

oooo

PRES

# MULTINOMIAL LOGISTIC REGRESSION

- Hajar Hanifah

oooo

# TABLE OF CONTENTS

- Multinomial Logistic Regression
- Tujuan Penelitian
- Dataset
- Data Understanding



# **MULTINOMIAL LOGISTIC REGRESSION**

Regresi logistik multinomial adalah perpanjangan sederhana dari regresi logistik biner yang memungkinkan lebih dari dua kategori variabel dependen atau hasil. Seperti regresi logistik biner, regresi logistik multinomial menggunakan estimasi kemungkinan maksimum untuk mengevaluasi probabilitas keanggotaan kategoris.



## **TUJUAN PENELITIAN**

Tujuan penelitian ini adalah melakukan Multinomial Logistic Regression, mencari NPL Score terhadap data seluruh emiten saham di sektor aneka industri untuk sub sektor Mesin dan Komponen Industri Q4 2021.

# **DATASET**

## **Data**

Data yang diambil merupakan data statictical financial ratio Q4 2021 untuk emiten di sektor Industri

## **Sumber Dataset**

Data di dapatkan dari website IDX - Laporan Statistic Financial Data Ratio.  
<https://www.idx.co.id/data-pasar/laporan-statistik/digital-statistic-beta/financial-data-ratio?>

# ooo DATA UNDERSTANDING

## Rasio yang digunakan :

- X1. DER = Total Debt / Total Equity
- X2. Return on Asset = Net Income / total Asset
- X3. Sales Growth = (Sales 2021 - sales 2020)/ sales 2020
- X4. Others
- X5 : others
- Y. Performance / Category = Using Clustering result yang hasilnya didapatkan dari Clustering Sektor Industri - Aneka Industri Q4 2021

ooo

# ooo DATA UNDERSTANDING

Emiten	Pihak yang melakukan penawaran umum, yaitu penawaran efek yang dilakukan oleh emiten untuk menjual efek kepada masyarakat berdasarkan tata cara yang diatur dalam peraturan undang-undang yang berlaku.
Sales Growth	Kenaikan jumlah penjualan dari tahun ke tahun atau dari waktu ke waktu.
Return of Asset (ROA)	Indikator untuk menunjukkan seberapa untuk sebuah perusahaan dibandingkan dengan total asetnya.
Debt to Equity Ratio (DER)	Rasio hutang terhadap ekuitas atau rasio keuangan yang membandingkan jumlah hutang dengan ekuitas

ooo

# ooo DATA UNDERSTANDING

Category	Hasil Clustering data Emiten sektor Industri – Aneka Industri , Mesin dan Komponen Industri
ROE (Return Of Equity)	Return on equity atau ROE adalah indikator kinerja perusahaan dengan membandingkan laba bersih dan total modal
NPM (Net Profit Margin)	ingkat keuntungan suatu perusahaan dari penjualan atau pendapatan yang diperoleh.

ooo

o o o o **RUMUS**

## **Sales Growth**

(penjualan periode 2021 - penjualan periode 2020)

---

penjualan periode 2020

**02 ROA**

Laba setelah pajak

---

Total Asset

**03 DER**

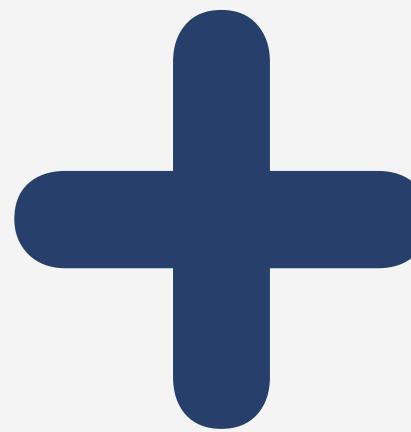
Total Uang

---

Total Ekuitas

o o o o

# DATA PREPARATION

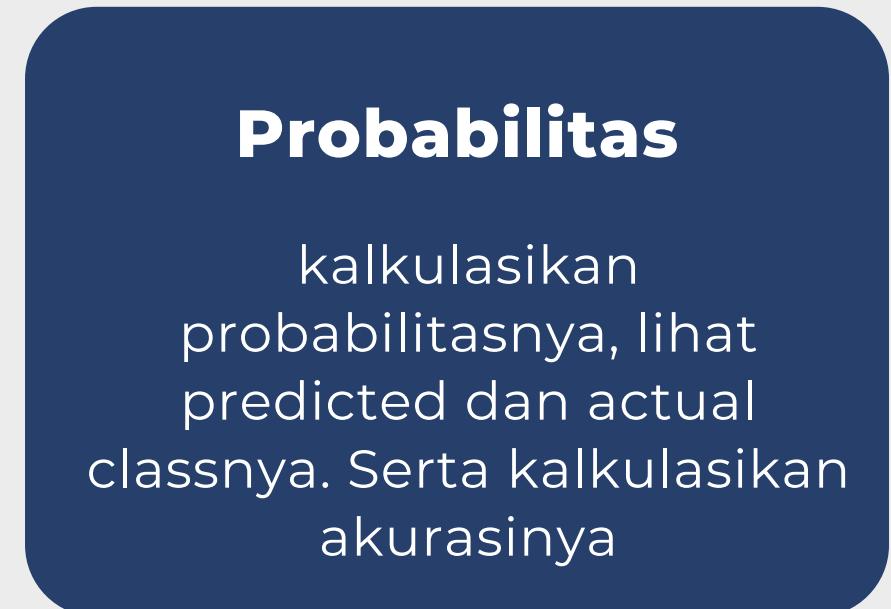
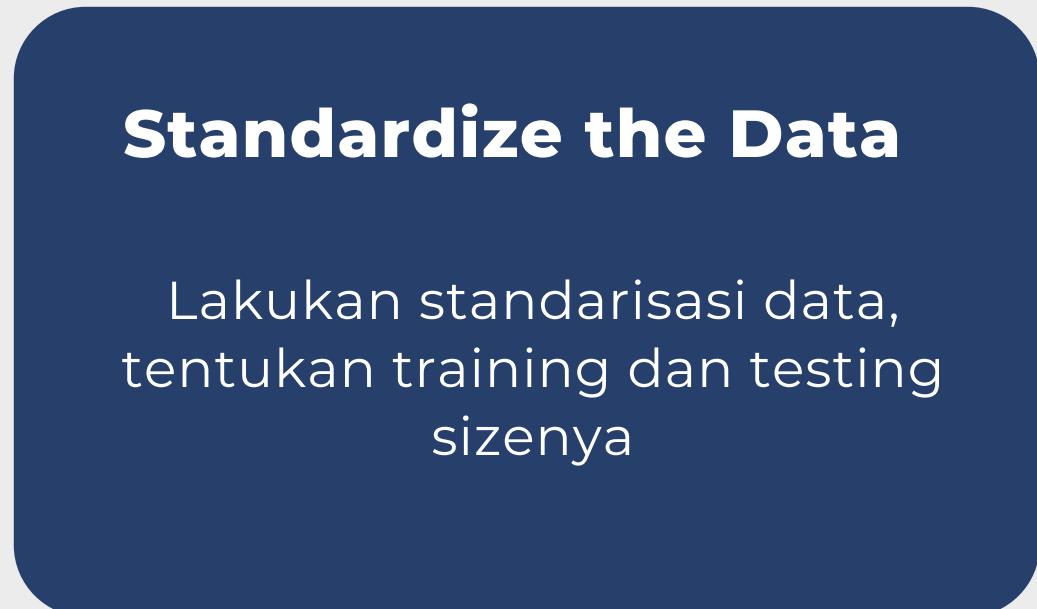
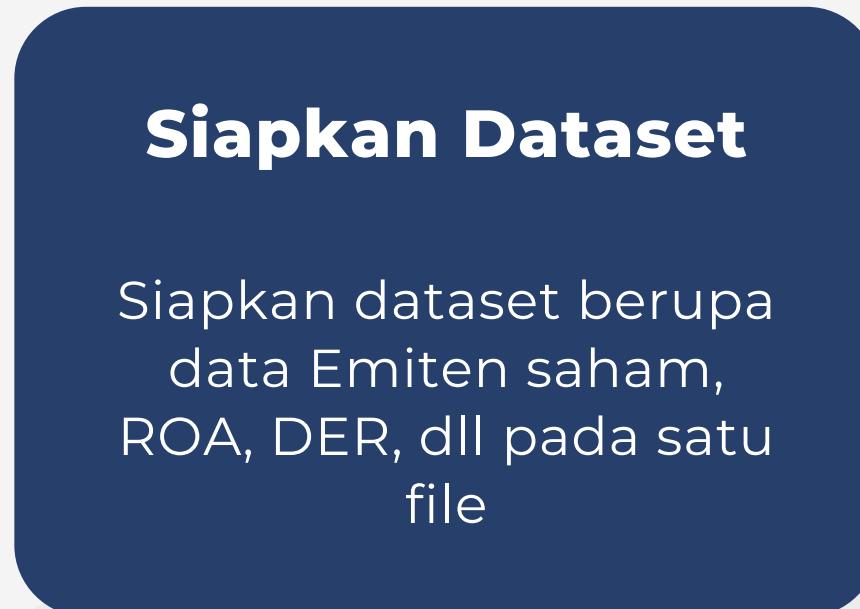


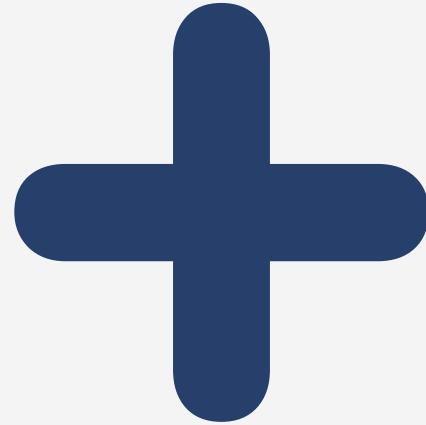
## Multinomial logistic regression with scikit-learn

- assign predictors dan kriteria untuk setiap objek
  - split x dan Y
  - Y = categori
- buat variabel training dan testing

## Model Validation

cari tahu akurasi dan error ratenya





1. Import Libraries
2. Exploratory Data Analysis
  - Data Understanding
  - Understanding the Table
  - Load Data
  - Modeling
3. Multinomial logistic regression with scikit-learn
  - Data Preparation
  - Fit the Model
  - Standardize the Data
  - Model Validation
  - Probabilitas
4. Conclusion berupa data Emiten saham, ROA, DER, dll pada satu file



# DEPLOYMENT

Bahasa pemrograman yang digunakan adalah Phyton, adapun code dapat di akses di :

<https://github.com/Hajarhanifah/big-data/blob/main/multinomial-logistic-regression/multinomial-logistic-regression-2-aneka-industri.ipynb>

# DATA

No	Industri	Company	Emiten	Category	DER	ROA	Sales Growth	ROE	NPM
1	Industrial Machinery & Components	Asahimas Flat Glass Tbk	AMFG	1	1.32	0.06	0.33	0.13	0.12
2	Industrial Machinery & Components	PT Ateliers Mecaniques D Indonesia Tbk.	AMIN	1	1.14	-0.05	0.41	-0.12	-0.13
3	Industrial Machinery & Components	PT Arita Prima Indonesia Tbk.	APII	1	0.49	0.05	0.03	0.07	0.13
4	Industrial Machinery & Components	PT Arkha Jayanti Persada Tbk.	ARKA	1	3.97	-0.02	0.22	-0.08	-0.16
5	Industrial Machinery & Components	Arwana Citramulia Tbk	ARNA	4	0.51	0.21	0.17	0.32	0.25
6	Industrial Machinery & Components	Cahayaputra Asa Keramik Tbk	CAKK	1	0.88	0.03	0.30	0.06	0.07
7	Industrial Machinery & Components	Communication Cable Systems Indonesia Tbk	CCSI	4	0.31	0.11	0.79	0.14	0.17
8	Industrial Machinery & Components	Citatah Tbk	CTTH	1	2.28	-0.05	-0.17	-0.16	-0.54
9	Industrial Machinery & Components	Hexindo Adiperkasa Tbk	HEXA	4	1.38	0.12	0.68	0.29	0.21
10	Industrial Machinery & Components	Sumi Indo Kabel Tbk	IKBI	1	0.67	0.01	0.67	0.01	0.01
11	Industrial Machinery & Components	Impack Pratama Industri Tbk	IMPC	1	0.72	0.08	0.28	0.13	0.13
12	Industrial Machinery & Components	Intraco Penta Tbk	INTA	3	-2.86	-0.32	-0.22	-	-1.99
13	Industrial Machinery & Components	Jembo Cable Company Tbk	JECC	1	1.58	-0.04	0.16	-0.10	-0.05
14	Industrial Machinery & Components	KMI Wire & Cable Tbk	KBLI	1	0.16	0.02	-0.17	0.02	0.05
15	Industrial Machinery & Components	Kabelindo Murni Tbk	KBLM	1	0.38	-	0.36	-0.01	-0.01
16	Industrial Machinery & Components	Keramika Indonesia Assosiasi Tbk	KIAS	1	0.18	-0.01	0.41	-0.02	-0.03
17	Industrial Machinery & Components	Kobexindo Tractors Tbk	KOBX	2	2.79	0.06	1.44	0.24	0.07
18	Industrial Machinery & Components	Kokoh Inti Arebama Tbk	KOIN	2	8.53	-0.02	0.99	-0.21	-0.01
19	Industrial Machinery & Components	Steadfast Marine Tbk	KPAL	1	3.15	-0.02	0.00	-0.08	-0.50
20	Industrial Machinery & Components	PT Grand Kartech Tbk	KRAH	0	16.33	-0.05	0.00	-0.93	-0.18
21	Industrial Machinery & Components	PT Mark Dynamics Indonesia Tbk.	MARK	4	0.67	0.31	1.42	0.52	0.40
22	Industrial Machinery & Components	Mulia Industrindo Tbk	MLIA	1	0.91	0.08	0.16	0.15	0.15
23	Industrial Machinery & Components	Supreme Cable Manufacturing & Commerce Tbk	SCCO	1	0.08	0.04	0.16	0.05	0.04
24	Industrial Machinery & Components	Singaraja Putra Tbk	SINI	1	4.09	0.02	0.29	0.10	0.01
25	Industrial Machinery & Components	Superkrane Mitra Utama Tbk	SKRN	1	1.68	0.01	-0.28	0.01	0.03
26	Industrial Machinery & Components	Surya Pertiwi Tbk	SPTO	1	0.54	0.06	0.17	0.08	0.11
27	Industrial Machinery & Components	Surya Toto Indonesia Tbk	TOTO	1	0.65	0.02	0.14	0.04	0.06
28	Industrial Machinery & Components	United Tractors Tbk	UNTR	1	0.59	0.09	0.24	0.15	0.18
29	Industrial Machinery & Components	Voksel Electric Tbk	VOKS	1	1.91	-0.06	-0.15	-0.18	-0.14

# MULTINOMIAL LOGISTIC REGRESSION WITH SCIKIT-LEARN

Tetapkan prediktor dan kriteria untuk setiap objek dan membagi datensatz menjadi bagian pelatihan dan pengujian

X : DER, ROA, SALES GROWTH,  
NPM  
Y : CATEGORY

```
[800]: x = df.drop(['Category', 'No', 'Industri', 'Company', 'Emiten'], axis=1)
x.head()

[80...]
```

	DER	ROA	Sales Growth	ROE	NPM
0	1.320000	0.060000	0.330000	0.130000	0.120000
1	1.140000	-0.050000	0.410000	-0.120000	-0.130000
2	0.490000	0.050000	0.030000	0.070000	0.130000
3	3.970000	-0.020000	0.220000	-0.080000	-0.160000
4	0.510000	0.210000	0.170000	0.320000	0.250000

```
▶ y = df['Category']
y.head()
```

```
[80...]
```

	Category
0	1
1	1
2	1
3	1
4	4

Name: Category, dtype: int64

+ Code + Markdown

# STANDARDIZE THE DATA

```
from sklearn.preprocessing import StandardScaler
from sklearn import preprocessing
x = preprocessing.StandardScaler().fit(x).transform(x)
```

```
] : display(df)
```

No	Industri	Company	Emiten	DER	ROA	Sales Growth	ROE	NPM	Category
0	1 Industrial Machinery & Components	Asahimas Flat Glass Tbk	AMFG	1.320000	0.060000	0.330000	0.130000	0.120000	1
1	2 Industrial Machinery & Components	PT Ateliers Mecaniques D Indonesia Tbk.	AMIN	1.140000	-0.050000	0.410000	-0.120000	-0.130000	1
2	3 Industrial Machinery & Components	PT Arita Prima Indonesia Tbk.	APII	0.490000	0.050000	0.030000	0.070000	0.130000	1
3	4 Industrial Machinery & Components	PT Arkha Jayanti Persada Tbk.	ARKA	3.970000	-0.020000	0.220000	-0.080000	-0.160000	1
4	5 Industrial Machinery & Components	Arwana Citramulia Tbk	ARNA	0.510000	0.210000	0.170000	0.320000	0.250000	4
5	6 Industrial Machinery & Components	Cahayaputra Asa Keramik Tbk	CAKK	0.880000	0.030000	0.300000	0.060000	0.070000	1

# STANDARDIZE THE DATA

[804]:

```
trainX, testX, trainY, testY = train_test_split(x, y, test_size = 0.3)
```

Diketahui :

- training : 70%
- testing 30%

Diketahui :

- training : 70%
- testing 30%

# MODEL AND VALIDATION

[90]:

```
log_reg = LogisticRegression(solver='newton-cg', multi_class='multinomial')
log_reg.fit(trainX, trainY)
y_pred = log_reg.predict(testX)
```

[91]:

```
print('Accuracy: {:.2f}'.format(accuracy_score(testY, y_pred)))
print('Error rate: {:.2f}'.format(1 - accuracy_score(testY, y_pred)))
```

Accuracy: 0.56  
Error rate: 0.44

+ Code

+ Markdown

# PROBABILITAS



```
probability = log_reg.predict_proba(testX)  
probability
```

```
[80... array([[1.78591952e-03, 5.56367161e-02, 1.07741149e-01, 3.01397561e-03,  
8.31822239e-01],  
[7.29347114e-03, 9.74290417e-01, 5.73791555e-03, 9.68931860e-03,  
2.98887734e-03],  
[4.08748707e-03, 9.08581603e-01, 1.52094179e-02, 6.92099996e-03,  
6.52004918e-02],  
[1.63135766e-02, 9.19471226e-01, 1.94887736e-02, 3.28740932e-02,  
1.18523304e-02],  
[2.29303720e-03, 8.78426656e-01, 9.21401430e-03, 3.40043371e-03,  
1.06665859e-01],  
[2.83844921e-03, 9.85103551e-01, 2.61857503e-03, 3.68271414e-03,  
5.75671110e-03],  
[2.77241445e-03, 9.43809243e-01, 7.60033533e-03, 5.60925672e-03,  
4.02087504e-02],  
[9.60024351e-04, 6.88843283e-01, 4.67203882e-03, 1.21248352e-03,  
3.04312170e-01],  
[7.53276628e-03, 9.53060093e-01, 1.41732040e-02, 1.11223252e-02,  
1.41116114e-02]])
```

+ Code

+ Markdown

- Setiap kolom merepresentasikan class
- class dengan probabilitas paling tinggi = output dari predicted class

# PROBABILITAS

```
df1 = pd.DataFrame(log_reg.predict_proba(testX), columns=log_reg.classes_)
df1.head()
```

	0	1	2	3	4
0	0.001786	0.055637	0.107741	0.003014	0.831822
1	0.007293	0.974290	0.005738	0.009689	0.002989
2	0.004087	0.908582	0.015209	0.006921	0.065200
3	0.016314	0.919471	0.019489	0.032874	0.011852
4	0.002293	0.878427	0.009214	0.003400	0.106666

```
display(df1)
```

	0	1	2	3	4
0	0.001786	0.055637	0.107741	0.003014	0.831822
1	0.007293	0.974290	0.005738	0.009689	0.002989
2	0.004087	0.908582	0.015209	0.006921	0.065200
3	0.016314	0.919471	0.019489	0.032874	0.011852
4	0.002293	0.878427	0.009214	0.003400	0.106666
5	0.002838	0.985104	0.002619	0.003683	0.005757
6	0.002772	0.943809	0.007600	0.005609	0.040209
7	0.000960	0.688843	0.004672	0.001212	0.304312
8	0.007533	0.953060	0.014173	0.011122	0.014112

UBAH OUTPUT YANG DITUNJUKKAN DI ATAS KE DALAM BENTUK  
DAN FORMAT YANG DAPAT DIBACA.



# SUM PROBABILITAS



```
df1[ 'sum' ] = df1.sum(axis=1)  
df1.head()
```

	0	1	2	3	4	sum
0	0.001786	0.055637	0.107741	0.003014	0.831822	1.000000
1	0.007293	0.974290	0.005738	0.009689	0.002989	1.000000
2	0.004087	0.908582	0.015209	0.006921	0.065200	1.000000
3	0.016314	0.919471	0.019489	0.032874	0.011852	1.000000
4	0.002293	0.878427	0.009214	0.003400	0.106666	1.000000

# PREDICTED CLASS



```
df1['predicted_class'] = y_pred  
df1.head()
```

	0	1	2	3	4	sum	predicted_class
0	0.001786	0.055637	0.107741	0.003014	0.831822	1.000000	4
1	0.007293	0.974290	0.005738	0.009689	0.002989	1.000000	1
2	0.004087	0.908582	0.015209	0.006921	0.065200	1.000000	1
3	0.016314	0.919471	0.019489	0.032874	0.011852	1.000000	1
4	0.002293	0.878427	0.009214	0.003400	0.106666	1.000000	1

# ACTUAL CLASS

```
df1['actual_class'] = testY.to_frame().reset_index().drop(columns='index')  
df1.head()
```

	0	1	2	3	4	sum	predicted_class	actual_class
0	0.001786	0.055637	0.107741	0.003014	0.831822	1.000000	4	2
1	0.007293	0.974290	0.005738	0.009689	0.002989	1.000000	1	1
2	0.004087	0.908582	0.015209	0.006921	0.065200	1.000000	1	1
3	0.016314	0.919471	0.019489	0.032874	0.011852	1.000000	1	1
4	0.002293	0.878427	0.009214	0.003400	0.106666	1.000000	1	1

Dari data diatas, kita dapat melihat hasil dari prediksi setiap class. Untuk memudahkan dalam membaca prediksi, lakukan konversi kelas prediksi (prediction class) dan actual class menggunakan encoder label dari scikit-learn, lanjutkan dengan nilai numerik.

DI SINI KITA MELIHAT BAHWA DUA VARIABEL (PREDICTED\_CLASS & ACTUAL\_CLASS) DIBERI KODE YANG SAMA (0 DAN 1).



```
le = preprocessing.LabelEncoder()

df1['label_pred'] = le.fit_transform(df1['predicted_class'])
df1['label_actual'] = le.fit_transform(df1['actual_class'])
df1.head()
```



	0	1	2	3	4	sum	predicted_class	actual_class	label_pred	label_actual
0	0.001786	0.055637	0.107741	0.003014	0.831822	1.000000	4	2	1	1
1	0.007293	0.974290	0.005738	0.009689	0.002989	1.000000	1	1	0	0
2	0.004087	0.908582	0.015209	0.006921	0.065200	1.000000	1	1	0	0
3	0.016314	0.919471	0.019489	0.032874	0.011852	1.000000	1	1	0	0
4	0.002293	0.878427	0.009214	0.003400	0.106666	1.000000	1	1	0	0

[+ Code](#) [+ Markdown](#)

```
targets = df1['predicted_class']
integerEncoded = le.fit_transform(targets)
integerMapping=dict(zip(targets,integerEncoded))
integerMapping
```

{4: 1, 1: 0}

```
df1['correct_prediction?'] = np.where(df1['check'] == 0, 'True', 'False')
df1 = df1.drop(['label_pred', 'label_actual', 'check'], axis=1)
df1.head()
```

	0	1	2	3	4	sum	predicted_class	actual_class	correct_prediction?
0	0.001786	0.055637	0.107741	0.003014	0.831822	1.000000	4	2	True
1	0.007293	0.974290	0.005738	0.009689	0.002989	1.000000	1	1	True
2	0.004087	0.908582	0.015209	0.006921	0.065200	1.000000	1	1	True
3	0.016314	0.919471	0.019489	0.032874	0.011852	1.000000	1	1	True
4	0.002293	0.878427	0.009214	0.003400	0.106666	1.000000	1	1	True

kalkulasikan value dari akurasi, dan probabilitas kelas yang salah prediksi

```
true_predictions = df1[(df1["correct_prediction?"] == 'True')].shape[0]
false_predictions = df1[(df1["correct_prediction?"] == 'False')].shape[0]
total = df1["correct_prediction?"].shape[0]

print('manual calculated Accuracy is:', (true_predictions / total * 100))
```

manual calculated Accuracy is: 88.88888888888889

```
wrong_pred = df1[(df1["correct_prediction?"] == 'False')]
wrong_pred
```

0	1	2	3	4	sum	predicted_class	actual_class	correct_prediction?	
7	0.000960	0.688843	0.004672	0.001212	0.304312	1.000000	1	4	False



# KESIMPULAN

- Regresi Logistik Multinomial dapat digunakan untuk memprediksi beberapa kelas
- Dengan data yang ada, dilakukan validasi dengan testing size 30% menghasilkan akurasi data 88,88%





# THANK YOU

- Hajar Hanifah

EMAIL : HAJAR.HANIFAH@GMAIL.COM  
GITHUB : GITHUB.COM/HAJARHANIFAH  
KAGGLE : HAJARHANIFAH

