	A	B	C	D	E	F	G	H	I	J	K	L
1	y	x				SUMMARY OUTPUT						
2	7.238462	0.025641										
3	6.310256	0.051282				*Regression Statistics*						
4	8.315385	0.076923				Multiple R	0.906270151					
5	4.787179	0.102564				R Square	0.821325586					
6	5.592308	0.128205				Adjusted R Square	0.819483582					
7	7.830769	0.153846				Standard Error	1.882513522					
8	9.902564	0.179487				Observations	99					
9	5.607692	0.205128										
10	5.146154	0.230769				ANOVA						
11	4.784615	0.25641					*df*	*SS*	*MS*	*F*	*Significance F*	
12	7.05641	0.282051				Regression	1	1580.159507	1580.16	445.8869	4.72338E-38	
13	9.394872	0.307692				Residual	97	343.7541446	3.543857			
14	6.8	0.333333				Total	98	1923.913652				
15	4.871795	0.358974										
16	5.376923	0.384615					*Coefficients*	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
17	10.71538	0.410256			'b' →	Intercept	5.922586409	0.381284372	15.53325	4.65E-28	5.165842474	6.679330343
18	11.55385	0.435897			'a' →	x	5.452241187	0.258203842	21.11603	4.72E-38	4.939778036	5.964704338
19	9.258974	0.461538										
20	8.097436	0.487179										
21	12.10256	0.512821										

$$y = b + a.x$$

How to interpret these values?

What do they mean?

The problem we are trying to solve:

- We have a dataset (x,y) and we know that it is a **random**, yet **representative sample** (let's call it sample **s1**) of the population
- Based on **s1** we have fitted an SLR model, **y = 5.4522 * x + 5.9226** by **estimating** the coefficients **a1=5.4522, b1=5.9226** (by minimizing SSE)
- If we had a different random sample **s2**, the calculated values **a2, b2** would have been different. In general, a random sample **si** will result in coefficient values **ai, bi** ...
- So, statistically, the calculated coefficient **ai** can possibly assume different values, depending on the **random sample si** that we get for analysis
- There is an important question that needs to be answered: **What is the probability that the calculated value of ai will be close to or equal to ZERO for some of the si?**
- Why is this a critical question?
  - In the regression model **y = a * x + b**, If **a** is **ZERO** then we do not really have a regression model - ie. y cannot really be predicted in terms of x !
- In the context of our example, the calculated value of **a = 5.4522 ..**
  - What is the probability that this value is obtained **by chance**, due to the peculiarity of sample **s1**?
  - What is the probability that other random samples, **si**, will result in values of **ai** that are very close or equal to **ZERO** - thereby making the model invalid?
  - (*BTW, why only **a**? Why are we not much concerned about **b**?*)
- Can we prove, based only on **s1**, that for any other **si** the calculated value of **ai** has a very high probability of being closer to **5.4522**, and almost never close to **ZERO - thereby establishing the validity of the model?**
  - More practically, can we prove, based only on **s1**, that for any other si the calculated value of **ai** has more than 95% probability of being closer to 5.4522, and less than 5% probability of being close to ZERO? (BTW, *why are we talking about 95% and 5%, and not 100% and 0%?*)
  - So, finally, it comes down to whether we can predict the **spread** of the values of **ai** based on just **s1** and **a1**
  - The **Sampling Distribution of *coefficient 'a'*** and the **Central Limit Theorem** help us to calculate this probability and, thus, decide about the validity of the **s1** based Regression model.

We will establish the **sampling distribution** of coefficients **a**,**b** based on s1, a1, b1 as follows:

- Based on **s1** calculate the coefficients **a1 (= 5.4522)** and **b1 (= 5.9226) -** both these are ***statistics*** and it is our goal to check if these values are obtained by chance, or they truly represent the values ***obtainable*** from most other samples as well.
    - The Sampling Distribution of any *statistic* provides us the mechanism to make this assessment
- To establish the Sampling Distributions of coefficients **a** and **b** we need to find out their standard deviations - the **Standard Errors**. In this context, note the following:
    - Our earlier discussion, we covered the **Sampling Distribution of the Sample Mean**. However, now our object of study is **not** the sample mean but the coefficients **a1**, **b1** which are calculated from **s1** using optimization methods. Hence the earlier standard error formula (= sigma/sqrt(n)) does not hold in this case.
- The Standard Error related to coefficients a1 and b1 are given by the formulae:

In the formulae alongside we need the value of 'sigma' the population standard deviation. Wherefrom do we get it?

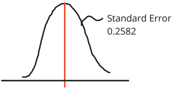
We rely on our sample 's1' for that!

Recollect that s1 is supposed to be representative of the population. Hence, we can assume that the standard deviation of the sample is close to the standard deviation of the population. As we are using this value for estimating the prediction error of a and b, this assumption is acceptable.

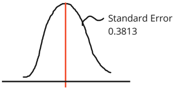
$$SE(a) = \sqrt{\frac{\hat{\sigma}^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}} \qquad SE(b) = \sqrt{\frac{\hat{\sigma}^2}{n}\left(1+\frac{\bar{x}^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2}\right)}$$

$\hat{\sigma}^2$ is the estimated variance of the error term in the model.

- On plugging the sample data (**s1**) into these formulae, the Standard Errors of the sampling distribution of **a** and **b** are 0.2582 and 0.3813, respectively, and we know that both Sampling Distributions follow the Normal Distribution.

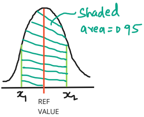


Sampling Distribution of coefficient **a**

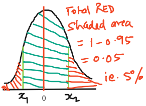


Sampling Distribution of coefficient **b**

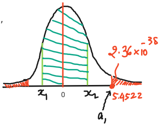
- Lets again recollect that the Sampling Distributions reflect the variations in the *statistics* calculated using different random samples **si**.
- Lets also recollect that the area under the Sampling Distribution curve (formally known as the **Probability Density Function**) represents probability. For example, the shaded area in the figure alongside gives the probability of the values between x1 and x2 - also note that the **total area** under the probability distribution curve is 1.0
- For any REF VALUE (we will use ZERO - why?), and the above calculated STANDARD ERROR values, we can find out the limits x1 and x2 using Normal Distribution Tables / Python functions / Excel formula, etc., such that the **shaded area under the curve is 0.95**.
    - The interval {x1,x2} is said to constitute the **95% Confidence Interval (95% CI)**.
    - The interpretation is that all values between x1 and x2 are **statistically not really different from the REFERENCE VALUE**.
    - Why? - Because, if the population parameter equals the REFERENCE VALUE, and If you were to take 100 random samples (si) from the population, 95 of those samples will result in the calculated value of the ***statistic*** lying between x1 and x2.



- Now we come back to our critical question, posed earlier, and ask it in the form of the following **Statistical Hypothesis: "Is a1 statistically different from ZERO"**
    - We want to check if the statistic **a1** is significantly different from ZERO
    - **Hence we choose the reference value to be ZERO**,
    - **Now, does the 95% CI {x1, x2} include the calculated value of the coefficient a1 (ie. 5.4522)**?
    - If YES, then the calculated value 5.4522 is **obtained by chance** and it is **NOT statistically different from ZERO** - hence our regression model is not valid.
    - If NO, then we can confidently say the following:
        - That the calculated value 5.4522 is SIGNIFICANTLY DIFFERENT from ZERO.
        - That, the value of 5.4522 is not resulting from chance because of the peculiarity of the specific sample s1
        - That 95 out of 100 random data samples would also have resulted in a value of ai closer to 5.4522.
    - All this is equivalent to checking whether the calculated value (eg. a1 = 5.4522) lies within the green region (the 95% region) around ZERO or, does it lie in the red region (the 5% region)?
        - If within the 95% (green) region: The calculated value is statistically not different from the reference value ZERO
        - If outside the 95% region (ie. within the remaining 5% region): The calculated value is statistically different from the reference value ZERO - and hence stated to be SIGNIFICANT, RELEVANT and VALID



- So, finally, **what is p-value and how should it be interpreted**?
    - We would like to check what is the position of **a1** (= 5.4522) with respect to the reference value (ZERO) of the established Sampling Distribution of **a**, and this we do by calculating the sum of the area(s) under the curve to the far sides of a1, and then comparing it to the reference value of 0.05 (ie. 5%).
    - **p-value** is defined as the **sum** of the areas under the curve beyond the far side of the calculated value, and the corresponding area on the opposite side - since a1 can take positive and negative values. For example, in the figure alongside, the red shaded area to the right lies beyond the calculated value of a1, and this area is 2.36E-38. Similar sized area is also considered to the left, and these two add to a total of 4.72E-38, which is the p-value value reported in the regression output. This belongs to the class of Hypothesis Tests known as the **two-tailed** tests because a1 can take both positive and negative values.
        - In some Hypothesis Tests, the region of interest is on only one side, and such tests are known as **one-tailed** tests.
    - As this p-value is much less than the threshold value of 0.05 (ie. 5%), we can conclude that 5.4522 is outside the 95% confidence interval for the reference value of ZERO, hence it STATISTICALLY SIGNIFICANT (ie. statistically different from ZERO) and we can rely on it.
    - If the p-value was greater than 0.05, it would necessarily imply that the calculated point 5.4522 is actually lying inside the 95% CI (the green zone), hence NOT STATISTICALLY SIGNIFICANT (ie. not different from ZERO).
    - As we will see later, in MLR, the p-values help us to select the most appropriate independent variables (**x**'s) to be included in the model.



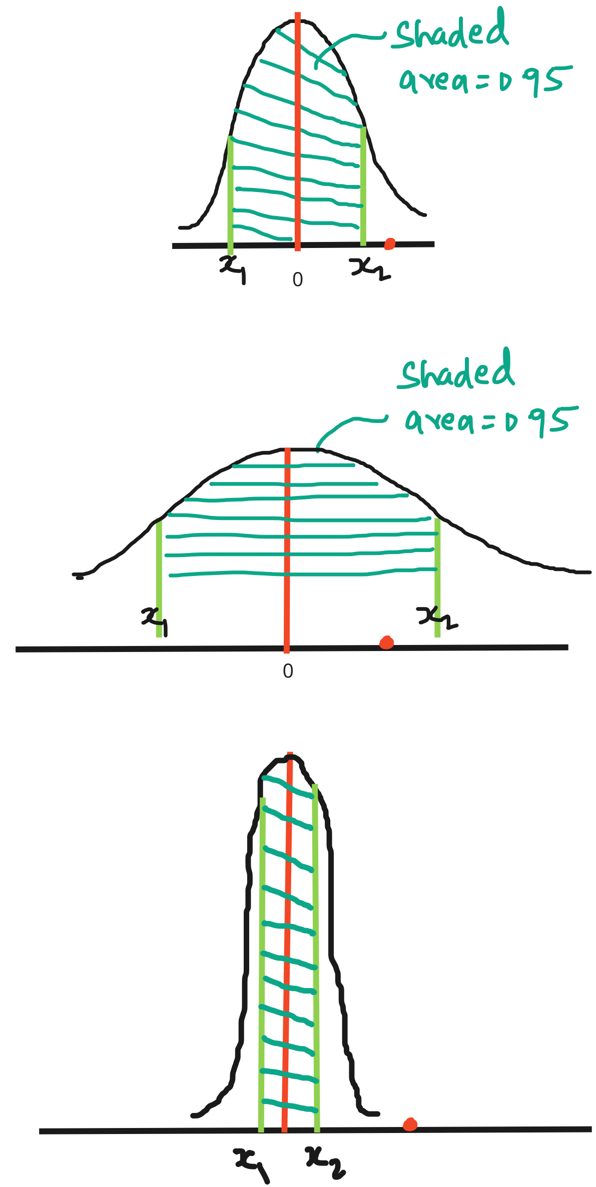
LR output also calculates x1 and x2 with the sampling distribution centered around a1 = 5.4522.
In the example:
x1 = 4.9398
x2 = 5.9647

It is worth noting that ZERO is NOT a part of this interval

This also tells us that 95 out of 100 samples(si) will result in an ai value lying within this interval

The implications of sample size on the confidence interval:

- It should be noted that the formulae for standard errors have the sample size **n** in the denominator.
- Consequently, **if the sample size is reduced** the standard error value becomes larger and the sampling distribution curve gets stouter and wider as shown alongside, and the x1 and x2 values get pushed further away from ZERO
- The net effect is that values such as the red dot, which was SIGNIFICANT earlier now falls within the **not statistically different from ZERO zone**! The zone of uncertainty has widened, and it is more difficult to get a valid model.
- Conversely, if the sample size is increased, the standard error value becomes smaller and the sampling distribution curve gets slimmer and the x1, x2 values get pulled towards ZERO - the range of values considered *statistically equal to ZERO* becomes small.
  - This increases the chance that the red dot (calculated a1) will lie far outside the 95% CI for ZERO and hence it will considered as statistically significant. In effect, **uncertainty in the model reduces as the sample size increases**.



What about **F** and *SIgnificance F?*

- These are relevant in case of Multiple Linear Regression (MLR) where **y = f(x1, x2, x3, x4, ...)**
- The **F-statistic** is defined as follows:

$$F = \frac{\frac{SSR}{n-k-1}}{\frac{SSE}{n-1}}$$

n is the number of observations,

k is the number of predictors (independent variables) in the model.

ANOVA					
	df	*SS*	*MS*	*F*	*ignificance F*
Regressior	4	85.75865	21.43966	222.0403	1.26E-67
Residual	176	16.99412	0.096558		
Total	180	102.7528			

	Coefficients	*andard Err*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	0.31406	0.11176	2.810135	0.005513	0.093498	0.534622
x1	12.33273	1.5577	7.917267	2.62E-13	9.258555	15.40691
x2	-38.302	6.359842	-6.02247	9.77E-09	-50.8533	-25.7506
x3	30.31208	9.568272	3.167978	0.00181	11.42877	49.19539
x4	-4.00187	4.746218	-0.84317	0.400277	-13.3687	5.36495

- As we can see, it is the ratio of **average variance explained by regression** and **average variance attributable to random errors (MSR / MSE)**
- The better the regression model, the larger will be the value of the **F-statistic** - that is, significantly more variance in the data will be explained by the regression model.
- **Significance F** is nothing but the p-value associated with the **F-statistic** Therefore, if **Significance F** is less than 0.05 (5%) we can rely on the calculated **F-statistic** and consider it **statistically significant** and use it to confidently make judgements about the overall quality of the Linear Regression model.
- ***We assess the LR model quality by taking into account the calculated coefficients (and their p-values) and the F-statistic (and it's p-value)***
- This is not the end !! There are a few more metrics, that we will encounter soon ...

```
-------------------------------------------------------------
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.891
Model:                            OLS   Adj. R-squared:                  0.890
Method:                 Least Squares   F-statistic:                     607.6
Date:                Tue, 23 Jan 2024   Prob (F-statistic):           2.04e-37
Time:                        22:20:19   Log-Likelihood:                -102.30
No. Observations:                  76   AIC:                             208.6
Df Residuals:                      74   BIC:                             213.3
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          3.2028      0.221     14.499      0.000       2.763       3.643
x              9.1104      0.370     24.650      0.000       8.374       9.847
==============================================================================
Omnibus:                        5.816   Durbin-Watson:                   1.896
Prob(Omnibus):                  0.055   Jarque-Bera (JB):                2.579
Skew:                          -0.112   Prob(JB):                        0.275
Kurtosis:                       2.126   Cond. No.                         4.42
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
-------------------------------------------------------------
```

(ORDINARY LEAST SQ.)

OUTPUT CREATED BY OLS function of STATSMODELS library of Python.

We have already encountered some of the generated numbers like R2, F-statistic, etc. But the adjacent block of out contains many others … what are they, why are they important, and how to interpret them?

AIC (Akaike Information Criteria) and BIC (Bayes Information ...)

- Both these are estimators of **prediction error**. They help in model selection.
- These numbers are used to compare across models. **A lower number indicates a better model**
- A difference in AIC or BIC value of 2, between models being compared, is considered significant. The model with a lower AIC or BIC value is designated as the better model, and becomes a candidate for selection.

- **Omnibus statistic:** This is a numeric value calculated from the skewness and kurtosis of the residuals (the difference between the predicted and actual values). A low value suggests the residuals are closer to a normal distribution, while a high value indicates deviation from normality.
- **Omnibus p-value:** This value represents the probability of observing the calculated Omnibus statistic, assuming the null hypothesis of normally distributed residuals is true. A low p-value (typically < 0.05) suggests there is significant evidence to reject the null hypothesis, implying the residuals are not normally distributed.

Skewness measures the asymmetry of the probability distribution of a dataset. For a normal distribution, the skewness should be **close to 0**.

- **Skewness = 0**: The data is symmetric, which is a characteristic of a normal distribution.
- **Skewness > 0**: The data is positively skewed (right-tailed), meaning the tail on the right side is longer or fatter.
- **Skewness < 0**: The data is negatively skewed (left-tailed), meaning the tail on the left side is longer or fatter.

Interpretation:

- If the skewness is significantly different from 0, the data is unlikely to follow a normal distribution.

Kurtosis measures the "tailedness" of the probability distribution, indicating whether the data has heavy or light tails compared to a normal distribution. For a normal distribution, the kurtosis is **close to 3** (or 0 if using excess kurtosis).

- **Kurtosis = 3 (or Excess Kurtosis = 0)**: The data has tails similar to a normal distribution.
- **Kurtosis > 3 (or Excess Kurtosis > 0)**: The data has heavier tails (more outliers) than a normal distribution (leptokurtic).
- **Kurtosis < 3 (or Excess Kurtosis < 0)**: The data has lighter tails (fewer outliers) than a normal distribution (platykurtic).

Interpretation:

- If the kurtosis is significantly different from 3 (or excess kurtosis significantly different from 0), the data is unlikely to follow a normal distribution.

Jarque-Bera Test

- In the context of Ordinary Least Squares (OLS) regression, the Jarque-Bera test is used to **check the normality of the residuals.**
    - **Residuals**, the difference between the predicted and actual values in your model, play a crucial role in OLS analysis.
    - Their normality is one of the key assumptions for the validity of statistical inferences drawn from the model.
- A **low statistic** : Low value of the Jarque-Bera statistic (< 2) along with high p-value (ie. > 0.05) indicate that the residuals follow Normal Distribution.
- A **high statistic** : High value of the Jarque-Bera statistic (> 6) often accompanied with low p-values (ie. < 0.05) indicate that the residuals DO NOT follow Normal Distribution.

Durbin-Watson Test

- In the context of Ordinary Least Squares (OLS) regression, the Durbin-Watson (DW) test is a diagnostic tool used to check for **autocorrelation** in the residuals (errors) of the model. Autocorrelation occurs when there's a dependence between subsequent errors, meaning the error term at one point in time influences the error term at another point.
- Its value always falls between 0 and 4, with specific interpretations:
    - **2.0:** Indicates no autocorrelation (ideal scenario). Good regression!
    - **0 to less than 2.0:** Suggests positive autocorrelation (errors tend to cluster together, either positive or negative).
    - **More than 2.0 to 4:** Suggests negative autocorrelation (errors tend to alternate between positive and negative).
- This test is used as a first check, and not a definitive test.

Condition Number

The Condition Number in Ordinary Least Squares (OLS) refers to a **measure of how sensitive the estimated coefficients are to small changes in the data**. It's not directly related to any specific variable or error term, but rather evaluates the overall stability and robustness of the model's solution. Its calculation is based on eigenvalues

- A **low Condition Number** indicates that the coefficients react minimally to small changes in the data (stable, robust model).
- A **high Condition Number** signifies that even slight data variations can significantly alter the coefficients (sensitive, potentially unstable model).

Why is it important?

- A high Condition Number suggests the model might be fitting noise or capturing spurious relationships due to its sensitivity to slight data changes. This makes the estimated coefficients less reliable and conclusions less trustworthy.
- In extreme cases, a very high Condition Number can lead to numerical issues during calculations, rendering the model estimation altogether unstable.

Interpretation:

There's no single threshold for a "good" or "bad" Condition Number. However, in general:

- **Values below 10** are considered acceptable, indicating a relatively stable model.
- **Values above 30** raise concerns about sensitivity and potential instability.
- **Values above 100** are a strong indicator of an unreliable model requiring further investigation or improvement.

Output	Interpretation	Specific Limits/Values (if applicable)
R^2	Proportion of variance in the dependent variable explained by the model.	Range: [0, 1], higher values are desirable.
Adjusted R^2	R^2 adjusted for the number of predictors; a measure of model fit.	Like R^2, but adjusted for model complexity.
F-statistic	Tests the overall significance of the regression model.	Critical values based on significance level (e.g., 0.05).
AIC (Akaike's IC)	A measure of model goodness-of-fit, balancing complexity and fit.	Lower values are better; used for model comparison.
BIC (Bayesian IC)	Similar to AIC but penalizes model complexity more heavily.	Lower values are better; stricter penalty for complexity.
Log Likelihood	A measure of how well the model explains the observed data.	Higher values indicate better model fit.
Omnibus	Refers to a specific statistic and its associated p-value that test the normality of the residuals. It is a combination of multiple tests like Jarques-Bera test, Shapiro-Wilkes test and Kolmongoriv-Smirnov test.	For the residuals to have Normal Distribution, the Omnibus statistic should have low value and p-value should be > 0.05
Durbin-Watson test	Tests for autocorrelation in the residuals; values around 2 suggest no autocorrelation.	Range: [0, 4], close to 2 indicates no significant autocorrelation.
Jarque-Bera test	Tests for normality of residuals based on skewness and kurtosis. Test statistic value closer to zero implies residuals are normally distributed NULL Hypothesis: The residuals are Normally Distributed	Critical values based on significance level (e.g., 0.05). If p-value < 0.05, then NULL Hypothesis is rejected implying the residuals are NOT normally distributed. If residuals are normally distributed, p-value > 0.05.
Condition Number	Measures sensitivity to changes in input variables; high values indicate multicollinearity.	No strict limits; values above 30 may indicate multicollinearity.
Skew	A measure of the asymmetry of the residuals distribution.	Range: (-∞, ∞); 0 for a perfectly symmetric distribution.
Kurtosis	A measure of the "tailedness" of the residuals distribution.	Range: (-∞, ∞); 3 for a normal distribution (excess kurtosis).

MULTIPLE LINEAR REGRESSION.

→ SLR deals with only one independent variable, and takes the form:

$$y = a \cdot x + b \quad \underline{\text{or}} \quad y = \beta_0 + \beta_1 x.$$

→ MLR deals with more than one independent variable, and takes the form:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots\cdots + \beta_k x_k$$

here $[x_1, x_2, x_3, \ldots x_k]$ are the independent variables, also known as "<u>features</u>"

A data Set for MLR will look as shown below

y	x1	x2	x3	x4
0.038117	0	0	0	0
0.896468	0.005556	3.09E-05	1.71E-07	9.53E-10
0.159546	0.011111	0.000123	1.37E-06	1.52E-08
0.863764	0.016667	0.000278	4.63E-06	7.72E-08
1.106349	0.022222	0.000494	1.1E-05	2.44E-07
1.010169	0.027778	0.000772	2.14E-05	5.95E-07
0.278498	0.033333	0.001111	3.7E-05	1.23E-06
1.114231	0.038889	0.001512	5.88E-05	2.29E-06
1.029804	0.044444	0.001975	8.78E-05	3.9E-06
0.37387	0.05	0.0025	0.000125	6.25E-06
0.971634	0.055556	0.003086	0.000171	9.53E-06
0.975377	0.061111	0.003735	0.000228	1.39E-05
1.079774	0.066667	0.004444	0.000296	1.98E-05
1.24279	0.072222	0.005216	0.000377	2.72E-05
0.644699	0.077778	0.006049	0.000471	3.66E-05
0.656177	0.083333	0.006944	0.000579	4.82E-05

features.

$\boxed{\text{TRAIN DATA.}}$

In MLR, the goal is to express 'y' as a linear combination of $x_1, x_2, \dots$

$$\therefore \quad y_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{21} + \dots + \beta_K x_{K1} + e_1$$

$$y_2 = \beta_0 + \beta_1 x_{12} + \beta_2 x_{22} + \dots + \beta_K x_{K2} + e_2$$

$$\vdots$$

$$y_n = \beta_0 + \beta_1 x_{1n} + \dots + \beta_K x_{Kn} + e_n$$

'K' features

What is MLR?

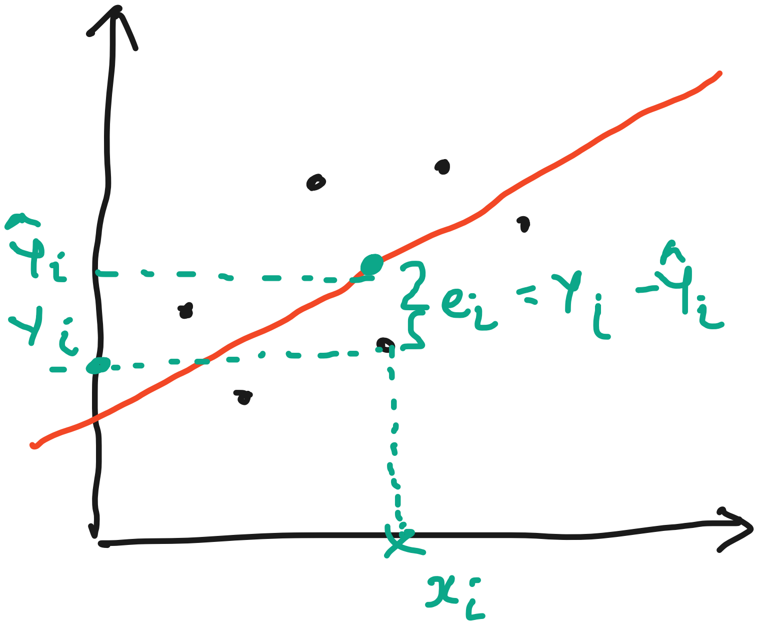
Using the values of all x_{ij} and the corresponding values of y_j, find out the most appropriate $\beta_0, \beta_1, \dots \beta_K$.

How do we go about it?

The model that we create, i.e. the values of β_0, β_1, etc. that we identify should be such that

$$\sum_{i=1}^{n} e_i^2 \quad \text{should be minimized}$$

(minimize the sum of square of errors, SSE)

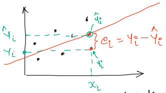


Note:

$$y_i = \beta_0 + \sum_{j=1}^{m} \beta_j x_{ij} + e_i$$

$$\hat{y}_i = \beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}$$

MLR GRADIENT DESCENT (OPTIONAL READING!)



$$y_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{21} + \beta_3 x_{31} + \cdots + e_1$$

Matrices used in the derivations...

$$\begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}_{m\times 1} = \begin{bmatrix} 1 & x_{11} & x_{21} & x_{31} & \cdots & x_{k1} \\ - & - & - & - & \cdots & - \\ - & - & - & - & \cdots & - \\ 1 & x_{1m} & x_{2m} & x_{3m} & \cdots & x_{km} \end{bmatrix}_{m\times l} \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_k \end{bmatrix}_{l\times 1} + \begin{bmatrix} e_1 \\ \vdots \\ e_m \end{bmatrix}_{m\times 1}$$

m = Number of observations (records)

k = Number of features (independent variables)

$l = k + 1$

→ $Y = X.\beta + E$ and $\hat{Y} = X.\beta$ ... the regression 'line'

$E = Y - X.\beta$

$E^T.E = (Y - X\beta)^T (Y - X\beta)$ .... Sum of squares of errors

$J = \frac{1}{2m}(E^T.E)$ ... Cost function.

Goal: minimize J (ie: make $\frac{dJ}{d\beta} = 0$)

$$J = \frac{1}{2m}(Y - X\beta)^T(Y - X\beta)$$

$$= \frac{1}{2m}\left[(Y^T - \beta^T X^T)(Y - X\beta)\right]$$

$$= \frac{1}{2m}\left[Y^T.Y - Y^T(X\beta) - (\beta^T X^T)Y + (\beta^T X^T).(X\beta)\right]$$

$$\frac{dJ}{d\beta} = \frac{1}{2m}\left[\underbrace{\frac{d}{d\beta}(Y^T.Y)}_{A} - \underbrace{\frac{d}{d\beta}\{Y^T.(X\beta)\}}_{B} - \underbrace{\frac{d}{d\beta}\{(\beta^T X^T).Y\}}_{C} + \underbrace{\frac{d}{d\beta}\{(\beta^T X^T).(X\beta)\}}_{D}\right]$$

$A = \frac{d}{d\beta}(Y^T.Y) = 0$ ... Y is a vector of constants.

$B = \frac{d}{d\beta}\{Y^T.(X\beta)\} = Y^T.\frac{d}{d\beta}(X\beta) + \frac{dY^T}{d\beta}.(X\beta)$

$= Y^T.X + 0 = Y^T X$ ... Row vector $(1 \times l)$

$C = \frac{d}{d\beta}\{(\beta^T X^T).Y\} = \beta^T X^T \frac{dY}{d\beta} + \frac{d(\beta^T X^T)}{d\beta}.Y$

$= 0 + X^T Y$ ... Column Vector $(l \times 1)$

$D = \frac{d}{d\beta}\{(\beta^T X^T).(X\beta)\} = (\beta^T X^T)\frac{d}{d\beta}(X\beta) + \frac{d}{d\beta}(\beta^T X^T).(X\beta)$

$= \underbrace{(\beta^T X^T X)}_{1\times l} + \underbrace{(X^T X \beta)}_{l\times 1}$

$$\frac{dJ}{d\beta} = \frac{1}{2m}\left[(\beta^T X^T X) + (X^T X \beta) - Y^T X - X^T Y\right]$$

These are all 'vectors' and the results in the pairs shown below are equal in values. Hence re-arranging and simplifying ...

$$= \frac{1}{2m}\left[\underbrace{2.X^T X\beta}_{(l\times 1)} - \underbrace{2X^T Y}_{(l\times 1)}\right]$$

$$= X^T(X\beta - Y) \quad \ldots$$

$$\boxed{\frac{dJ}{d\beta} = X^T(\hat{Y} - Y)}$$ ... Since $\hat{Y} = X\beta$

The Gradient Descent process

① assume some value for β. Eg $\begin{bmatrix} \vdots \end{bmatrix}$ ←

② Using $\hat{Y} = X\beta$, evaluate $\hat{Y}$ ←

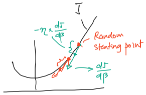
③ Calculate $\frac{dJ}{d\beta}$ as per the above expression

by assuming some value for η (eg: 0.05)

④ Calculate new values for β using the following expression

$$\beta_{new} \leftarrow \beta_{old} - \eta(\nabla J)$$

⑤ Repeat steps 2 to 4 till $[abs(\beta_{new} - \beta_{old})]$ reaches a threshold level (eg: 0.0001)



MULTIPLE LINEAR REGRESSION. (MLR)

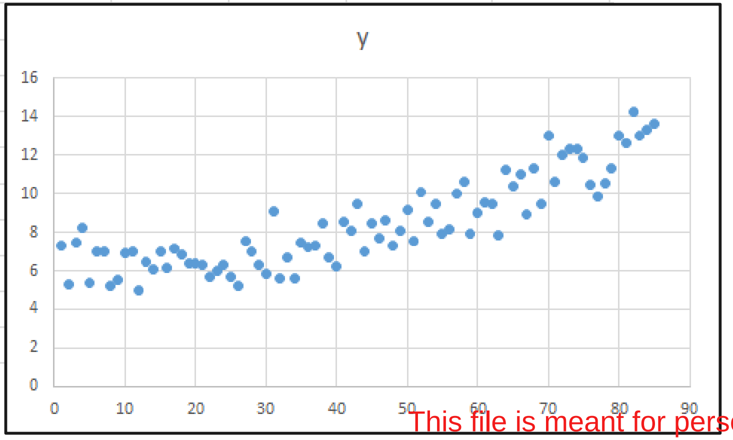
- In MLR, more than one independent variable x_i potentially determine the dependent variable 'y'

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots\cdots + \beta_K x_K$$

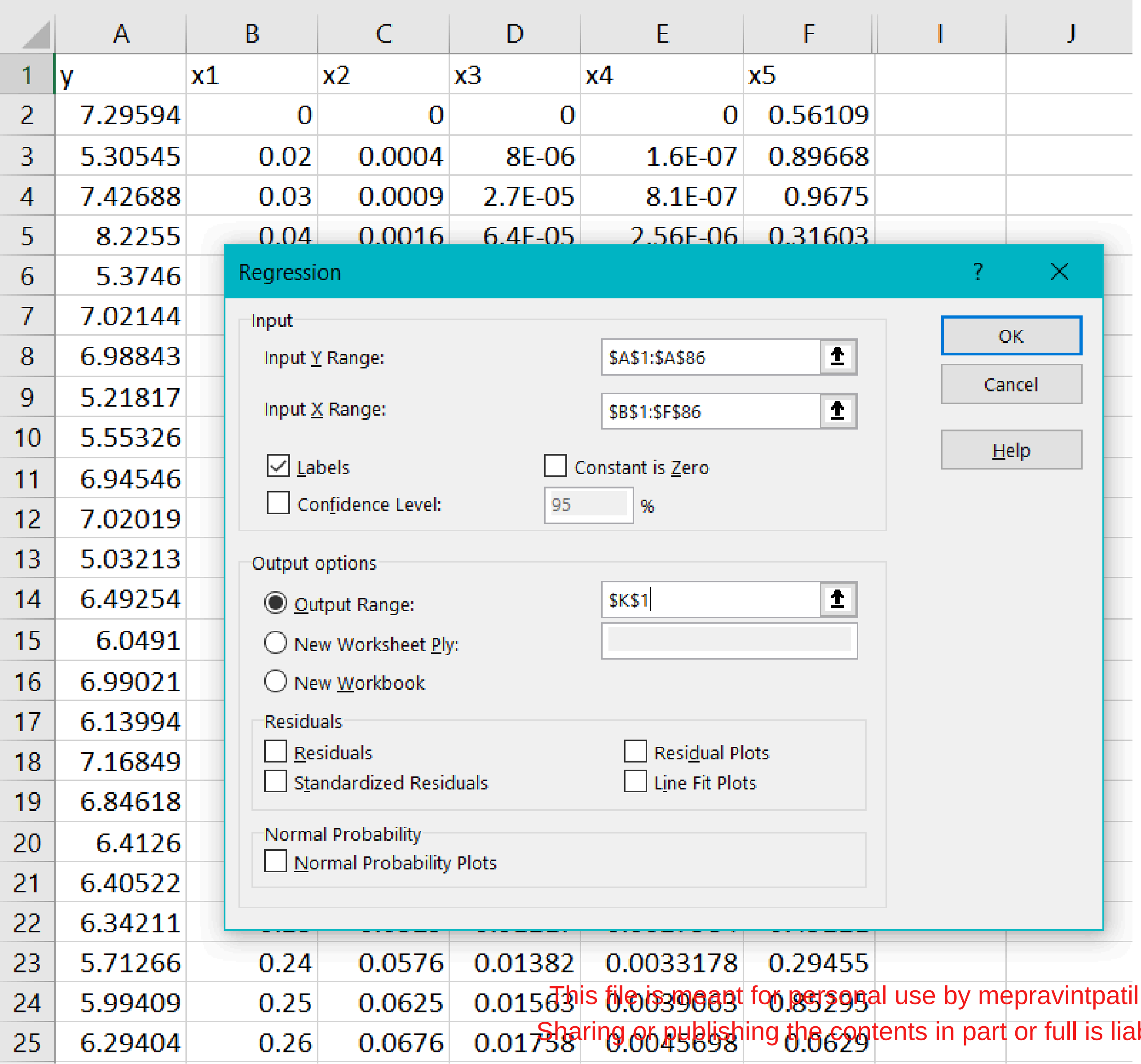
- MLR involves calculating the coefficients β_i using the given dataset (TRAIN DATA).
- As in the case of SLR, these values are obtained by minimizing the SSE (explained in a separate document)
- MLR involves identifying the most relevant predictors as explained subsequently.

MLR using the dataset **data-set-for-MLR.xlsx**

	A	B	C	D	E	F
1	y	x1	x2	x3	x4	x5
2	7.29594	0	0	0	0	0.56109
3	5.30545	0.02	0.0004	8E-06	1.6E-07	0.89668
4	7.42688	0.03	0.0009	2.7E-05	8.1E-07	0.9675
5	8.2255	0.04	0.0016	6.4E-05	2.56E-06	0.31603
6	5.3746	0.05	0.0025	0.00013	6.25E-06	0.74414
7	7.02144	0.06	0.0036	0.00022	1.296E-05	0.19197
8	6.98843	0.07	0.0049	0.00034	2.401E-05	0.86862
9	5.21817	0.08	0.0064	0.00051	4.096E-05	0.74443
10	5.55326	0.09	0.0081	0.00073	6.561E-05	0.801
11	6.94546	0.1	0.01	0.001	0.0001	0.05507
12	7.02019	0.11	0.0121	0.00133	0.0001464	0.61864
13	5.03213	0.12	0.0144	0.00173	0.0002074	0.80112
14	6.49254	0.13	0.0169	0.0022	0.0002856	0.78146
15	6.0491					
16	6.99021					
17	6.13994					
18	7.16849					
19	6.84618					
20	6.4126					
21	6.40522					
22	6.34211					
23	5.71266					
24	5.99409					
25	6.29404					



- The dataset in the file consists of the **train** dataset of 85 observations and the **test** dataset of 15 observations
- We will create an MLR model using the train dataset and subsequently validate the model using the test dataset
- We start by creating an MLR model using all the **x** variables (also known as ***features***)
- A scatter plot of **y** reveals that the observations are non-linear …
    - So, will **Linear Regression** be able to create a good and acceptable model??



Invoking the Linear Regression functionality of Excel and selecting the variables ...

y	x1	x2	x3	x4	x5
7.29594	0	0	0	0	0.56109
5.30545	0.02	0.0004	8E-06	1.6E-07	0.89668
7.42688	0.03	0.0009	2.7E-05	8.1E-07	0.9675
8.2255	0.04	0.0016	6.4E-05	2.56E-06	0.31603
5.3746	0.05	0.0025	0.00013	6.25E-06	0.74414
7.02144	0.06	0.0036	0.00022	1.296E-05	0.19197
6.98843	0.07	0.0049	0.00034	2.401E-05	0.86862
5.21817	0.08	0.0064	0.00051	4.096E-05	0.74443
5.55326	0.09	0.0081	0.00073	6.561E-05	0.801
6.94546	0.1	0.01	0.001	0.0001	0.05507
7.02019	0.11	0.0121	0.00133	0.0001464	0.61864
5.03213	0.12	0.0144	0.00173	0.0002074	0.80112
6.49254	0.13	0.0169	0.0022	0.0002856	0.78146
6.0491	0.15	0.0225	0.00338	0.0005063	0.68791
6.99021	0.16	0.0256	0.0041	0.0006554	0.59236
6.13994	0.17	0.0289	0.00491	0.0008352	0.20935
7.16849	0.18	0.0324	0.00583	0.0010498	0.96488
6.84618	0.19	0.0361	0.00686	0.0013032	0.48896
6.4126	0.21	0.0441	0.00926	0.0019448	0.50876
6.40522	0.22	0.0484	0.01065	0.0023426	0.71971
6.34211	0.23	0.0529	0.01217	0.0027984	0.49221
5.71266	0.24	0.0576	0.01382	0.0033178	0.29455
5.99409	0.25	0.0625	0.01563	0.0039063	0.85295
6.29404	0.26	0.0676	0.01758	0.0045698	0.0629

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.912936908
R Square	0.833453799
Adjusted R Square	0.8229129
Standard Error	0.991752189
Observations	85

— ok

ANOVA

	df	*SS*	*MS*	*F*	*gnificance F*
Regression	5	388.848	77.7697	79.0686	2.7E-29
Residual	79	77.7022	0.98357		
Total	84	466.551			

→ ok

	Coefficients	*andard Err*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	6.908046001	0.59691	11.5731	1.1E-18	5.71993	8.09616
x1	-9.846555611	7.41707	-1.32755	0.18815	-24.6099	4.91676
x2	39.10941696	30.7379	1.27235	0.20698	22.0728	100.292
x3	-42.57714279	46.9855	-0.90618	0.3676	136.099	50.9451
x4	20.26808239	23.6657	0.85643	0.39435	-26.8374	67.3736
x5	0.10562362	0.41144	0.25672	0.79806	-0.71333	0.92457

None of these values are acceptable since their corresponding p-values are all MUCH GREATER than 0.05

So, we DISCARD x_5 — which has the highest p-value, and re-create the model.

y	x1	x2	x3	x4	x5
7.29594	0	0	0	0	0.56109
5.30545	0.02	0.0004	8E-06	1.6E-07	0.89668
7.42688	0.03	0.0009	2.7E-05	8.1E-07	0.9675
8.2255	0.04	0.0016	6.4E-05	2.56E-06	0.31603
5.3746	0.05	0.0025	0.00013	6.25E-06	0.74414
7.02144	0.06	0.0036	0.00022	1.296E-05	0.19197
6.98843	0.07	0.0049	0.00034	2.401E-05	0.86862
5.21817	0.08	0.0064	0.00051	4.096E-05	0.74443
5.55326	0.09	0.0081	0.00073	6.561E-05	0.801
6.94546	0.1	0.01	0.001	0.0001	0.05507
7.02019	0.11	0.0121	0.00133	0.0001464	0.61864
5.03213	0.12	0.0144	0.00173	0.0002074	0.80112
6.49254	0.13	0.0169	0.0022	0.0002856	0.78146
6.0491	0.15	0.0225	0.00338	0.0005063	0.68791
6.99021	0.16	0.0256	0.0041	0.0006554	0.59236
6.13994	0.17	0.0289	0.00491	0.0008352	0.20935
7.16849	0.18	0.0324	0.00583	0.0010498	0.96488
6.84618	0.19	0.0361	0.00686	0.0013032	0.48896
6.4126	0.21	0.0441	0.00926	0.0019448	0.50876
6.40522	0.22	0.0484	0.01065	0.0023426	0.71971
6.34211	0.23	0.0529	0.01217	0.0027984	0.49221
5.71266	0.24	0.0576	0.01382	0.0033178	0.29455
5.99409	0.25	0.0625	0.01563	0.0039063	0.85295
6.29404	0.26	0.0676	0.01758	0.0045698	0.0629

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.912860812
R Square	0.833314862
Adjusted R Square	0.824980605
Standard Error	0.985945239
Observations	85

ANOVA

	df	*SS*	*MS*	*F*	*gnificance F*
Regression	4	388.783	97.1959	99.9867	2.6E-30
Residual	80	77.767	0.97209		
Total	84	466.551			

	Coefficients	*andard Err*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	6.982707415	0.51821	13.4746	2.8E-22	5.95143	8.01398
x1	-9.95691922	7.36125	-1.35261	0.17999	-24.6063	4.69243
x2	39.18955261	30.5563	1.28254	0.20336	-21.6194	99.9986
x3	-42.50561677	46.7095	-0.91	0.36556	-135.461	50.4493
x4	20.19742663	23.5256	0.85853	0.39316	-26.62	67.0148

Discard x_4 and proceed ...

	y	x1	x2	x3	x4	x5
2	7.29594	0	0	0	0	0.56109
3	5.30545	0.02	0.0004	8E-06	1.6E-07	0.89668
4	7.42688	0.03	0.0009	2.7E-05	8.1E-07	0.9675
5	8.2255	0.04	0.0016	6.4E-05	2.56E-06	0.31603
6	5.3746	0.05	0.0025	0.00013	6.25E-06	0.74414
7	7.02144	0.06	0.0036	0.00022	1.296E-05	0.19197
8	6.98843	0.07	0.0049	0.00034	2.401E-05	0.86862
9	5.21817	0.08	0.0064	0.00051	4.096E-05	0.74443
10	5.55326	0.09	0.0081	0.00073	6.561E-05	0.801
11	6.94546	0.1	0.01	0.001	0.0001	0.05507
12	7.02019	0.11	0.0121	0.00133	0.0001464	0.61864
13	5.03213	0.12	0.0144	0.00173	0.0002074	0.80112
14	6.49254	0.13	0.0169	0.0022	0.0002856	0.78146
15	6.0491	0.15	0.0225	0.00338	0.0005063	0.68791
16	6.99021	0.16	0.0256	0.0041	0.0006554	0.59236
17	6.13994	0.17	0.0289	0.00491	0.0008352	0.20935
18	7.16849	0.18	0.0324	0.00583	0.0010498	0.96488
19	6.84618	0.19	0.0361	0.00686	0.0013032	0.48896
20	6.4126	0.21	0.0441	0.00926	0.0019448	0.50876
21	6.40522	0.22	0.0484	0.01065	0.0023426	0.71971
22	6.34211	0.23	0.0529	0.01217	0.0027984	0.49221
23	5.71266	0.24	0.0576	0.01382	0.0033178	0.29455
24	5.99409	0.25	0.0625	0.01563	0.0039063	0.85295
25	6.29404	0.26	0.0676	0.01758	0.0045698	0.0629

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.912019254
R Square	0.83177912
Adjusted R Square	0.825548717
Standard Error	0.984343752
Observations	85

ANOVA

	df	*SS*	*MS*	*F*	*gnificance F*
Regression	3	388.067	129.356	133.503	2.9E-31
Residual	81	78.4835	0.96893		
Total	84	466.551			

	Coefficients	*andard Err*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	6.717731373	0.4156	16.164	4.5E-27	5.89082	7.54464
x1	-4.499675935	3.70651	-1.21399	0.22828	-11.8745	2.87513
x2	14.05360633	8.73189	1.60946	0.11141	-3.32013	31.4273
x3	-2.717152907	5.81602	-0.46718	0.64162	-14.2892	8.8549

Discard x_3 and proceed ...

	A	B	C	D	E	F
1	y	x1	x2	x3	x4	x5
2	7.29594	0	0	0	0	0.56109
3	5.30545	0.02	0.0004	8E-06	1.6E-07	0.89668
4	7.42688	0.03	0.0009	2.7E-05	8.1E-07	0.9675
5	8.2255	0.04	0.0016	6.4E-05	2.56E-06	0.31603
6	5.3746	0.05	0.0025	0.00013	6.25E-06	0.74414
7	7.02144	0.06	0.0036	0.00022	1.296E-05	0.19197
8	6.98843	0.07	0.0049	0.00034	2.401E-05	0.86862
9	5.21817	0.08	0.0064	0.00051	4.096E-05	0.74443
10	5.55326	0.09	0.0081	0.00073	6.561E-05	0.801
11	6.94546	0.1	0.01	0.001	0.0001	0.05507
12	7.02019	0.11	0.0121	0.00133	0.0001464	0.61864
13	5.03213	0.12	0.0144	0.00173	0.0002074	0.80112
14	6.49254	0.13	0.0169	0.0022	0.0002856	0.78146
15	6.0491	0.15	0.0225	0.00338	0.0005063	0.68791
16	6.99021	0.16	0.0256	0.0041	0.0006554	0.59236
17	6.13994	0.17	0.0289	0.00491	0.0008352	0.20935
18	7.16849	0.18	0.0324	0.00583	0.0010498	0.96488
19	6.84618	0.19	0.0361	0.00686	0.0013032	0.48896
20	6.4126	0.21	0.0441	0.00926	0.0019448	0.50876
21	6.40522	0.22	0.0484	0.01065	0.0023426	0.71971
22	6.34211	0.23	0.0529	0.01217	0.0027984	0.49221
23	5.71266	0.24	0.0576	0.01382	0.0033178	0.29455
24	5.99409	0.25	0.0625	0.01563	0.0039063	0.85295
25	6.29404	0.26	0.0676	0.01758	0.0045698	0.0629

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.911770714
R Square	0.831325834
Adjusted R Square	0.82721183
Standard Error	0.979640446
Observations	85

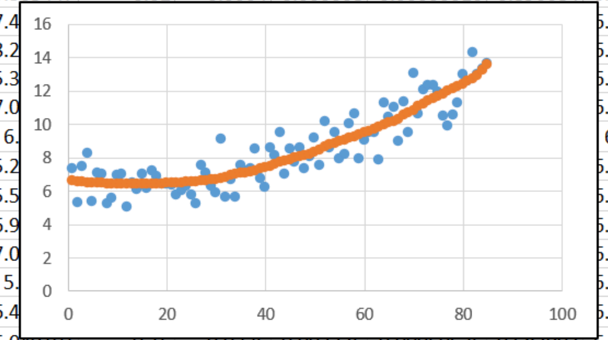
ANOVA

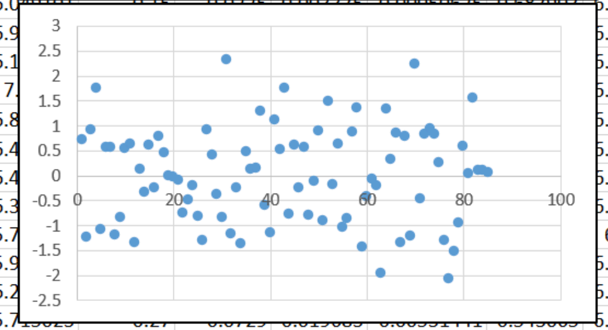
	df	*SS*	*MS*	*F*	*gnificance F*
Regression	2	387.856	193.928	202.072	2E-32
Residual	82	78.695	0.9597		
Total	84	466.551			

	Coefficients	*andard Err*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	6.589411803	0.31041	21.2281	4.6E-35	5.97191	7.20692
x1	-2.914457293	1.48449	-1.96328	0.053	-5.86757	0.03866
x2	10.03277678	1.46738	6.8372	1.3E-09	7.11369	12.9519

The model seems ok now ...
p-value of x_1 is very close to 0.05, so we choose to keep it ...

y	x1	x2	x3	x4	x5	ycap	e
7.295939	0	0	0	0	0.561088	6.589412	0.706527
5.305447	0.02	0.0004	0.000008	0.00000016	0.896684	6.535136	-1.22969
7.4						6.511008	0.915876
8.2						6.488886	1.736618
5.3						6.468771	-1.09418
7.0						6.450662	0.570781
6.						6.43456	0.55387
5.2						6.420465	-1.2023
5.5						6.408376	-0.85512
6.9						6.398294	0.547164
7.0						6.390218	0.629977
5.						6.384149	-1.35202
6.4						6.380086	0.112449
6.0						6.377981	-0.32888
6.9						6.379938	0.610267
6.1						6.383901	-0.24396
7.						6.389871	0.778618
6.8						6.397848	0.448328
6.4						6.419821	-0.00722
6.4						6.433818	-0.02859
6.3						6.449821	-0.10771
5.7						6.46783	-0.75517
5.9						6.487846	-0.49375
6.2						6.509869	-0.21582
5.7						6.533898	-0.81888
5.2							





SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.911770714
R Square	0.831325834
Adjusted R Square	0.82721183
Standard Error	0.979640446
Observations	85

ANOVA

	df	*SS*	*MS*	*F*	*gnificance F*
Regression	2	387.8555	193.9278	202.0722	2.04E-32
Residual	82	78.69502	0.959695		
Total	84	466.5505			

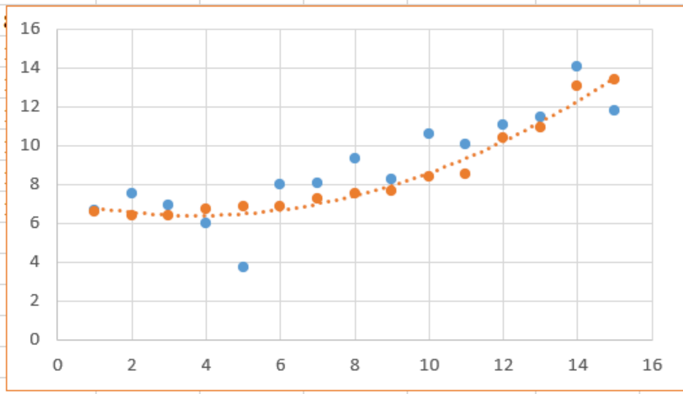
	Coefficients	*andard Err*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	6.589411803	0.31041	21.22811	4.59E-35	5.971908	7.206916
x1	-2.914457293	1.484485	-1.96328	0.053004	-5.86757	0.038656
x2	10.03277678	1.467381	6.837202	1.32E-09	7.113689	12.95186

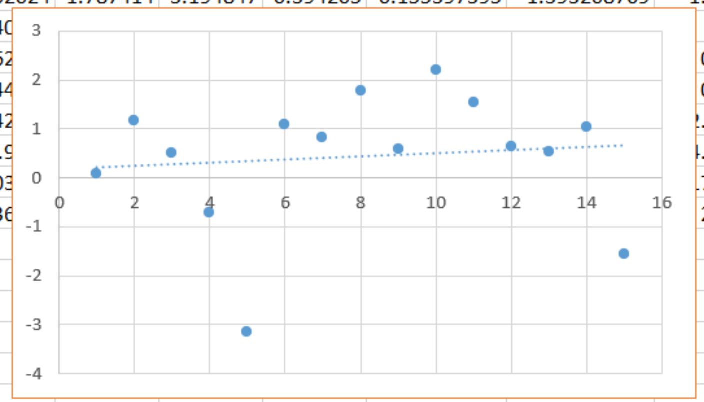
- Plot of y (blue dots) and y_cap (red dots) seems to indicate that the model has captured the non-linear nature of **y** (how? why?)
- Plot of residuals also indicate no visible trend
- Model indicators like R2 and F-statistic also seem Ok
- Overall - the model seems to be good and acceptable based its performance on the train data
- We now need to check its performance on the test data

The illustrated method of iteratively eliminating the **least important features** features, based on their p-values, is known as **BACKWARD FEATURES ELIMINATION**

Model performance on **Test Data**

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	y	x1	x2	x3	x4	x5	**ycap**	**e**	e_sq	y-y_bar	y-y_bar_sq	y_cap-y_bar	y_cap-y_bar_sq				
2	6.65375	0.01	0.0001	0.000001	0.00000001	0.419067	6.561271	0.092479	0.008552	-2.24148	5.024246447	-2.333962333	5.447380171			Coefficients	
3	7.539954	0.14	0.0196	0.002744	0.00038416	0.209814	6.37803	1.161924	1.350067	-1.35528	1.836780485	-2.517202634	6.3363091		Intercept	6.589411803	
4	6.914127	0.2	0.04	0.008	0.0016	0.375459	6.407831	0.506296	0.256335	-1.98111	3.924780608	-2.487401425	6.18716585		x1	-2.914457293	
5	6.006237	0.33	0.1089	0.035937	0.01185921	0.752656	6.72021	-0.71397	0.509758	-2.889	8.346296524	-2.175022553	4.730723107		x2	10.03277678	
6	3.702046	0.36	0.1296	0.046656	0.01679616	0.393727	6.840455	-3.13841	9.849613	-5.19319	26.96919177	-2.054777793	4.222111778				
7	7.970599	0.37	0.1369	0.050653	0.01874161	0.416966	6.88455	1.08605	1.179504	-0.92463	0.854947091	-2.010683095	4.042846509		SSE	28.70223718	
8	8.067204	0.44	0.1936	0.085184	0.03748096	0.815868	7.249396	0.817808	0.66881	-0.82803	0.685631517	-1.645836662	2.708778319		MSE_test	1.913482479	
9	9.289438	0.48	0.2304	0.110592	0.05308416	0.327166	7.502024	1.787414	3.194847	0.394205	0.155397395	-1.393208769	1.941030673		MSE_train	0.9258238	
10													1.5746623				
11													0.28402637		y_bar	8.895232841	
12													0.12281082		SST	99.03386445	
13													.325148205		SSR	81.29945607	
14													.203988792				
15													7.16036458		R2	0.820925817	(SSR/SST)
16													20.0121095		R2	0.71017755	(1 - SSE/SST)





why would this be more correct?

- The y and ycap plots seem to indicate that the model, created using train data also performs reasonably well on the test data
- R2 value on test data seems Ok and close to the R2 value using train data
-

- The MSE values are differing, indicating some level of **overfitting** to the train data. However, the size of the test data is small, so the errors are possibly magnified.
- Usually, the technique of **cross-validation** is used - wherein multiple test data sets are used to evaluate the model. This results in an unbiased error estimate on test data.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.831
Model:                            OLS   Adj. R-squared:                  0.827
Method:                 Least Squares   F-statistic:                     202.1
Date:                Fri, 26 Jan 2024   Prob (F-statistic):           2.04e-32
Time:                        14:19:48   Log-Likelihood:                -117.33
No. Observations:                  85   AIC:                             240.7
Df Residuals:                      82   BIC:                             248.0
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          6.5894      0.310     21.228      0.000       5.972       7.207
x1            -2.9145      1.484     -1.963      0.053      -5.868       0.039
x2            10.0328      1.467      6.837      0.000       7.114      12.952
==============================================================================
Omnibus:                        1.380   Durbin-Watson:                   2.318
Prob(Omnibus):                  0.502   Jarque-Bera (JB):                1.164
Skew:                           0.080   Prob(JB):                        0.559
Kurtosis:                       2.450   Cond. No.                         23.2
==============================================================================
```

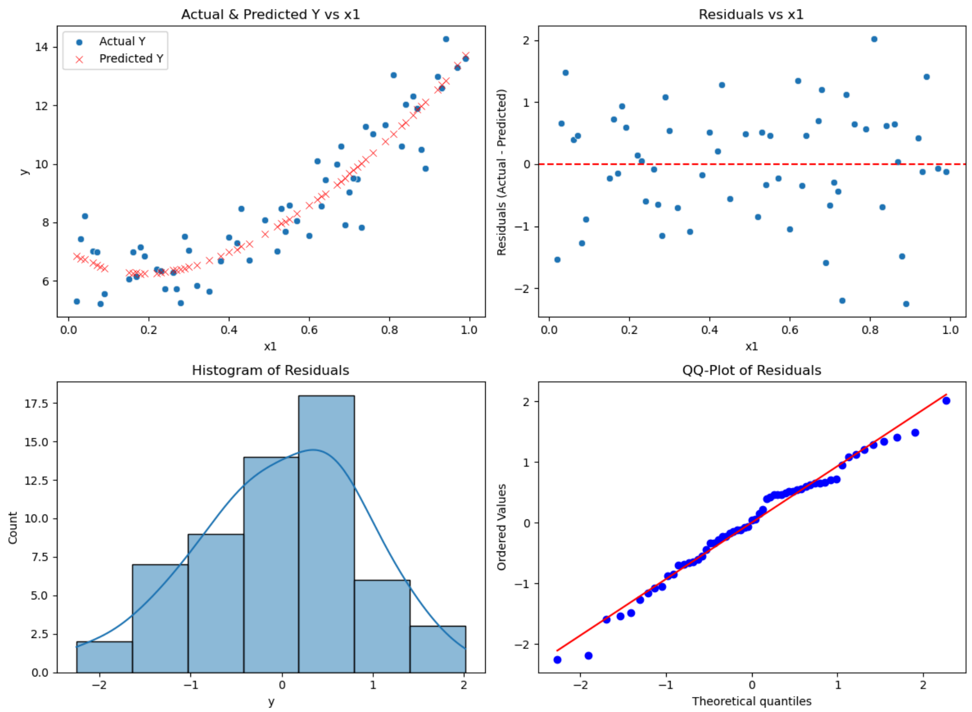
Exercise-2:

1. Re-create and validate the MLR model yourself, using the steps outlined in this document
2. Use the statsmodels based OLS function to repeat **all** these steps, and analyze the additional metrics created (Omnibus, Durbin-Watson, Jarque-Bera, AIC, BIC, Condition Number, etc.) at each stage

CREATE THE PLOTS & DATA SHOWN IN THE NEXT SLIDE

Note: In the data file '**data-set-for-MLR.xlsx**' the train and test data have been given on two different sheets

Sklearn - Train Analysis



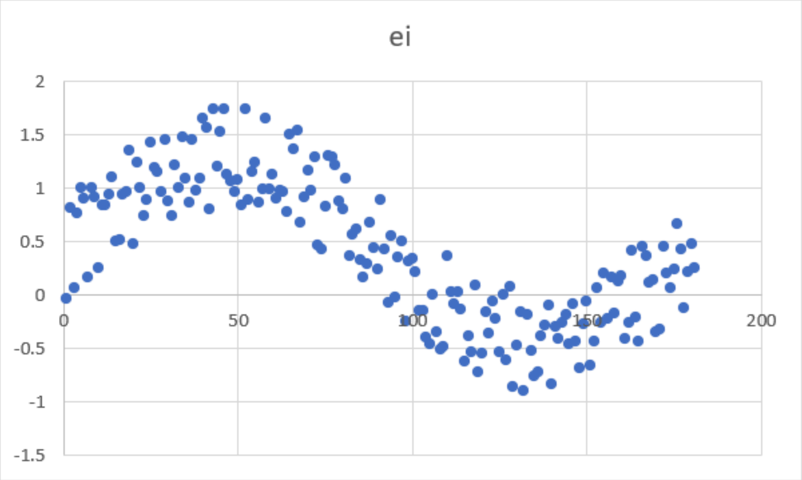

```
DataFrame 'df' successfully loaded.
--------------------------------------------------
Target variable (y): y
Feature variables (x): ['x1', 'x2', 'x3', 'x4', 'x5']
First x variable for plotting: x1
--------------------------------------------------
Data Split: Train size = 59, Test size = 26
--------------------------------------------------


################ SCKIT-LEARN LINEAR REGRESSION ################

--- SKLEARN: Train Set Results ---
--- Sklearn Metrics ---
R2: 0.8602
MAE: 0.7375
RMSE: 0.9094
--------------------
```

What if the relationship between the dependent variable (y) and the independent variables (X) is not linear as in the case below? We have seen that the regression errors are not random and the other important regression parameters like R2 are also very low.

How do we remedy this situation?

	A	B	C	H	I
1	y	x1		ycap	ei
2	0.038116834	0		0.078554	-0.04044
3	0.896467788	0.005555556		0.083296	0.813172
4	0.159545792	0.011111111		0.088032	0.071514
5	0.863764416	0.016666667		0.092756	0.771008
6	1.106349076	0.022222222		0.097463	1.008886
7	1.010169458	0.027777778		0.102147	0.908022
8	0.278498289	0.033333333		0.106803	0.171695
9	1.114230685	0.038888889		0.111424	1.002807
10	1.029803908	0.044444444		0.116005	0.913799
11	0.373869889	0.05		0.12054	0.25333
12	0.971634374	0.055555556		0.125024	0.84661
13	0.975376766	0.061111111		0.129452	0.845925
14	1.079774246	0.066666667		0.133817	0.945957
15	1.242790434	0.072222222		0.138116	1.104675
16	0.644698738	0.077777778		0.142341	0.502358
17	0.656177067	0.083333333		0.146489	0.509688
18	1.09549189	0.088888889		0.150554	0.944938
19	1.115274736	0.094444444		0.154532	0.960743
20	1.512547878	0.1		0.158416	1.354131
21	0.639395626	0.105555556		0.162204	0.477192

	K	L	M	N	O	P	Q	R	S	T
1	SUMMARY OUTPUT									
3	*Regression Statistics*									
4	Multiple R	0.713818524								
5	R Square	0.509536885								
6	Adjusted R Square	0.506796867								
7	Standard Error	0.530607579								
8	Observations	181								
10	ANOVA									
11		*df*	*SS*							
12	Regression	1	52.35633091							
13	Residual	179	50.3964481							
14	Total	180	102.752779							
16		*Coefficients*	*Standard Error*							
17	Intercept	1.417122114	0.078553776	18.04015	3.95E-42	1.262112	1.572133	1.262112	1.572133	0.547807
18	x1	-1.852833934	0.135870553	-13.6368	1.69E-29	-2.12095	-1.58472	-2.12095	-1.58472	12.74861



We can see that the error plot is not random, and it follows a pattern. This indicates that forcing a **line** to model this data results in incorrect results. We need to **introduce non-linear independent variables** in the system so that the Multiple Linear Regression method can 'use' this non-linearity to produce the desired non-linear y_cap.

So, we introduce additional columns x2, x3, x4 such that

x2 = x1 * x1
x3 = x1 * x1 * x1
x4 = x1 * x1 * x1 * x1

Note: The method is still Linear Regression. It is **Linear Regression of non-linear independent variables**.

The resulting regression method is known as **Polynomial Regression** - since polynomial terms are introduced as independent variable to handle non-linearity in y.

In general, introducing additional **x** variables to improve the performance of ML methods is known as **Feature Engineering**.

Hence, **Polynomial Regression** can be said to be an application of the **Feature Engineering** technique.

You are encouraged to create a dataset that is not good for being regressed by a line, but gets adequately represented by a Polynomial Regression.

After introducing the polynomial terms and carrying out MLR, the results are as follows:

- The first chart shows y and y_cap (blue and organge)
- The second chart shows the error scatter plot
- We observe that the R2 value is now quite good and all the p-values, except that for x4, are much less than 0.05. This indicates that x4 is not significant and needs to be dropped.

	A	B	C	D	E	F	G	H
1	y	x1	x2	x3	x4		ycap	ei
2	0.038116834	0	0	0	0		0.31406	-0.27594
3	0.896467788	0.005555556	3.09E-05	1.71E-07	9.53E-10		0.381398	0.51507
4	0.159545792	0.011111111	0.000123	1.37E-06	1.52E-08		0.446403	-0.28686
5	0.863764416	0.016666667	0.000278	4.63E-06	7.72E-08		0.509106	0.354658
6	1.106349076	0.022222222	0.000494	1.1E-05	2.44E-07		0.569538	0.536811
7	1.010169458	0.027777778	0.000772	2.14E-05	5.95E-07		0.627729	0.38244
8	0.278498289	0.033333333	0.001111	3.7E-05	1.23E-06		0.683711	-0.40521
9	1.114230685	0.038888889	0.001512	5.88E-05	2.29E-06		0.737514	0.376717
10	1.029803908	0.044444444	0.001975	8.78E-05	3.9E-06		0.789169	0.240635
11	0.373869889	0.05	0.0025	0.000125	6.25E-06		0.838706	-0.46484
12	0.971634374	0.055555556	0.003086	0.000171	9.53E-06		0.886155	0.085479
13	0.975376766	0.061111111	0.003735	0.000228	1.39E-05		0.931548	0.043829
14	1.079774246	0.066666667	0.004444	0.000296	1.98E-05		0.974913	0.104861
15	1.242790434	0.072222222	0.005216	0.000377	2.72E-05		1.016282	0.226508
16	0.644698738	0.077777778	0.006049	0.000471	3.66E-05		1.055685	-0.41099
17	0.656177067	0.083333333	0.006944	0.000579	4.82E-05		1.09315	-0.43697
18	1.09549189	0.088888889	0.007901	0.000702	6.24E-05		1.128709	-0.03322
19	1.115274736	0.094444444	0.00892	0.000842	7.96E-05		1.162391	-0.04712
20	1.512547878	0.1	0.01	0.001	0.0001		1.194225	0.318323
21	0.639395626	0.105555556	0.011142	0.001176	0.000124		1.224242	-0.58485
22	1.406626554	0.111111111	0.012346	0.001372	0.000152		1.25247	0.154157
23	1.172479286	0.116666667	0.013611	0.001588	0.000185		1.278939	-0.10646
24	0.909355558	0.122222222	0.014938	0.001826	0.000223		1.303679	-0.39432
25	1.067679037	0.127777778	0.016327	0.002086	0.000267		1.326718	-0.25904
26	1.603060114	0.133333333	0.017778	0.00237	0.000316		1.348086	0.254974

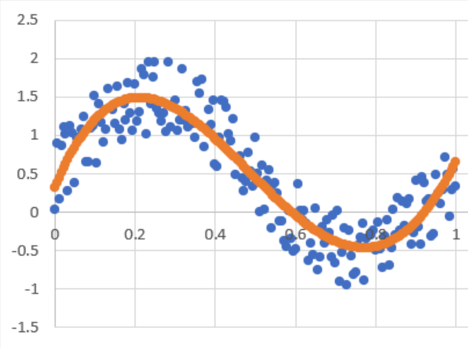
SUMMARY OUTPUT

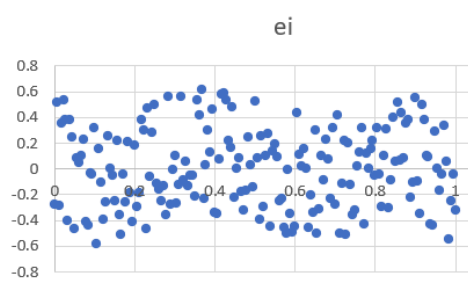
Regression Statistics	
Multiple R	0.913571
R Square	0.834612
Adjusted R	0.830853
Standard E	0.310737
Observatic	181

ANOVA

	df	*SS*	*MS*	*F*	*ignificance F*
Regressior	4	85.75865	21.43966	222.0403	1.26E-67
Residual	176	16.99412	0.096558		
Total	180	102.7528			

	Coefficients	*andard Errı*	*t Stat*	*P-value*	*Lower 95%*	*Uppe*
Intercept	0.31406	0.11176	2.810135	0.005513	0.093498	0.53
x1	12.33273	1.5577	7.917267	2.62E-13	9.258555	15.4
x2	-38.302	6.359842	-6.02247	9.77E-09	-50.8533	-25
x3	30.31208	9.568272	3.167978	0.00181	11.42877	49.1
x4	-4.00187	4.746218	-0.84317	0.400277	-13.3687	5.3





After dropping x4 from the model, the results are as follows:

- All p-values are now much lower than the threshold 0.05

This technique of starting off with all features and then **dropping** non-significant features one at a time is known as **Backward Feature Selection / Engineering.**

Backward feature engineering is a feature selection technique that removes features one by one until the model performance reaches a peak, and it is used to optimize the performance of the machine learning model by only including the most affecting feature and removing the least affecting feature.

y	x1	x2	x3	x4	ycap	ei
0.038116834	0	0	0	0	0.369343	-0.33123
0.896467788	0.005555556	3.09E-05	1.71E-07	9.53E-10	0.430539	0.465929
0.159545792	0.011111111	0.000123	1.37E-06	1.52E-08	0.48971	-0.33016
0.863764416	0.016666667	0.000278	4.63E-06	7.72E-08	0.54688	0.316884
1.106349076	0.022222222	0.000494	1.1E-05	2.44E-07	0.602072	0.504278
1.010169458	0.027777778	0.000772	2.14E-05	5.95E-07	0.655307	0.354862
0.278498289	0.033333333	0.001111	3.7E-05	1.23E-06	0.706611	-0.42811
1.114230685	0.038888889	0.001512	5.88E-05	2.29E-06	0.756005	0.358226
1.029803908	0.044444444	0.001975	8.78E-05	3.9E-06	0.803512	0.226292
0.373869889	0.05	0.0025	0.000125	6.25E-06	0.849155	-0.47529
0.971634374	0.055555556	0.003086	0.000171	9.53E-06	0.892958	0.078676
0.975376766	0.061111111	0.003735	0.000228	1.39E-05	0.934943	0.040434
1.079774246	0.066666667	0.004444	0.000296	1.98E-05	0.975133	0.104641
1.242790434	0.072222222	0.005216	0.000377	2.72E-05	1.013552	0.229239
0.644698738	0.077777778	0.006049	0.000471	3.66E-05	1.050221	-0.40552
0.656177067	0.083333333	0.006944	0.000579	4.82E-05	1.085164	-0.42899
1.09549189	0.088888889	0.007901	0.000702	6.24E-05	1.118405	-0.02291
1.115274736	0.094444444	0.00892	0.000842	7.96E-05	1.149965	-0.03469
1.512547878	0.1	0.01	0.001	0.0001	1.179868	0.33268
0.639395626	0.105555556	0.011142	0.001176	0.000124	1.208137	-0.56874
1.406626554	0.111111111	0.012346	0.001372	0.000152	1.234795	0.171832
1.172479286	0.116666667	0.013611	0.001588	0.000185	1.259864	-0.08738
0.909355558	0.122222222	0.014938	0.001826	0.000223	1.283368	-0.37401
1.067679037	0.127777778	0.016327	0.002086	0.000267	1.30533	-0.23765
1.603060114	0.133333333	0.017778	0.00237	0.000316	[illegible]	[illegible]
1.367903685	0.138888889	0.01929	0.002679	0.000372	1.344718	0.023186

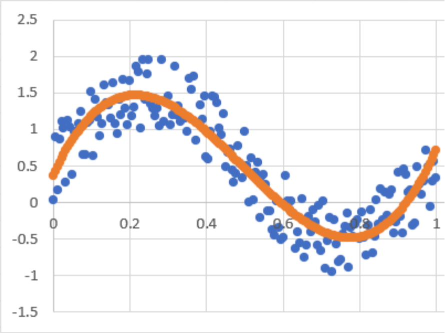
SUMMARY OUTPUT

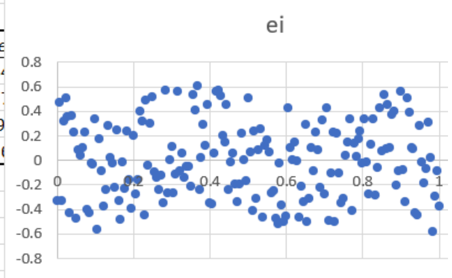
Regression Statistics	
Multiple R	0.913205
R Square	0.833943
Adjusted R	0.831129
Standard E	0.310483
Observatic	181

ANOVA

	df	SS	MS	F	ignificance F
Regressior	3	85.69001	28.56334	296.3007	9.58E-69
Residual	177	17.06277	0.0964		
Total	180	102.7528			

	Coefficients	andard Err	t Stat	P-value	Lower 95%	Uppe
Intercept	0.369343	0.090432	4.084204	6.69E-05	0.190879	0.54
x1	11.19878	0.785338	14.25982	3.25E-31	9.648946	12.7
x2	-33.1662	1.827791	-18.1455	3.01E-42	-36.7732	-29
x3	22.30833	1.201303	18.57012	2.04E-43	19.93761	24.0





Backward v/s Forward Feature Engineering

Forward Feature Engineering	Backward Feature Engineering
Starts with an empty feature set and iteratively adds one feature at a time based on their performance	Starts with a complete set of features and removes features one by one until the model performance reaches a peak
Goal is to identify the most accurate and informative features that contribute to the predictive power of the model	Goal is to identify the most accurate and relevant features that can be used in a model
Iteratively adds features to the model	Iteratively removes features from the model
Can be a more time-consuming process than backward feature engineering	Can be a more systematic approach than forward feature engineering
Can be useful when the number of features is relatively small	Can be useful when the number of features is relatively large
Can be prone to overfitting if too many features are added to the model	Can be prone to underfitting if too many features are removed from the model
Can be used in combination with backward feature engineering to optimize the feature selection process	Can be used in combination with forward feature engineering to optimize the feature selection process

In summary, forward feature engineering and backward feature engineering are two techniques used in machine learning for selecting relevant features to include in a model. Forward feature engineering starts with an empty feature set and iteratively adds one feature at a time based on their performance, while backward feature engineering starts with a complete set of features and removes features one by one until the model performance reaches a peak. Both techniques have their advantages and disadvantages and can be used in combination to optimize the feature selection process.

Exercise-3

- Try Backward Feature Elimination by adding polynomial and other relevant functions as base features to the data set in **non-linear-data-set-for-regression.csv**
- Try the Forward Feature Selection method for the same dataset
- Try the mixed approach (forward + backward) feature selection on the dataset.

Exercise-4

- Perform Linear Regression by adding appropriate features (polynomial / others) to the uploaded dataset **sine-segment-perturbed.csv.** What conclusions can you make?