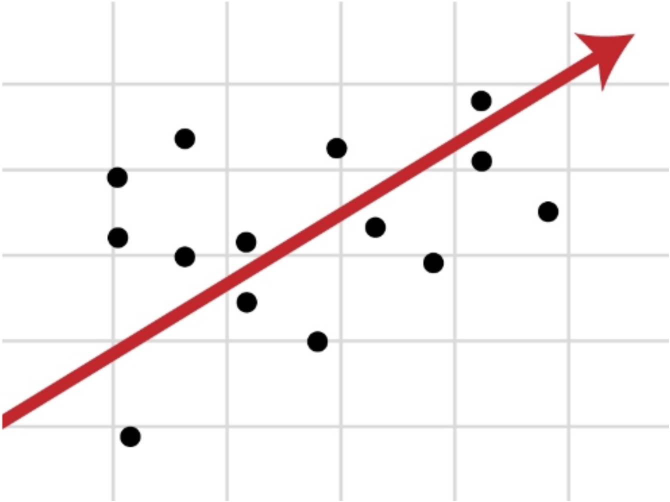
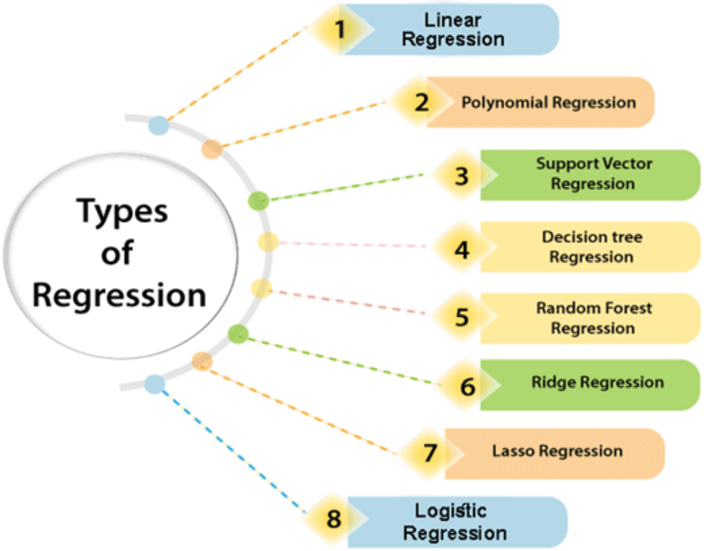
What is Regression Analysis?

Regression analysis is a statistical method used to examine the relationship between one dependent variable and one or more independent variables. The goal of regression analysis is to understand how the dependent variable changes when one or more independent variables are altered, and to create a model that can predict the value of the dependent variable based on the values of the independent variables.



Types of Linear Regression:

1. Linear Regression
2. Polynomial Regression
3. Support vector Regression
4. Decision tree Regression
5. Random Forest Regression
6. Ridge Regression
7. Lasso Regression
8. Logistic Regression



Step-by-Step process for conducting a linear regression analysis :

1. **Define the research question**: Identify the dependent variable (the variable you want to predict or explain) and the independent variable(s) (the variables that you think influence the dependent variable).

2. **Collect and prepare data**: Gather data for the dependent and independent variables. The data should be organized in a tabular format, with each row representing an observation and each column representing a variable. It's essential to clean and pre-process the data to handle missing values, outliers, and other potential issues that may affect the analysis.

3. **Visualize the data**: Before fitting a linear regression model, it's helpful to create scatter plots to visualize the relationship between the dependent variable and each independent variable. This can help you identify trends, outliers, and any potential issues with the data.

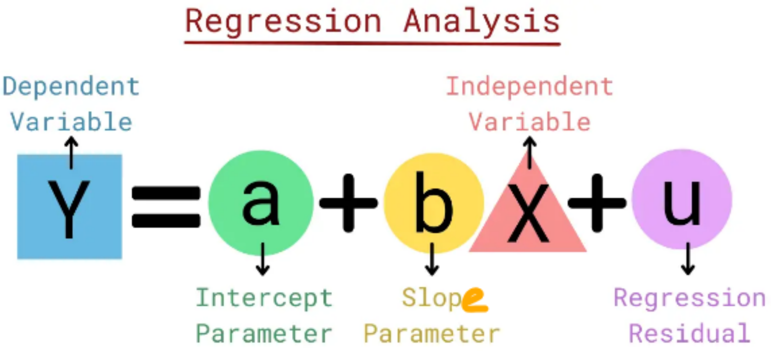
4. **Check assumptions**: Linear regression has some underlying assumptions, including linearity, independence of errors, homoscedasticity (constant variance of errors), and normality of errors. You can use diagnostic plots and statistical tests to check whether these assumptions hold for your data.

5. **Fit the linear regression model**: Use statistical software (e.g., R, Python, or Excel) to fit a linear regression model to your data. The model will estimate the regression coefficients (intercept and slope) that minimize the sum of squared residuals (i.e., the differences between the observed and predicted values of the dependent variable).

6. **Interpret the model**: Analyse the estimated regression coefficients, their standard errors, t-values, and p-values to determine the statistical significance of the relationship between the dependent and independent variables. The R-squared value and adjusted R-squared value can provide insights into the goodness-of-fit of the model and the proportion of variation in the dependent variable explained by the independent variables.

7. **Validate the model**: If you have a sufficiently large dataset, you can split it into a training and testing set. Fit the linear regression model to the training set, and then use the model to predict the dependent variable in the testing set. Calculate the mean squared error, root mean squared error, or another performance metric to assess the predictive accuracy of the model.

8. **Report results**: Summarize the findings of the linear regression analysis in a clear and concise manner, including the estimated coefficients, their interpretation, and any limitations or assumptions that may impact the results



What's the statistics connection? with Linear regression

Linear regression is a statistical technique used to model the relationship between a dependent variable and one or more independent variables. It is a fundamental tool in statistics for understanding and analyzing relationships between variables. Here are some key connections between linear regression and statistics:

1. **Estimation**: Linear regression estimates the parameters (coefficients) that best fit the data using statistical methods like ordinary least squares.

2. **Inference**: Statistical inference assesses the significance and reliability of the relationships between variables through tests like t-tests and p-values.

3. **Assumptions**: Linear regression relies on assumptions such as linearity, independence of errors, homoscedasticity, and normality of errors, which are checked using statistical tests and diagnostic plots.

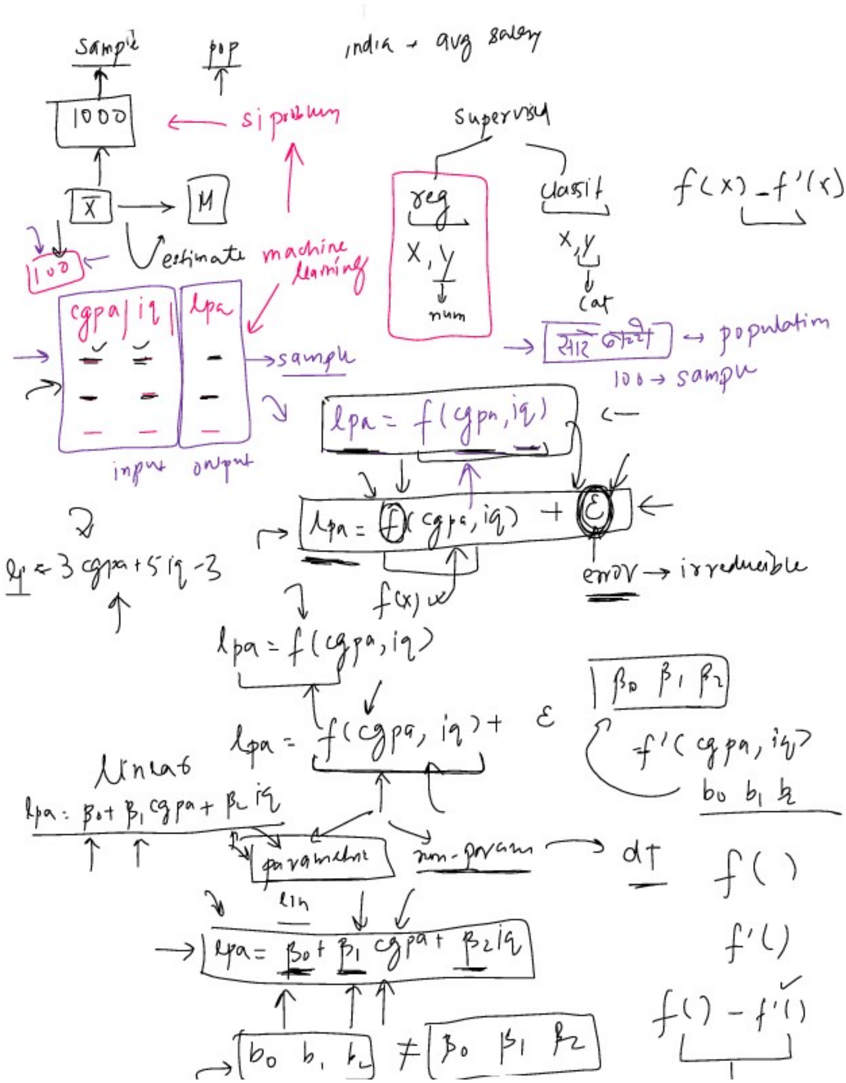
4. **Goodness-of-fit**: Statistics provides measures like R-squared and adjusted R-squared to evaluate how well the linear regression model fits the data.

5. **Predictive accuracy**:Linear regression can be used for prediction, and statistical metrics like mean squared error or root mean squared error assess the accuracy of the predictions.
6. **Interpretation**: Statistical tools interpret the results of linear regression, including estimated coefficients, standard errors, t-values, and p-values, to understand the strength and significance of relationships between variables.

Overall, linear regression is a statistical technique that relies on statistical principles and methods for estimation, inference, validation, and interpretation. It provides a rigorous framework for analyzing and understanding the relationships between variables based on observed data.

Why ML problems are a Statistical Inference Problems? With Example

Explanation:



$$\text{lpa} = f'(\text{cgpa}, \text{iq}) + \boxed{\text{reducible}} + \underset{\uparrow}{\varepsilon} \qquad \boxed{\text{reducible}}$$

estimate rel^n of x and y based on given data

$$f'(\,) \simeq f(\,)$$

↑ true of x, y for p.pn

ε

$$\boxed{y = \boxed{2x-5} + \boxed{\text{some randomness}}}$$

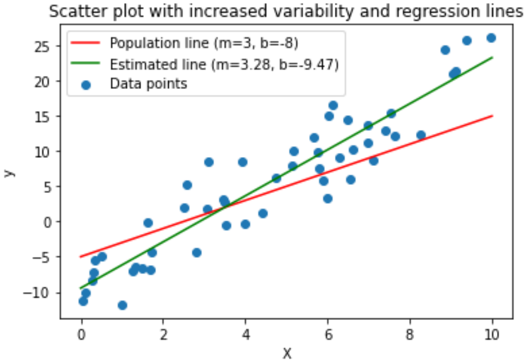
$$\underline{f}(x) = \underbrace{2x-5}_{\uparrow\ \ \uparrow} \qquad \boxed{\begin{matrix}\underline{\beta}_0 = -5 \\ \underline{\beta}_1 = 2\end{matrix}}$$

pop parameter

$$\underline{\boxed{b_0 \quad b_1}} \rightarrow \text{current set of 50 points}$$

In [2]:

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

# Generate the data
x = 10 * np.random.rand(50)
y = 3 * x - 8 + np.random.randn(50) * 4

# Fit a linear regression model
x = x.reshape(-1, 1)
model = LinearRegression()
model.fit(x, y)

# Calculate the predicted values
y_pred = model.predict(x)

# Plot the scatter plot and regression lines
plt.scatter(x, y, label="Data points")
plt.xlabel("X")
plt.ylabel("y")
plt.title("Scatter plot with increased variability and regression lines")

# Plot the actual population line
x_line = np.linspace(0, 10, 100)
y_actual = 2 * x_line - 5
plt.plot(x_line, y_actual, 'r', label="Population line (m=3, b=-8)")

# Plot the estimated regression line
y_estimated = model.coef_[0] * x_line + model.intercept_
plt.plot(x_line, y_estimated, 'g', label=f"Estimated line (m={model.coef_[0]:.2f}, b={model

# Add Legend and show the plot
plt.legend()
plt.show()
```



Why is Regression Analysis required?

1. **Relationship identification**: Regression analysis identifies and quantifies the relationship between variables.

2. **Prediction and forecasting**: Regression analysis builds models to predict or forecast the dependent variable based on the independent variables.

3. **Causal inference**: Regression analysis explores causal relationships between variables.

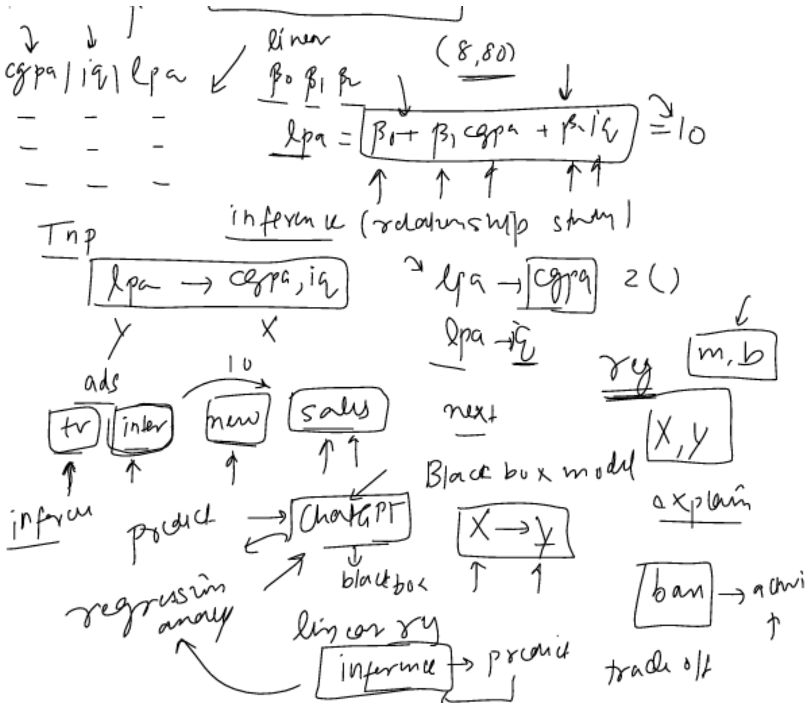
4. **Variable selection and feature importance**: Regression analysis determines the most important variables in explaining the dependent variable.

5. **Model evaluation and goodness-of-fit**: Regression analysis assesses how well the model fits the data.

6. **Hypothesis testing**: Regression analysis tests the statistical significance of relationships between variables.

Inference Vs Prediction [Why regression analysis is required?]

Explanation:

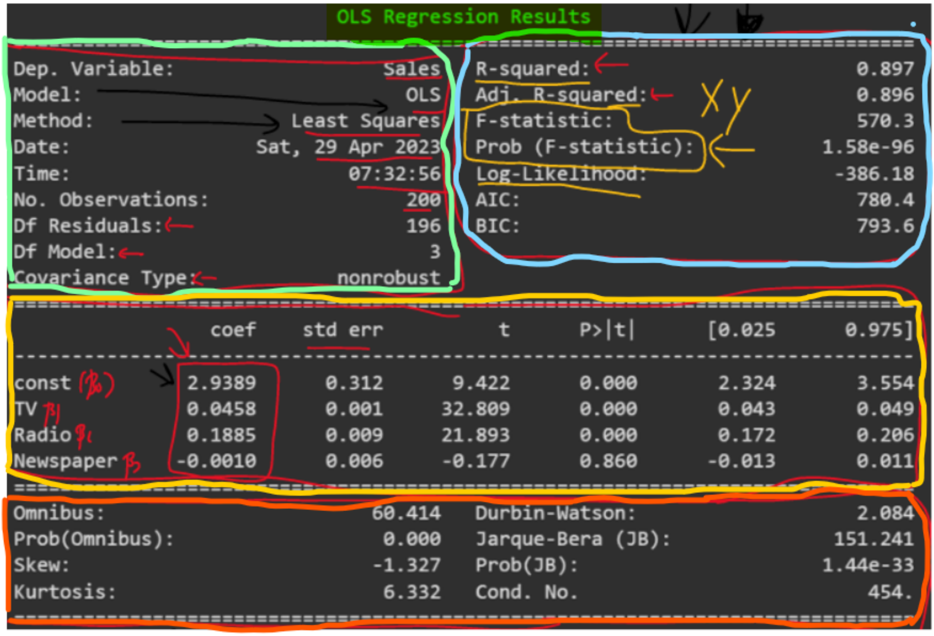


Statsmodel Linear Regression

In [3]:

```python
import pandas as pd
import statsmodels.api as sm

# Load the dataset
url = "https://raw.githubusercontent.com/justmarkham/scikit-learn-videos/master/data/Advert
data = pd.read_csv(url, index_col=0)

# Define the independent variables (add a constant for the intercept)
X = data[['TV', 'Radio', 'Newspaper']]
X = sm.add_constant(X)

# Define the dependent variable
y = data['Sales']

# Fit the model using the independent and dependent variables
model = sm.OLS(y, X).fit()

# Print the summary of the model
print(model.summary())
```

```
                            OLS Regression Results
============================================================================
==
Dep. Variable:                  Sales   R-squared:                       0.8
97
Model:                            OLS   Adj. R-squared:                  0.8
96
Method:                 Least Squares   F-statistic:                     57
0.3
Date:                Sat, 08 Jul 2023   Prob (F-statistic):           1.58e-
96
Time:                        11:33:27   Log-Likelihood:                -386.
18
No. Observations:                 200   AIC:                             78
0.4
Df Residuals:                     196   BIC:                             79
3.6
Df Model:                           3
Covariance Type:            nonrobust
============================================================================
==
                 coef    std err          t      P>|t|      [0.025      0.97
5]
----------------------------------------------------------------------------
--
const          2.9389      0.312      9.422      0.000       2.324       3.5
54
TV             0.0458      0.001     32.809      0.000       0.043       0.0
49
Radio          0.1885      0.009     21.893      0.000       0.172       0.2
06
Newspaper     -0.0010      0.006     -0.177      0.860      -0.013       0.0
11
============================================================================
==
Omnibus:                       60.414   Durbin-Watson:                   2.0
84
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              151.2
41
Skew:                          -1.327   Prob(JB):                     1.44e-
33
Kurtosis:                       6.332   Cond. No.                         45
4.
============================================================================
==

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is corre
ctly specified.

C:\Users\user\anaconda3\lib\site-packages\statsmodels\tsa\tsatools.py:142: F
utureWarning: In a future version of pandas all arguments of concat except f
or the argument 'objs' will be keyword-only
  x = pd.concat(x[::order], 1)
```



```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  Sales   R-squared:                       0.897
Model:                            OLS   Adj. R-squared:                  0.896
Method:                 Least Squares   F-statistic:                     570.3
Date:                Sat, 29 Apr 2023   Prob (F-statistic):           1.58e-96
Time:                        07:32:56   Log-Likelihood:                -386.18
No. Observations:                 200   AIC:                             780.4
Df Residuals:                     196   BIC:                             793.6
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          2.9389      0.312      9.422      0.000       2.324       3.554
TV             0.0458      0.001     32.809      0.000       0.043       0.049
Radio          0.1885      0.009     21.893      0.000       0.172       0.206
Newspaper     -0.0010      0.006     -0.177      0.860      -0.013       0.011
==============================================================================
Omnibus:                       60.414   Durbin-Watson:                   2.084
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              151.241
Skew:                          -1.327   Prob(JB):                     1.44e-33
Kurtosis:                       6.332   Cond. No.                         454.
==============================================================================
```

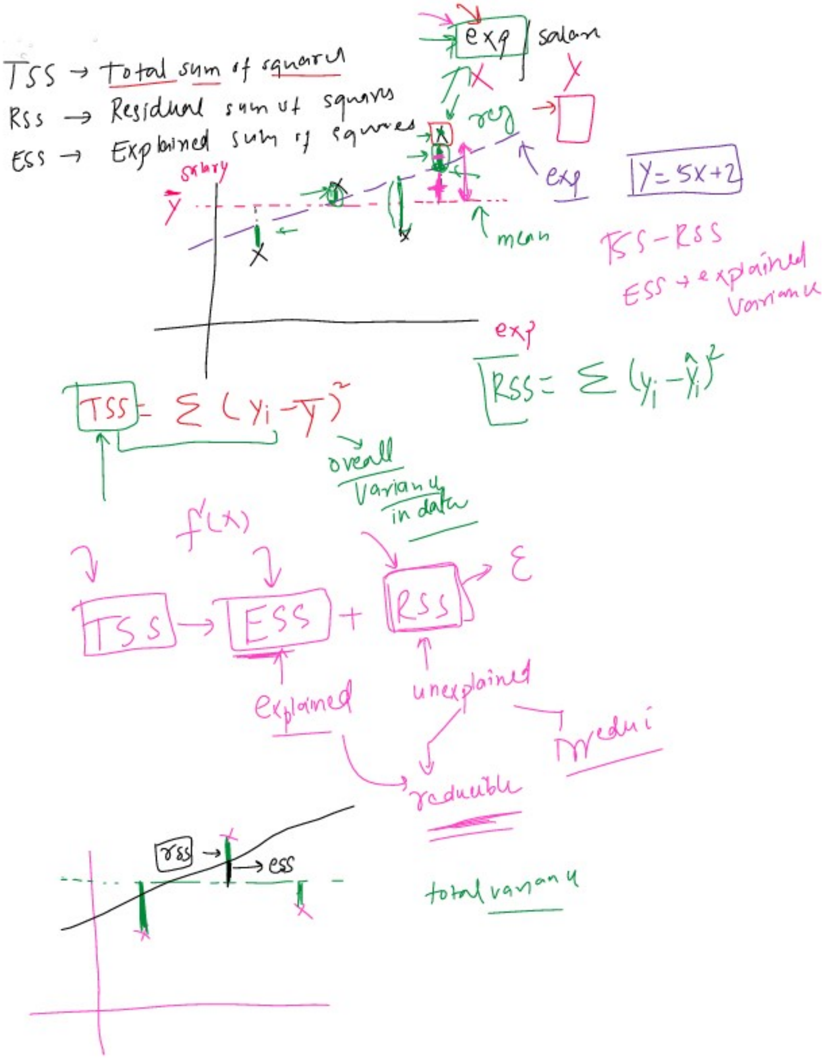
What is TSS, RSS and ESS ?

1. **Total Sum of Squares (TSS)**: TSS measures the total variation in the dependent variable.

2. **Residual Sum of Squares (RSS)**: RSS quantifies the unexplained variation in the dependent variable.

3. **Explained Sum of Squares (ESS)**: ESS represents the portion of the total variation in the dependent variable that is explained by the independent variables.

Explanation:



TSS → Total sum of squares

RSS → Residual sum of squares

ESS → Explained sum of squares

$Y = 5X+2$

TSS−RSS

ESS → explained variance

$RSS = \sum (y_i - \hat{y}_i)^2$

$TSS = \sum (y_i - \bar{y})^2$

overall variance in data

$TSS \rightarrow ESS + RSS \rightarrow \varepsilon$

explained

unexplained

reducible

irredui

what is Degree of Freedom ?

In linear regression, the total degrees of freedom (df_total) represent the total number of data points minus 1. It represents the overall variability in the dataset that can be attributed to both the model and the residuals.

For a linear regression with n data points (observations), the total degrees of freedom can be calculated as:

- **df_total = n - 1**

where: n is the number of data points (observations) in the dataset

The total degrees of freedom in linear regression is divided into two components:

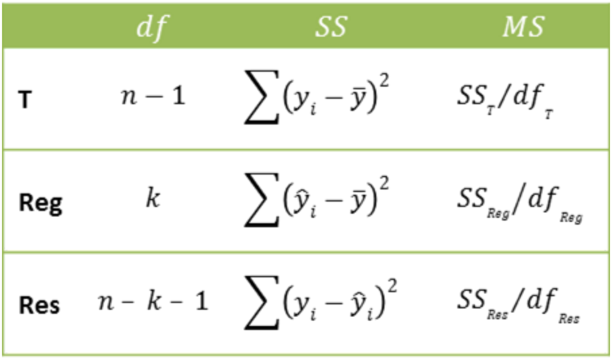
1. **Degrees of freedom for the model (df_model)**: This is equal to the number of independent variables in the model (k).

2. **Degrees of freedom for the residuals (df_residuals)**: The degrees of freedom for the residuals indicate the number of independent pieces of information that are available for estimating the variability in the residuals (errors) after fitting the regression model.

This is equal to the number of data points (n) minus the number of estimated parameters, including the intercept (k+1).

- The sum of the degrees of freedom for the model and the degrees of freedom for the residuals is equal to the total degrees of freedom:

df_total = df_model + df_residuals



	df	SS	MS
T	$n-1$	$\sum(y_i - \bar{y})^2$	SS_T/df_T
Reg	k	$\sum(\hat{y}_i - \bar{y})^2$	SS_{Reg}/df_{Reg}
Res	$n-k-1$	$\sum(y_i - \hat{y}_i)^2$	SS_{Res}/df_{Res}

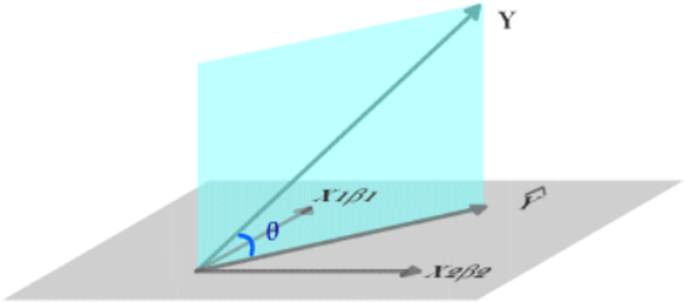
In linear regression, the calculation of degrees of freedom depends on the number of observations (sample size) and the number of predictor variables (independent variables) in the model. Here's how degrees of freedom are typically determined in linear regression:

1. **Degrees of freedom for regression**: The degrees of freedom for the regression model is equal to the number of predictor variables (p) in the model. This represents the number of independent pieces of information available to estimate the regression coefficients.

2. **Degrees of freedom for residual (error)**: The degrees of freedom for residual, also known as the degrees of freedom for error, is calculated as the difference between the total sample size (n) and the number of predictor variables (p) and the intercept term (1) if it is included in the model. It is given by df = n - p - 1.

The degrees of freedom for regression and residual are used in various calculations and statistical tests associated with linear regression. For example, they are used to determine the appropriate critical values for hypothesis testing, calculate the mean squared error, perform model diagnostics, and assess the statistical significance of the regression coefficients.

It's worth noting that the degrees of freedom can vary if there are specific constraints or assumptions imposed on the regression model. However, in the typical case of linear regression, the degrees of freedom are determined as described above.



In statistics , degrees of freedom (df) refers to the number of independent pieces of information available in a sample or dataset. The concept of degrees of freedom is used in various statistical tests and calculations. The calculation of degrees of freedom can vary depending on the specific statistical procedure or test being performed. Here are some common examples:

1. **Degrees of freedom in a sample**: In a dataset of size n, the degrees of freedom for calculating sample statistics (such as the sample mean or sample variance) is n - 1. This adjustment accounts for the fact that one degree of freedom is lost when estimating the population parameters from the sample.

2. **Degrees of freedom in a t-test**: In a two-sample t-test, the degrees of freedom are calculated as df = n1 + n2 - 2, where n1 and n2 represent the sample sizes of the two groups being compared. Again, the adjustment of -2 accounts for the two estimated parameters (mean) in the t-test.

3. **Degrees of freedom in chi-square tests**: In chi-square tests, the degrees of freedom are determined by the number of categories or groups involved. For a chi-square test of independence, the degrees of freedom are calculated as (number of rows - 1) multiplied by (number of columns - 1).

4. **Degrees of freedom in regression**: In linear regression, the degrees of freedom are based on the sample size and the number of predictor variables. Specifically, the degrees of freedom for residual (error) is n - p - 1, where n is the sample size and p is the number of predictor variables.

It's important to note that degrees of freedom play a crucial role in determining the appropriate critical values and interpreting the results of statistical tests. The specific calculation of degrees of freedom depends on the context and statistical procedure being used.

`https://youtu.be/rATNoxKg1yA`

what is F-statistic & Prob(F-statistic) ?

The F-test for overall significance is a statistical test used to determine whether a linear regression model is statistically significant, meaning it provides a better fit to the data than just using the mean of the dependent variable.

$$F - ratio \rightarrow \qquad F = \frac{MSR}{MSE} = \frac{\frac{SSR}{df_{MSR}}}{\frac{SSE}{df_{MSE}}}$$

$$degree\ of\ freedom \rightarrow \qquad df_{MSR} = p \qquad df_{MSE} = n - p - 1$$

Here are the steps involved in conducting an F-test for overall significance:

1.State the null and alternative hypotheses:

- **Null hypothesis (H0)**: All regression coefficients (except the intercept) are equal to zero (β1 = β2 = ... = βk = 0), meaning that none of the independent variables contribute significantly to the explanation of the dependent variable's variation.
- **Alternative hypothesis (H1)**: At least one regression coefficient is not equal to zero, indicating that at least one independent variable contributes significantly to the explanation of the dependent variable's variation.

2. Fit the linear regression model to the data, estimating the regression coefficients (intercept and slopes).

3. Calculate the Sum of Squares (SS) values:

- **Total Sum of Squares (TSS)**: The sum of squared differences between each observed value of the dependent variable and its mean.
- **Regression Sum of Squares (ESS)**: The sum of squared differences between the predicted values of the dependent variable and its mean.
- **Residual Sum of Squares (RSS)**: The sum of squared differences between the observed values and the predicted values of the dependent variable.

4.Compute the Mean Squares (MS) values:

- **Mean Square Regression (MSR)**: ESS divided by the degrees of freedom for the model (df_model), which is the number of independent variables (k). This could also be called as Average Explained Variance per independent feature.
- **Mean Square Error (MSE)**: RSS divided by the degrees of freedom for the residuals (df_residuals), which is the number of data points (n) minus the number of estimated parameters, including the intercept (k+1). This could also be called as average unexplained variance per degree of freedom.

5. Calculate the F-statistic: F-statistic = MSR / MSE

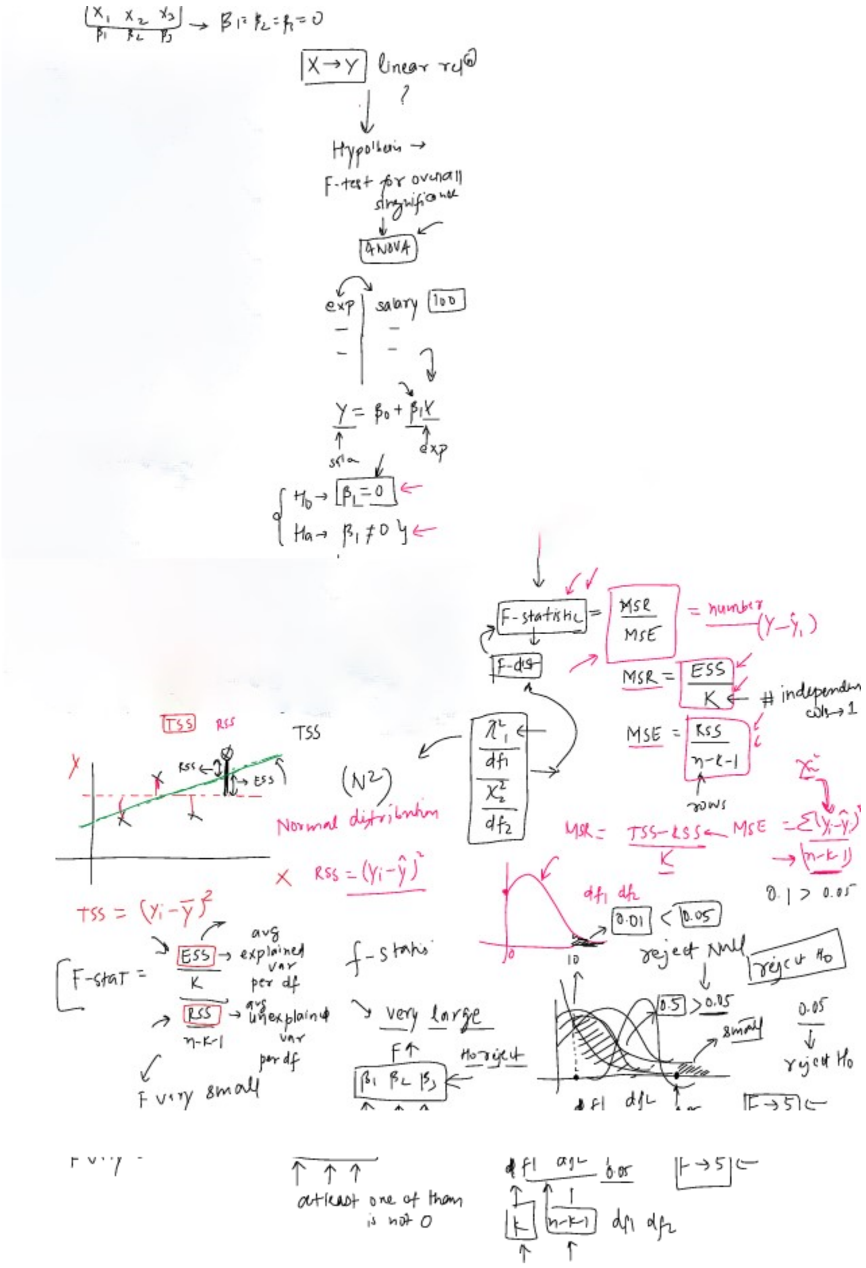
6. **Determine the p-value**: Compute the p-value associated with the calculated F-statistic using the F-distribution or a statistical software package.

7. **Compare the calculated F-statistic to the p-value to the chosen significance level (α)**:

- If the p-value < α, reject the null hypothesis. This indicates that at least one independent variable contributes significantly to the prediction of the dependent variable, and the overall regression model is statistically significant.

- If the p-value ≥ α, fail to reject the null hypothesis. This suggests that none of the independent variables in the model contribute significantly to the prediction of the dependent variable, and the overall regression model is not statistically significant.

Following these steps, you can perform an F-test for overall significance in a linear regression analysis and determine whether the regression model is statistically significant.



$\frac{X_1 \; X_2 \; X_3}{\beta_1 \; \beta_2 \; \beta_3} \rightarrow \beta_1 = \beta_2 = \beta_3 = 0$

$\boxed{X \rightarrow Y}$ linear rel^n ?

Hypothesis →

F-test for overall significance

ANOVA

exp | salary $\boxed{100}$

$Y = \beta_0 + \beta_1 X$

sal exp

$H_0 \rightarrow \boxed{\beta_1 = 0}$

$H_a \rightarrow \beta_1 \neq 0$

$\boxed{\text{F-statistic}} = \boxed{\frac{MSR}{MSE}}$ = number

$(Y - \hat{Y}_1)$

F-dis

$MSR = \boxed{\frac{ESS}{K}}$ ← # independent cols → 1

$MSE = \boxed{\frac{RSS}{n-k-1}}$

rows

$\frac{\chi_1^2 / df_1}{\chi_2^2 / df_2}$

(N^2)

Normal distribution

TSS RSS

TSS

$RSS = (y_i - \hat{y})^2$

$TSS = (y_i - \bar{y})^2$

$MSR = \frac{TSS - RSS}{K}$

$MSE = \frac{\Sigma(y_i - \hat{y}_i)^2}{(n-k-1)}$

χ^2

$0.1 > 0.05$

df1 df2

$\boxed{0.01} < \boxed{0.05}$

reject Null

reject H_0

$$\text{F-stat} = \frac{\boxed{ESS} / K}{\boxed{RSS} / (n-k-1)}$$

ESS/K → avg explained var per df

RSS/(n−k−1) → avg unexplained var per df

f-stats → very large

F↑

H_0 reject

$\boxed{\beta_1 \; \beta_2 \; \beta_3}$

at least one of them is not 0

F very small

$\boxed{0.5} > 0.05$

small

0.05 ↓ reject H_0

$\boxed{F \rightarrow 5}$

df1 df2 0.05

$\boxed{k}$ $\boxed{n-k-1}$ df1 df2

what is R-squared ?

R-squared (R2), also known as the **coefficient of determination**, is a measure used in regression analysis to assess the goodness-of-fit of a model. It quantifies the proportion of the variance in the dependent variable (response variable) that can be explained by the independent variables (predictor variables) in the regression model. **R-squared is a value between 0 and 1**, with higher values indicating a better fit of the model to the observed data.

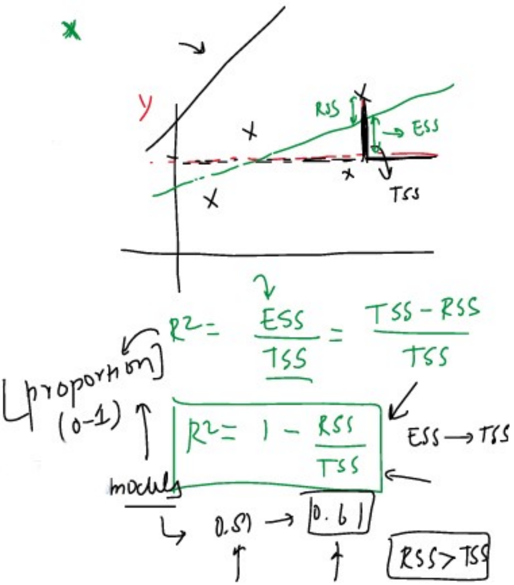
In the context of a simple linear regression, R2 is calculated as the square of the correlation coefficient (r) between the observed and predicted values.

- In multiple regression,R2 is obtained from the ratio of the explained sum of squares (ESS) to the total sum of squares (TSS):

R2 = ESS / TSS

where:

- ESS (Explained Sum of Squares) is the sum of squared differences between the predicted values and the mean of the observed values. It represents the variation in the response variable that can be explained by the predictor variables in the model.
- TSS (Total Sum of Squares) is the sum of squared differences between the observed values and the mean of the observed values. It represents the total variation in the response variable.


y
RSS
ESS
TSS
$R^2 = \frac{ESS}{TSS} = \frac{TSS - RSS}{TSS}$
proportion
(0-1)
$R^2 = 1 - \frac{RSS}{TSS}$
ESS → TSS
models
0.51 → 0.61
RSS>TSS

An R-squared value of 0 indicates that the model does not explain any of the variance in the response variable, while an R-squared value of 1 indicates that the model explains all of the variance. However,

- **R-squared can be misleading in some cases, especially when the number of predictor variables is large or when the predictor variables are not relevant to the response variable.**

What is Adjusted R-squared?

Adjusted R-squared is a modified version of R-squared (R2) that **adjusts for the number of predictor variables in a multiple regression model. It provides a more accurate measure of the goodness-of-fit of a model by considering the model's complexity.**

In a multiple regression model, R-squared (R2) measures the proportion of variance in the response variable that is explained by the predictor variables. However, R-squared always increases or stays the same with the addition of new predictor variables, regardless of whether those variables contribute valuable information to the model. This can lead to overfitting, where a model becomes too complex and starts capturing noise in the data instead of the underlying relationships.

Adjusted R-squared accounts for the number of predictor variables in the model and the sample size, **penalizing the model** for adding unnecessary complexity. Adjusted R-squared can decrease when an irrelevant predictor variable is added to the model, making it a better metric for comparing models with different numbers of predictor variables.

$$Adjusted\ R^2 = 1 - \frac{(1 - R^2)(N - 1)}{N - p - 1}$$

Where

R^2 Sample R-Squared

N Total Sample Size

p Number of independent variable

By using adjusted R-squared, you can more accurately assess the goodness-of-fit of a model and choose the optimal set of predictor variables for your analysis.

In [ ]:

```
# Pratical
```

In [5]:

```
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score

# Generate synthetic data
np.random.seed(42)
n = 100
x1 = np.random.normal(0, 1, n)
x2 = np.random.normal(0, 1, n)
irrelevant_predictors = np.random.normal(0, 1, (n, 10))

y = 2 * x1 + 3 * x2 + np.random.normal(0, 1, n)

# Helper function to calculate adjusted R-squared
def adjusted_r2(r2, n, k):
    return 1 - (1 - r2) * (n - 1) / (n - k - 1)

# Fit linear regression models with different predictors
X = pd.DataFrame({'x1': x1, 'x2': x2})
X_with_irrelevant = pd.concat([X] + [pd.Series(irrelevant_predictors[:, i], name=f"irreleva
```

In [4]:

```
X
```

Out[4]:

```
          x1         x2
 0   0.496714  -1.415371
 1  -0.138264  -0.420645
 2   0.647689  -0.342715
 3   1.523030  -0.802277
 4  -0.234153  -0.161286
...       ...        ...
95  -1.463515   0.385317
96   0.296120  -0.883857
97   0.261055   0.153725
98   0.005113   0.058209
99  -0.234587  -1.142970

100 rows × 2 columns
```

In [2]:

```
X_with_irrelevant
```

Out[2]:

	x1	x2	irrelevant_0	irrelevant_1	irrelevant_2	irrelevant_3	irrelevant_4	irreleva
0	0.496714	-1.415371	0.357787	0.560785	1.083051	1.053802	-1.377669	-0.93
1	-0.138264	-0.420645	0.570891	1.135566	0.954002	0.651391	-0.315269	0.75
2	0.647689	-0.342715	2.314659	-1.867265	0.686260	-1.612716	-0.471932	1.08
3	1.523030	-0.802277	-0.730367	0.216459	0.045572	-0.651600	2.143944	0.63
4	-0.234153	-0.161286	-0.792521	-0.114736	0.504987	0.865755	-1.200296	-0.33
...	...	...	...	...	...	...	...	
95	-1.463515	0.385317	-0.991392	-2.153390	-0.638962	-1.323090	1.642015	1.00
96	0.296120	-0.883857	-2.499406	2.290943	-1.389572	-1.645399	1.022570	2.43
97	0.261055	0.153725	0.758929	0.281191	0.104201	-0.062593	-0.753965	-0.28
98	0.005113	0.058209	0.179894	1.392002	0.918317	-1.570501	-0.989628	0.94
99	-0.234587	-1.142970	0.105376	-1.334025	-0.601368	0.319782	-1.592994	0.44

100 rows × 12 columns

In [7]:

```
# Importing necessary libraries
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score

# Fitting the models
model1 = LinearRegression().fit(X, y)
model2 = LinearRegression().fit(X_with_irrelevant, y)

# Calculate R-squared and adjusted R-squared for each model
models = [
    ('Model with relevant predictors', model1, X.shape[1]),
    ('Model with irrelevant predictors', model2, X_with_irrelevant.shape[1])
]

for name, model, k in models:
    # Calculate R-squared and adjusted R-squared
    r2 = r2_score(y, model.predict(X_with_irrelevant.iloc[:, :k]))
    adj_r2 = adjusted_r2(r2, n, k)

    # Print the results
    print(f"{name}: R-squared = {r2:.3f}, Adjusted R-squared = {adj_r2:.3f}")
```

```
Model with relevant predictors: R-squared = 0.912, Adjusted R-squared = 0.91
0
Model with irrelevant predictors: R-squared = 0.919, Adjusted R-squared = 0.
908
```

Which one should be used?

The choice between using R-squared and adjusted R-squared depends on the context and the goals of your analysis. Here are some guidelines to help you decide which one to use:

1. **Model comparison**: If you're comparing models with different numbers of predictor variables, it's better to use adjusted R-squared. This is because adjusted R-squared takes into account the complexity of the model, penalizing models that include irrelevant predictor variables. R-squared, on the other hand, can be misleading in this context, as it tends to increase with the addition of more predictor variables, even if they don't contribute valuable information to the model.

2. **Model interpretation**: If you're interested in understanding the proportion of variance in the response variable that can be explained by the predictor variables in the model, R- squared can be a useful metric. However, keep in mind that R-squared does not provide information about the significance or relevance of individual predictor variables. It's also important to remember that a high R-squared value does not necessarily imply causation or a good predictive model.

3. **Model selection and overfitting**: When building a model and selecting predictor variables, it's important to guard against overfitting. In this context, adjusted R-squared can be a helpful metric, as it accounts for the number of predictor variables and penalizes the model for unnecessary complexity. By using adjusted R-squared, you can avoid including irrelevant predictor variables that might lead to overfitting.

In summary, adjusted R-squared is generally more suitable when comparing models with different numbers of predictor variables or when you're concerned about overfitting. R- squared can be useful for understanding the overall explanatory power of the model,

but it should be interpreted with caution, especially in cases with many predictor variables or potential multicollinearity.

what is T-statistic?

Performing a t-test for a simple linear regression, including the intercept term and using the p-value approach, involves the following steps:

1.**For the slope coefficient (β1)**:

- Null hypothesis (H0): β1 = 0 (no relationship between the predictor variable (X) and the response variable (y))
- Alternative hypothesis (H1): β1 ≠ 0 (a relationship exists between the predictor variable and the response variable)

For the intercept coefficient (β0):

- Null hypothesis (H0): β0 = 0 (the regression line passes through the origin)
- Alternative hypothesis (H1): β0 ≠ 0 (the regression line does not pass through the origin)

2. **Estimate the slope and intercept coefficients (b0 and b1)**: Using the sample data, calculate the slope (b1) and intercept (b0) coefficients for the regression model.

3. **Calculate the standard errors for the slope and intercept coefficients (SE(b0) and SE(b1))**: Compute the standard errors of the slope and intercept coefficients using the following formulas:

$$SE(b_1) = \sqrt{\frac{\Sigma (y_i - \hat{y}_i)^2}{(n-2)\, \Sigma (x_i - \bar{x})^2}} \quad \longleftarrow$$

$$SE(b_0) = \sqrt{\frac{\Sigma (y_i - \hat{y}_i)^2}{(n-2)} \left[\frac{1}{n} + \frac{\bar{x}^2}{\Sigma (x_i - \bar{x})^2} \right]}$$

4. **Compute the t-statistics for the slope and intercept coefficients**: Calculate the t-statistics for the slope and intercept coefficients using the following formulas:

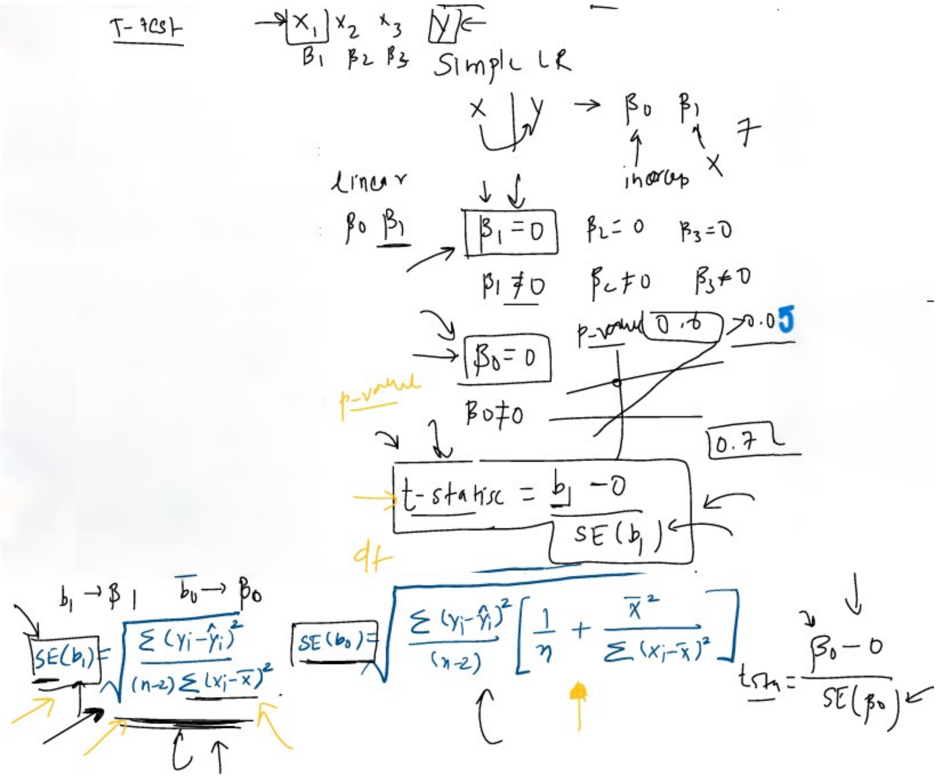
$$t\text{-}value_{b_0} = \frac{b_0 - 0}{SE(b_0)}$$

$$t\text{-}value_{b_1} = \frac{b_1 - 0}{SE(b_1)}$$

5. **Calculate the p-values for the slope and intercept coefficients**: Using the t-statistics and the degrees of freedom, look up the corresponding p-values from the t-distribution table or use a statistical calculator.

6. **Compare the p-values to the chosen significance level (α)**: A common choice for α is 0.05, which corresponds to a 95% confidence level. Compare the calculated p-values to α:

- If the p-value is less than or equal to α, reject the null hypothesis.
- If the p-value is greater than α, fail to reject the null hypothesis.



what is Confidence Intervals for Coefficients ?

1. **Estimate the slope and intercept coefficients (b0 and b1)**: Using the sample data, calculate the slope (b1) and intercept (b0) coefficients for the regression model.

2. **Calculate the standard errors for the slope and intercept coefficients (SE(b0) and SE(b1)):**

$$SE(b_1) = \sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{(n-2)\sum (x_i - \bar{x})^2}} \longleftarrow$$

$$SE(b_0) = \sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{(n-2)}\left[\frac{1}{n} + \frac{\bar{x}^2}{\sum (x_i - \bar{x})^2}\right]}$$

3. **Determine the degrees of freedom**: In a simple linear regression, the degrees of freedom (df) is equal to the number of observations (n) minus the number of estimated parameters (2: the intercept and the slope coefficient). df = n - 2

4. **Find the critical t-value**: Look up the critical t-value from the t- distribution table or use a statistical calculator based on the chosen confidence level (e.g., 95%) and the degrees of freedom calculated in step 3.

5. **Calculate the confidence intervals for the slope and intercept coefficients**: Compute the confidence intervals for the slope (b1) and intercept (b0) coefficients using the following formulas:

$$CI_{b_0} = \underline{b_0} \pm \underline{t\_value} * SE(b_0)$$

$$CI_{b_1} = \underline{b_1} \pm \underline{t\_value} * \underline{SE(b_1)}$$

These confidence intervals represent the range within which the true population regression coefficients are likely to fall with a specified level of confidence (e.g., 95%)

Explanation:

t- dist

significance
↓
0.05

↗ 95% prob

$$\underbrace{b_1 \pm 3.18 \times SE(b_1)}_{\text{lower + upper } b_1}$$

In [8]:

```
import pandas as pd

df = pd.DataFrame()

df['X'] = [2,3,5,5,7]
df['y'] = [2,4,3,5,5]

df
```

Out[8]:

	X	y
0	2	2
1	3	4
2	5	3
3	5	5
4	7	5

In [9]:

```
import statsmodels.api as sm


# Add a constant to the independent variable
X = sm.add_constant(df['X'])

# Fit the linear regression model
model = sm.OLS(df['y'], X).fit()

# Print the summary of the model
print(model.summary())
```

```
                             OLS Regression Results
==============================================================================
==
Dep. Variable:                       y   R-squared:                        0.5
30
Model:                             OLS   Adj. R-squared:                   0.3
73
Method:                  Least Squares   F-statistic:                      3.3
80
Date:                 Sat, 08 Jul 2023   Prob (F-statistic):               0.1
63
Time:                         13:38:09   Log-Likelihood:                 -5.97
69
No. Observations:                    5   AIC:                              15.
95
Df Residuals:                        3   BIC:                              15.
17
Df Model:                            1
Covariance Type:             nonrobust
==============================================================================
==
                  coef    std err          t      P>|t|      [0.025       0.97
5]
------------------------------------------------------------------------------
--
const           1.6579      1.253      1.323      0.278      -2.331        5.6
46
X               0.4868      0.265      1.839      0.163      -0.356        1.3
30
==============================================================================
==
Omnibus:                           nan   Durbin-Watson:                    3.4
82
Prob(Omnibus):                     nan   Jarque-Bera (JB):                 0.5
34
Skew:                           -0.047   Prob(JB):                         0.7
66
Kurtosis:                        1.401   Cond. No.                          1
3.3
==============================================================================
==

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

```
C:\Users\user\anaconda3\lib\site-packages\statsmodels\tsa\tsatools.py:142: F
utureWarning: In a future version of pandas all arguments of concat except f
or the argument 'objs' will be keyword-only
  x = pd.concat(x[::order], 1)
C:\Users\user\anaconda3\lib\site-packages\statsmodels\stats\stattools.py:74:
ValueWarning: omni_normtest is not valid with less than 8 observations; 5 sa
mples were given.
  warn("omni_normtest is not valid with less than 8 observations; %i "
```

In [ ]: