DigitalSkola

LEARNING PROGRESS REVIEW WEEK 13

Data Science Bootcamp Batch 11

BY OMICRON

ANGGOTA KELOMPOK

ANUGRAH YAZID HANI

https://www.linkedin.com/in/anugrah-yazid-7253bb221/

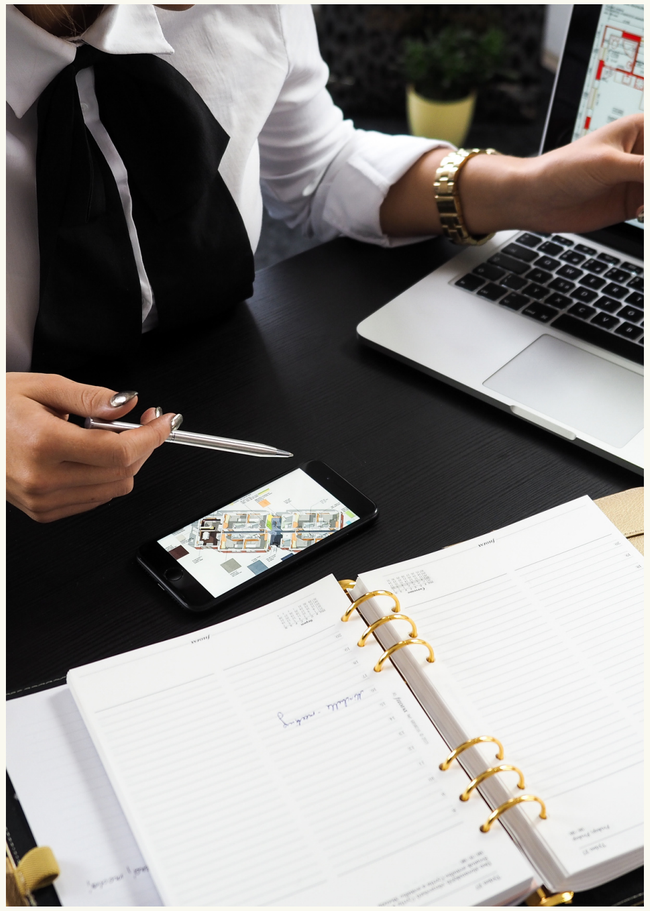
EDO M. HADAD GIBRAN

https://www.linkedin.com/in/edo-gibran-38505a142 /

MUHAMMAD FIKRI FADILA

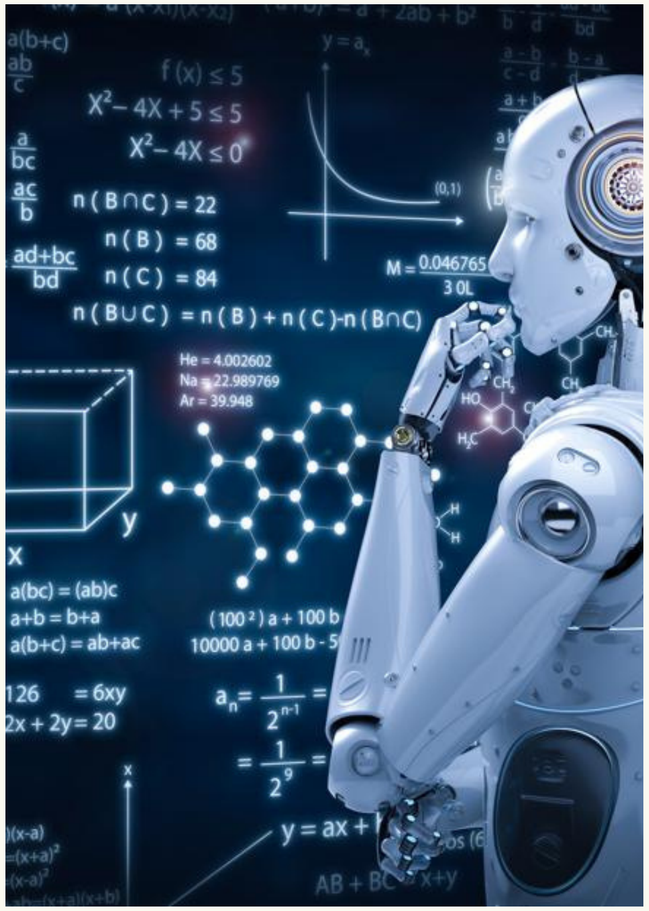
https://www.linkedin.com/in/muhammad-fikri-fadila-a551161a6 /

OUTLINE



COMMUNICATION & PRESENTATTION SKILL

EVALUATION METRICS & MODEL SELECTION



ADVANCED MACHINE LEARNING

DigitalSkola

COMMUNICATION & PRESENTATTION SKILL

Tips Presentasi

- Sederhana
- Hindari Animasi
- Pemilihan font & color
- Pahami siapa penontonnya
- Buat *outline*

DigitalSkola

Pemilihan Font & Color

Font Source

- https://fonts.google.com/
- https://www.dafont.com/
- https://fonts.adobe.com/

Color Inspiration

- https://color.adobe.com/explore
- https://coolors.co/
- https://colorhunt.co/

BIG TIPS

Ikuti ***guideline*** perusahaanmu atau perusahaan *client*

DATA STORYTELLING

Oh damn .. Why the slide is so complicated ?

Finally, final part !

Easy peasy ...

Okay, let's put everything in the slide ..

How to make our CEO understand this ...

All stories have content/data but NOT all content/data have stories

DigitalSkola

DigitalSkola

5 Cara Menyaikan Data Dengan Baik

Identify Your Audience
Is there just one group or different audiences ?

Establish an Objective
Am I recommending a decision or proving the fact ?

Decide What Data Will Help You
What analysis techniques can I use to surface the insights ?

Decide How to Tell a Story
What visualization should I use ?

Improve Next Time
Did my audience understand everything ?

Bagaimana Membuat Visualisasi Data yang Efektif ?

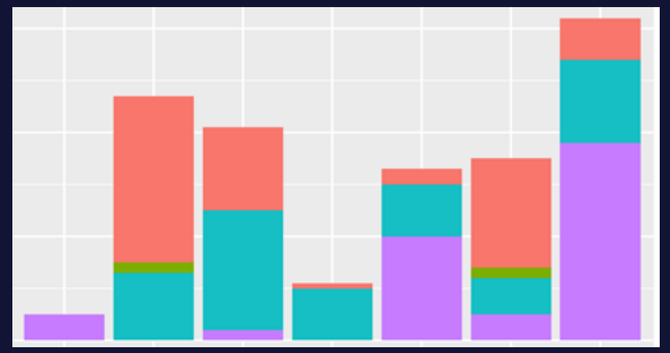
Tetapkan tujuan visualisasi data

Variabel apa yang ingin ditampilkan?

Pilih grafik yang tepat !

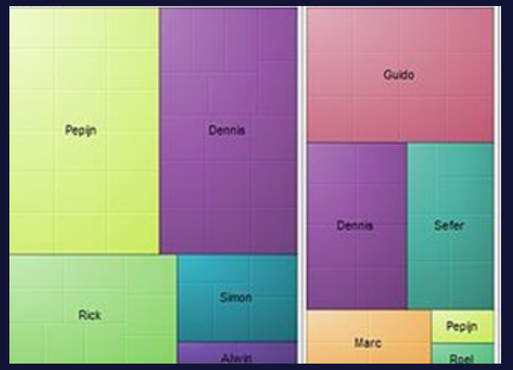
VISUALISASI PROPORSI

→ Menyampaikan perbedaan/kemiripan dalam suatu bagian data secara keseluruhan.



Stacked Bar Graph

Contoh :
Proporsi transaksi tiap item di sebuah toko.



Treemap

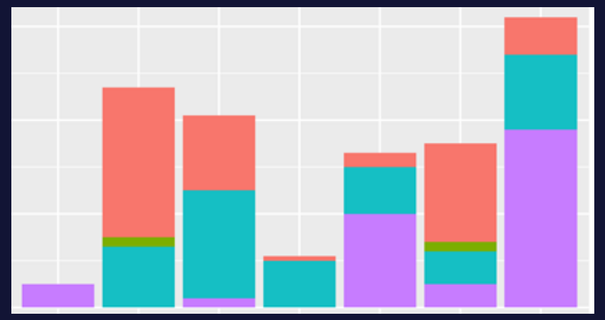
Contoh :
Proporsi transaksi per area

Pie Chart

Contoh :
Proporsi transaksi per gender customer

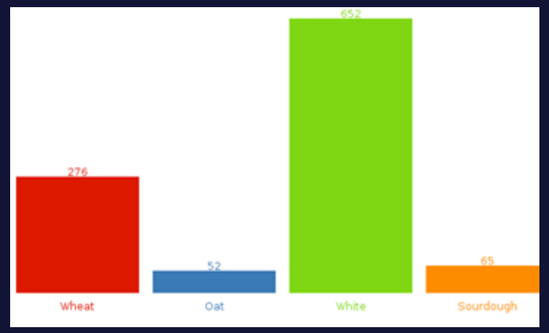
VISUALISASI PERBANDINGAN

→ Menyampaikan perbedaan/ kemiripan tiap kategori.



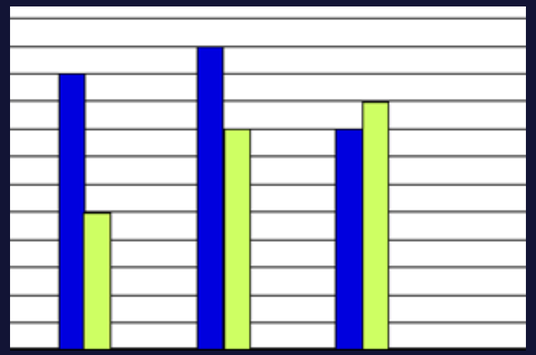
Stacked Bar Graph

Contoh :
Proporsi transaksi tiap item di sebuah toko.



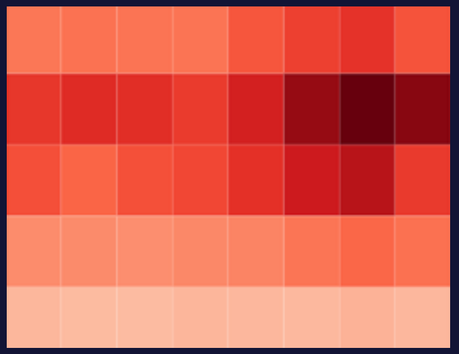
Bar Chart

Contoh :
Jumlah transaksi per item.



Multi-set Bar Chart

Contoh :
Jumlah pengguna per segmentasi.

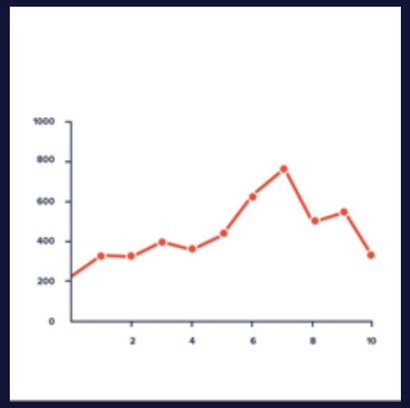


Heatmap

Contoh :
Korelasi tiap variabel dari suatu data.

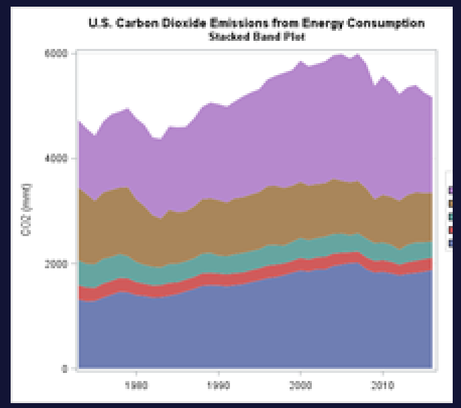
VISUALISASI DATA OVER TIME

→ Menyampaikan perubahan/ kecenderungan data dalam periode waktu tertentu.



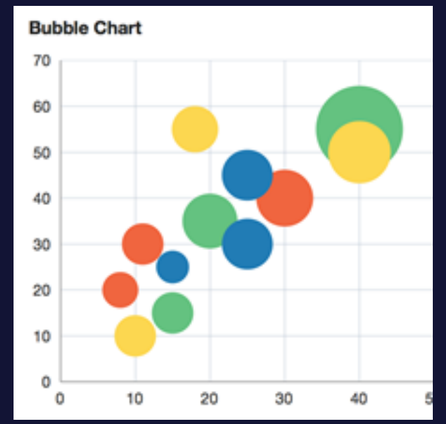
Line Graph

Contoh :
Tren transaksi per bulan.



Stacked Area Graph

Contoh :
Tren transaksi pelanggan di setiap toko per bulan.



Bubble Chart

Contoh :
Tren transaksi pelanggan pada tiap produk per bulan.

5 Tahapan Untuk Menyampaikan Cerita Dengan Menyajikan Data

BIG TIPS

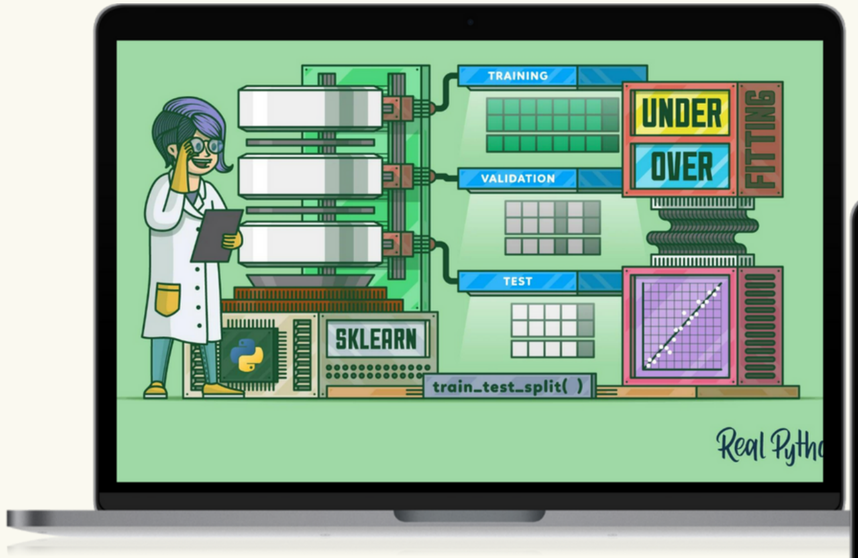
Think Stories, Not Charts!

DigitalSkola

EVALUATION METRICS & MODEL SELECTION

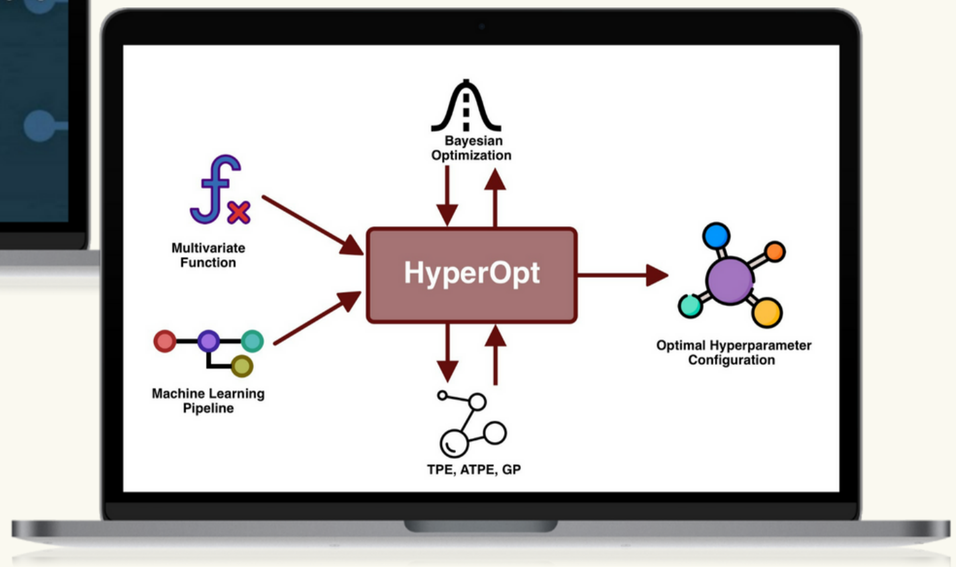
DigitalSkola

CONTENT



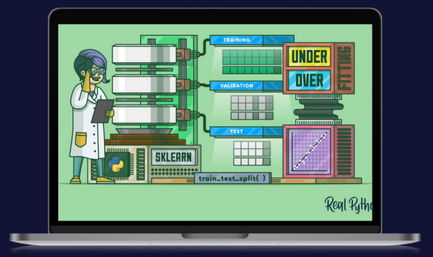
TRAIN TEST SPLIT

EVALUATION METRICS



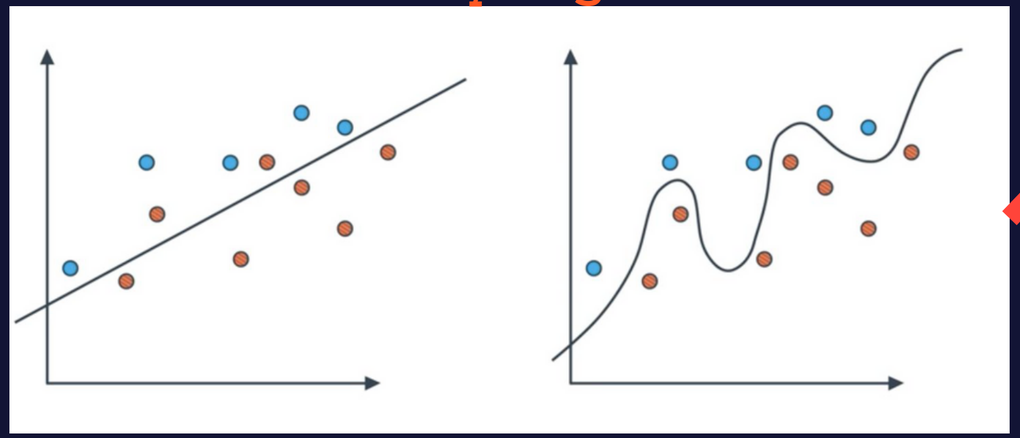
HYPERPARAMETER TUNING

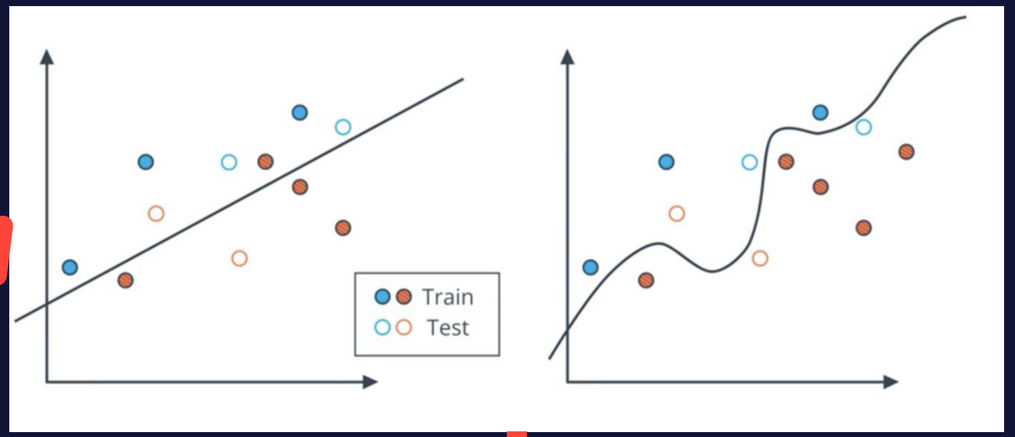
DigitalSkola

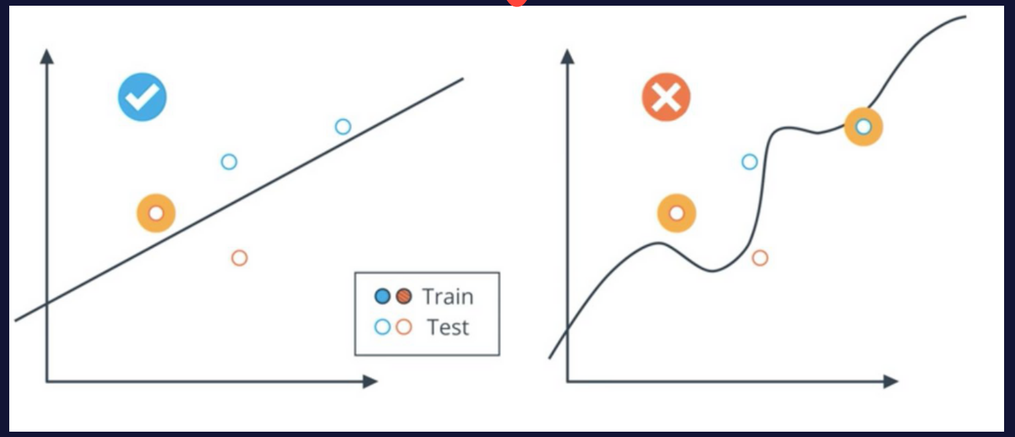


Train Test Split

Model mana yang lebih baik?







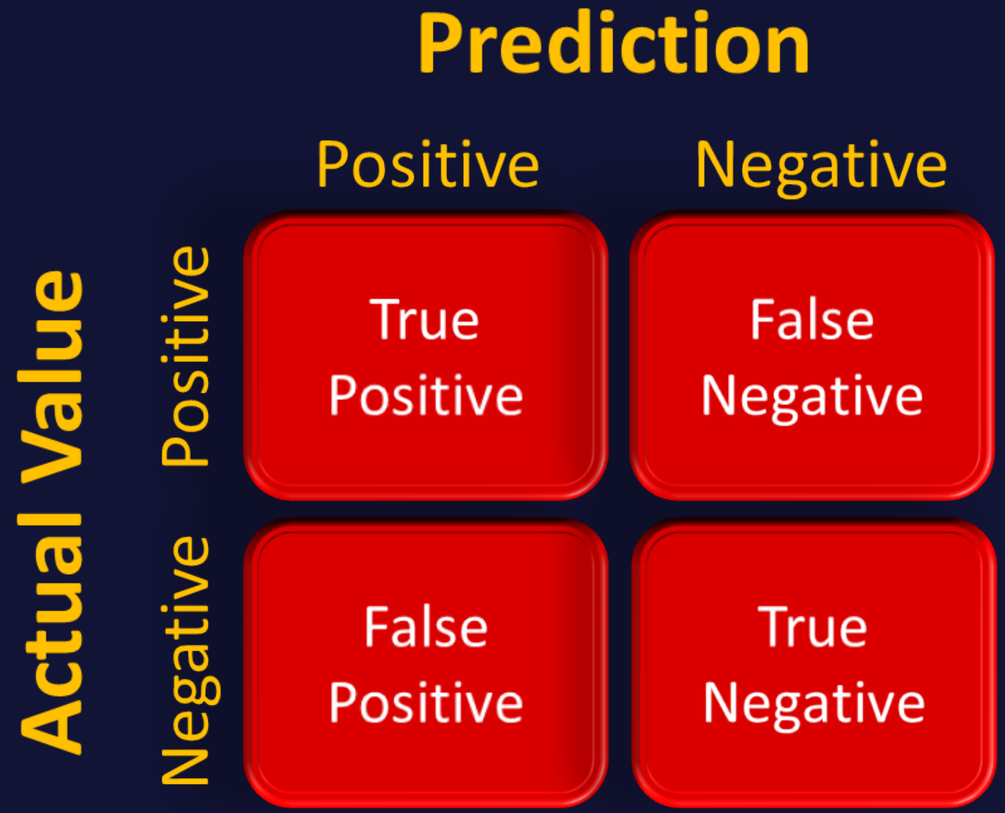

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test =
train_test_split ( X, y, test_size = 0.25 )
```

Evaluation Metrics

CONFUSSION MATRIX

adalah tabel sederhana yang menyimpan keempat nilai. terdapat 2 tipe error, yaitu:

1. Tipe error 1 (***False Positive***), saat prediktor melakukan kesalahan dimana nilai *Positive* di nilai *Negative*
2. Tipe error 2 (***False Negative***), saat prediktor melakukan kesalahan di mana nilai *Negative* dinilai *Positive*



DigitalSkola

Accuracy

dari seluruh target data, berapa banyak kita dapat mengklasifikasikan dengan tepat?

10, 000 PATIENTS

DIAGNOSIS

PATIENTS

	Diagnosed Sick	Diagnosed Healthy
Sick	1000 True Positives	200 False Negatives
Healthy	800 False Positives	8000 True Negatives

$$Accuracy = \frac{(TruePositive+TrueNegative)}{(TP+FP+FN+TN)}$$

$$= \frac{1.000 + 8.000}{10.000}$$

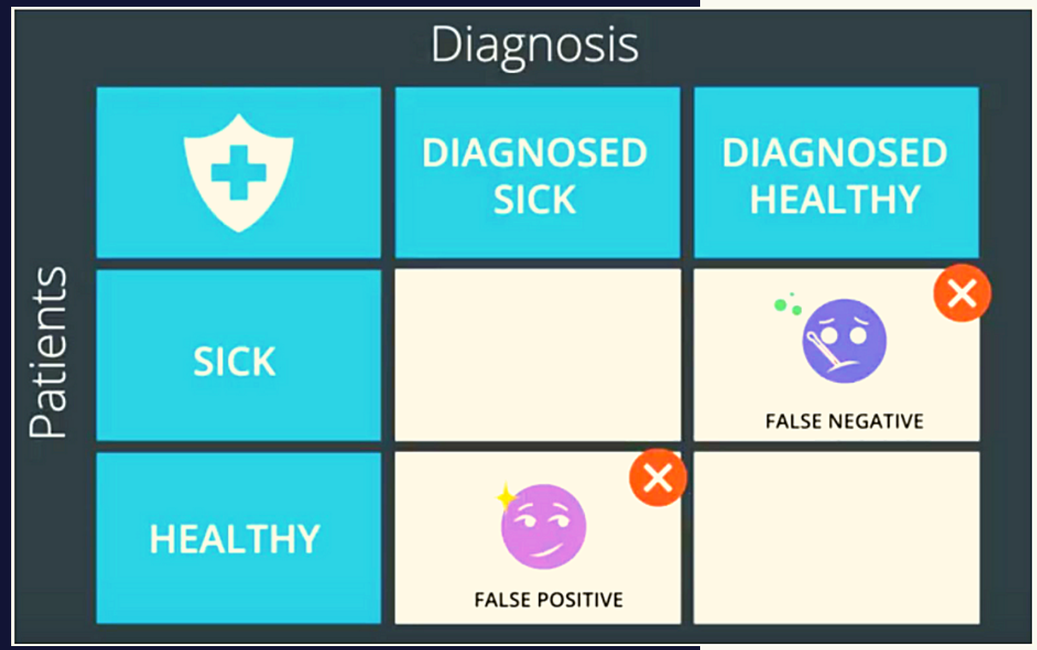
$$= \mathbf{90\%}$$

```
from sklearn.metrics import accuracy_score
accuracy_score (y_true, y_pred)
```

namun, ***Accuracy*** mungkin bukan metrik terbaik.

DigitalSkola

PRECISION, RECALL & F1 SCORE



False Positive: ketika pasien sehat, dan Anda mendiagnosisnya sakit

False Negative: ketika pasien sakit, dan Anda mendiagnosisnya sebagai sehat

$$Precision = \frac{(TruePositives)}{(TP+FP)}$$

Dari semua point yang di prediksi positif, berapa banyak dari mereka yang benar-benar positif?

$$Recall = \frac{(TruePositive)}{(TP+FN)}$$

Dari point yang berlabel positif, berapa banyak dari mereka di prediksi dengan benar sebagai positif?

F1 Score: nilai rata-rata antara *Precision* dan *Recall*

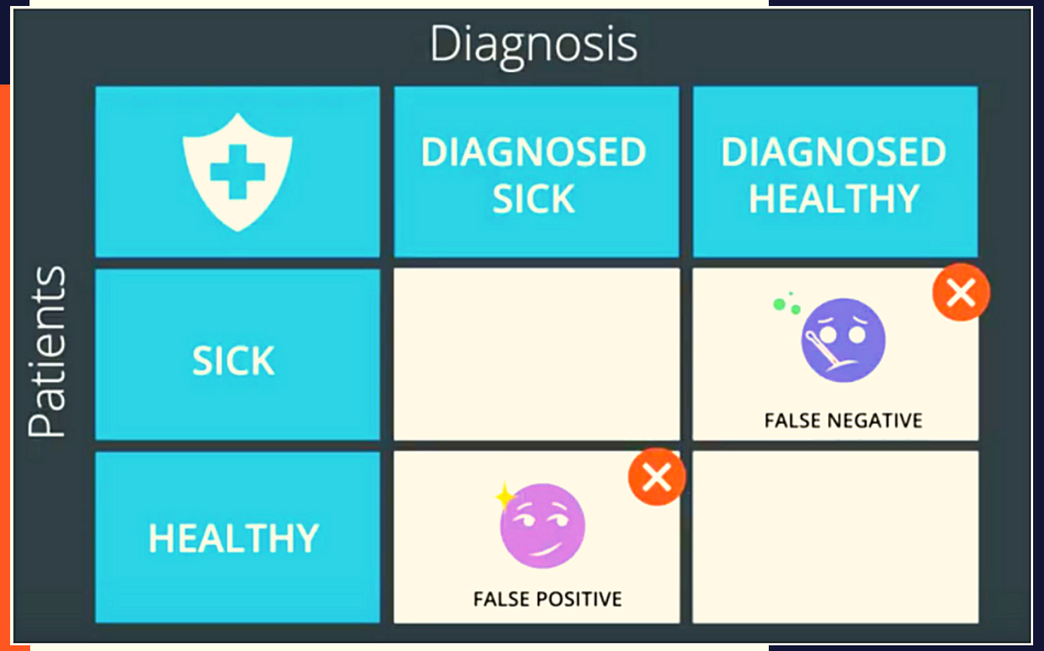
DigitalSkola

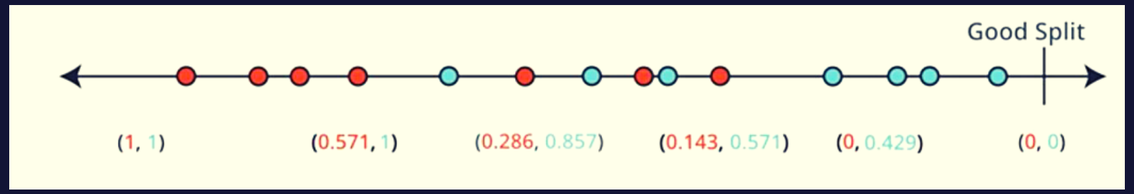
ROC CURVE

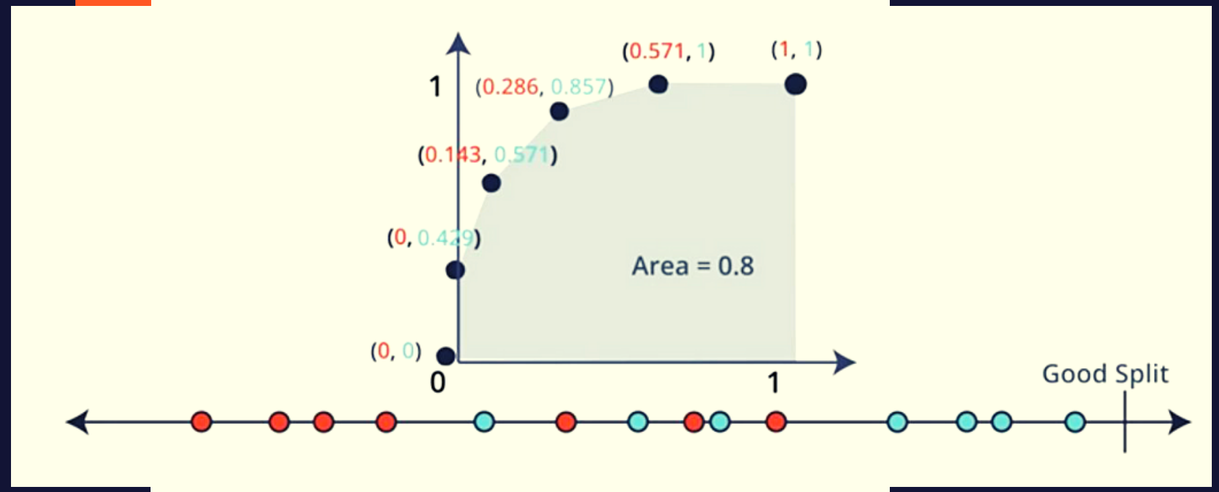
Receiver Operator Characteristic Curve

$$\text{True Positive Rate} = \frac{\text{True Positives}}{\text{All Positives}}$$ ***Sensitivity***

$$\text{False Positive Rate} = \frac{\text{False Positives}}{\text{All Negatives}}$$ ***Specificity***



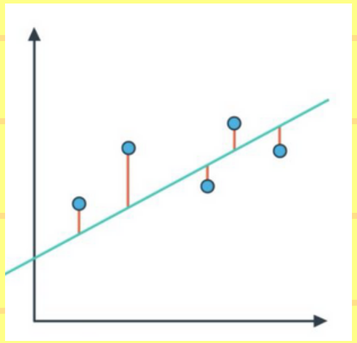




Regression Case

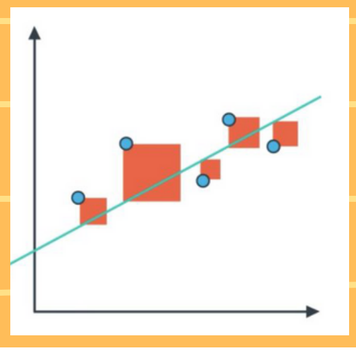
Terdapat 3 metriks yang dapat digunakan untuk menilai seberapa bagus model kita bekerja pada kasus regresi

Mean Absolute Error, yaitu menghitung dengan fungsi rata-rata kesalahan absolut



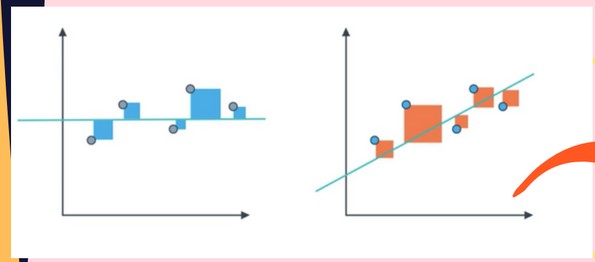
```
from sklearn.metrics import mean_absolute_error
from sklearn.linear_model import LinearRegression

classifier = LinearRegression()
classifier.fit(X,y)

guesses = classifier.predict(X)

error = mean_absolute_error(y, guesses)
```

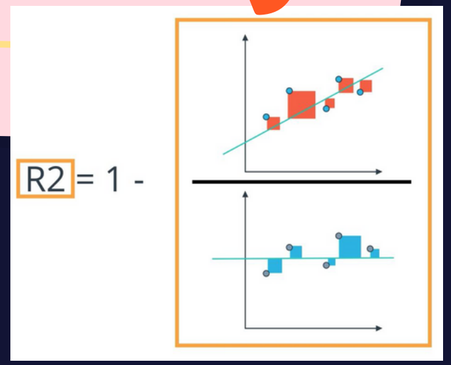
Mean Squared Error, menambahkan kuadrat jarak antara *points* dan *line*



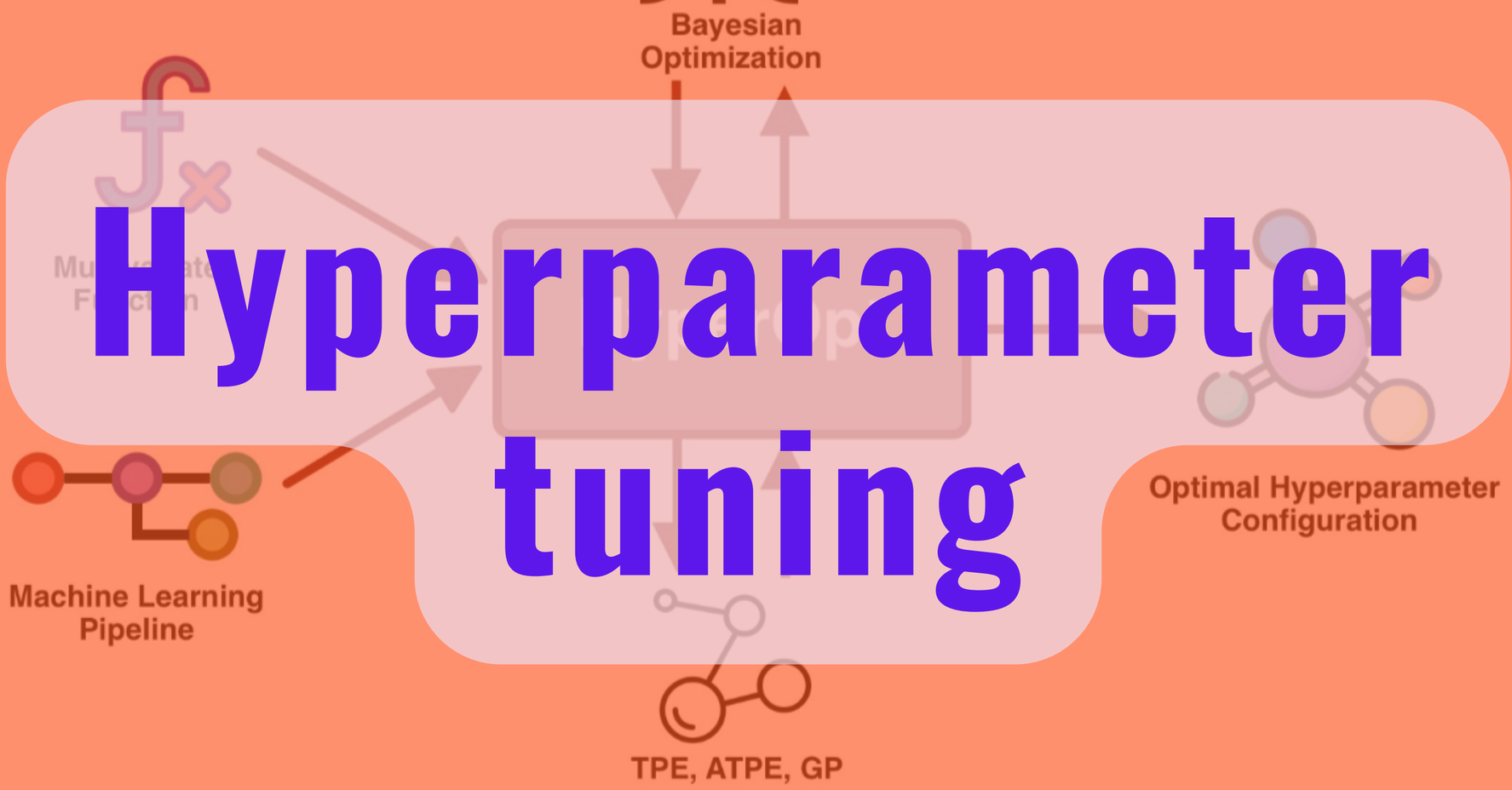
```
from sklearn.metrics import mean_squared_error
from sklearn.linear_model import LinearRegression

classifier = LinearRegression()
classifier.fit(X,y)

guesses = classifier.predict(X)

error = mean_squared_error(y, guesses)
```

R2 Score, metrik yang didasarkan pada perbandingan model kita dengan model yang paling sederhana





DigitalSkola



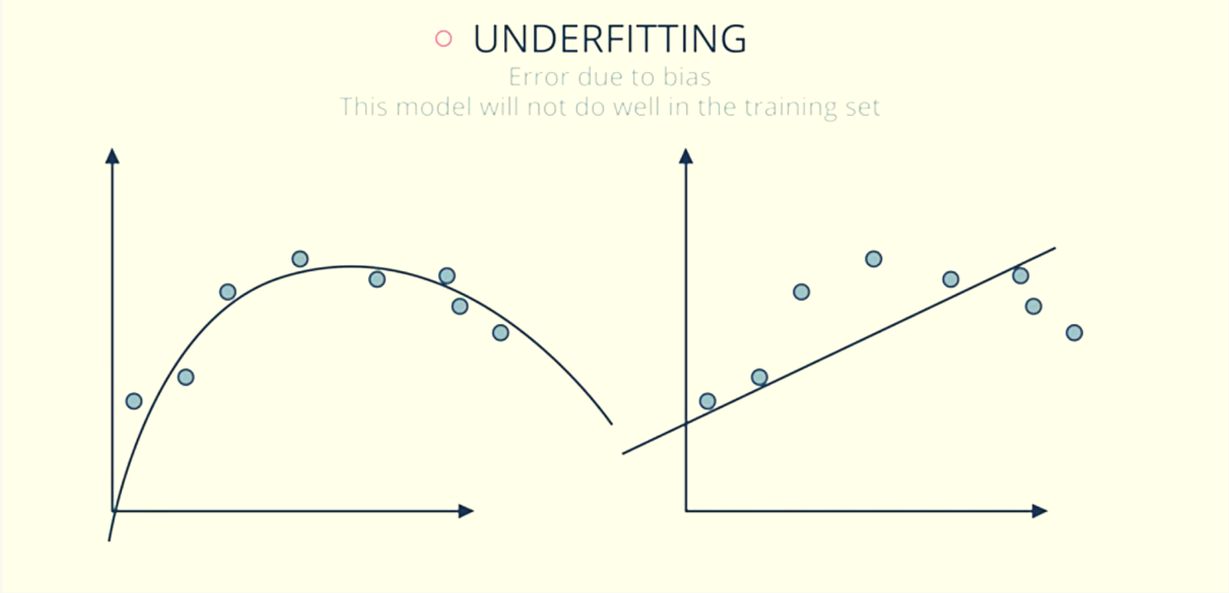
Hyperparameter tuning

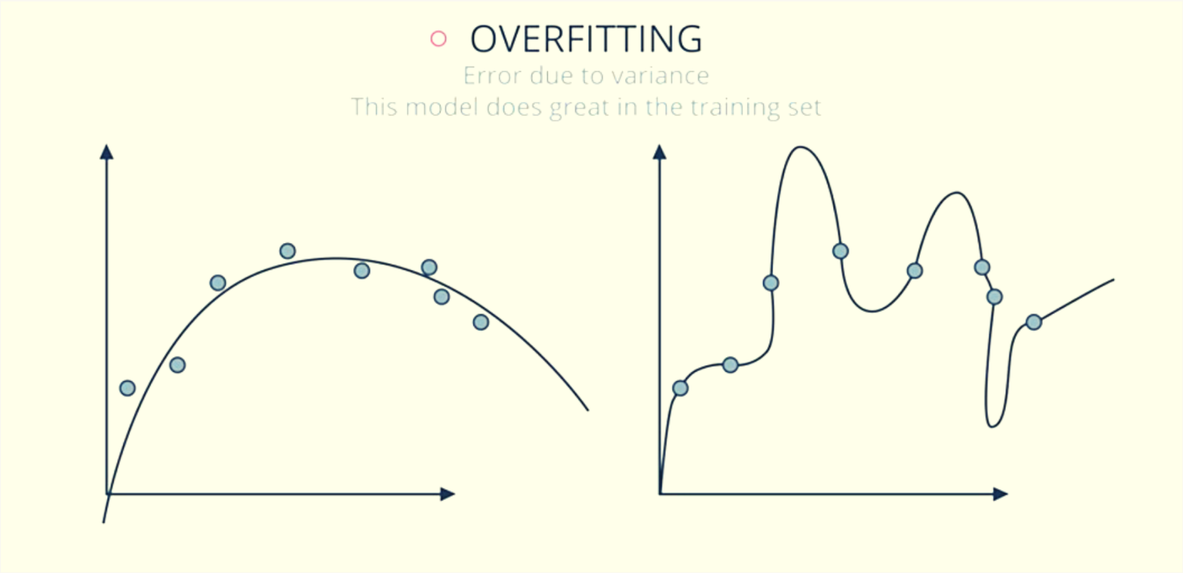
TIPE-TIPE ERROR

- **Underfitting**, saat kita terlalu menyederhanakan masalah

- **Overfitting**, saat kita terlalu mempersulit masalah

DigitalSkola



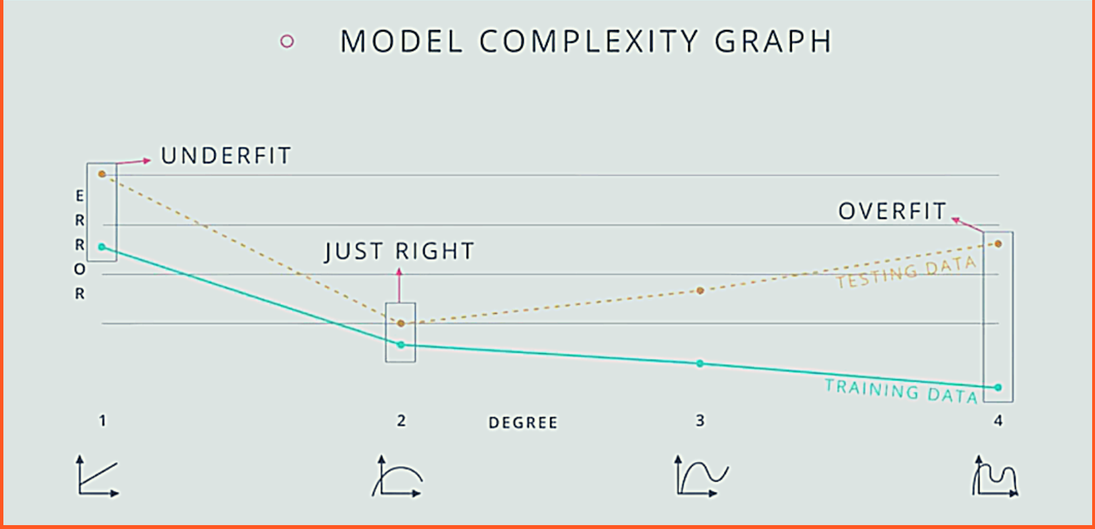


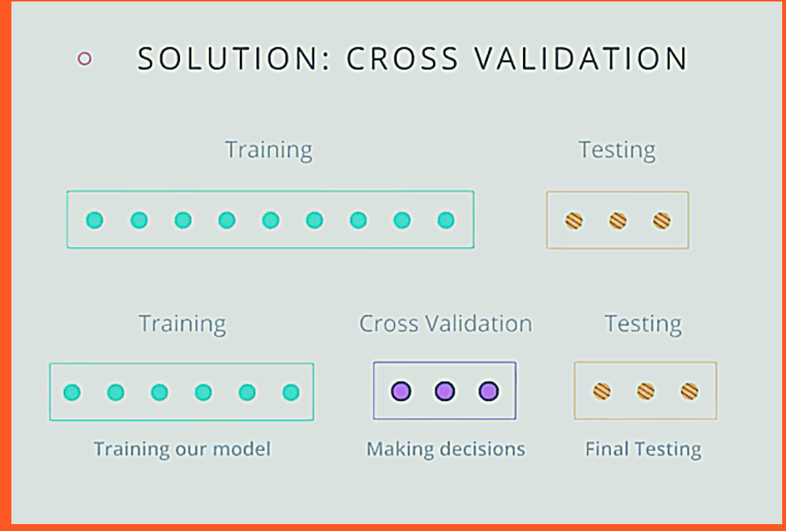
DigitalSkola

Cross Validation

Alih-alih memiliki set pelatihan dan pengujian, kita dapat menggunakan validasi silang yang akan digunakan untuk membuat keputusan

JANGAN PERNAH GUNAKAN TESTING SET UNTUK TRAINING!





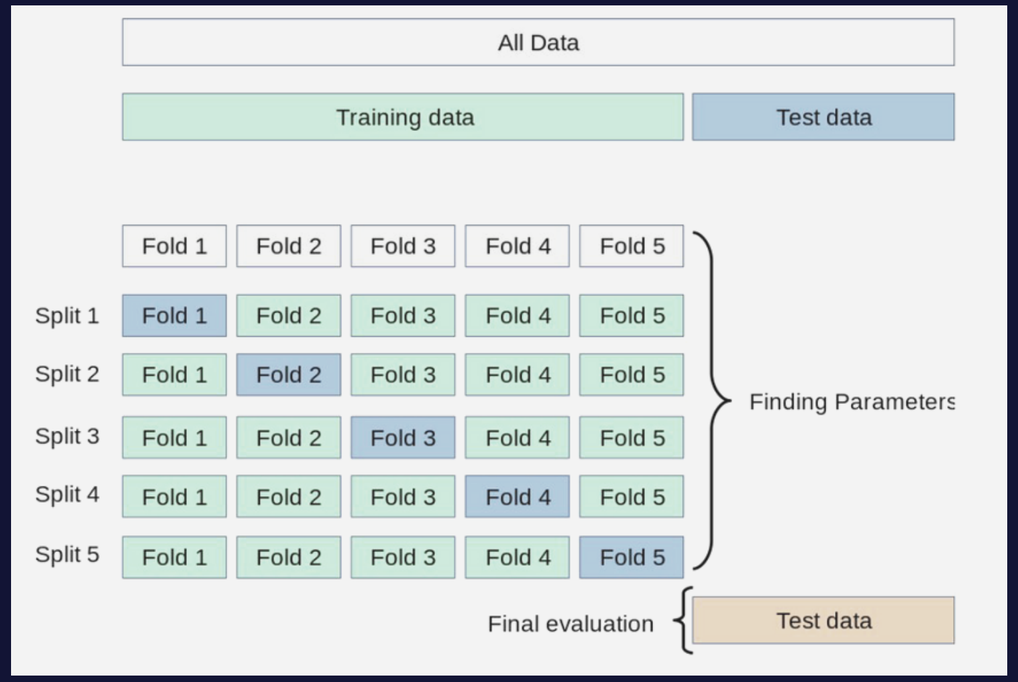
K-Fold - Cross Validation

Sekalipun kita sudah memisahkan data menjadi tiga set, sebaiknya jangan membuang data yang bisa bermanfaat untuk melatih algoritme kita

DENGAN K-FOLD - CROSS VALIDATION

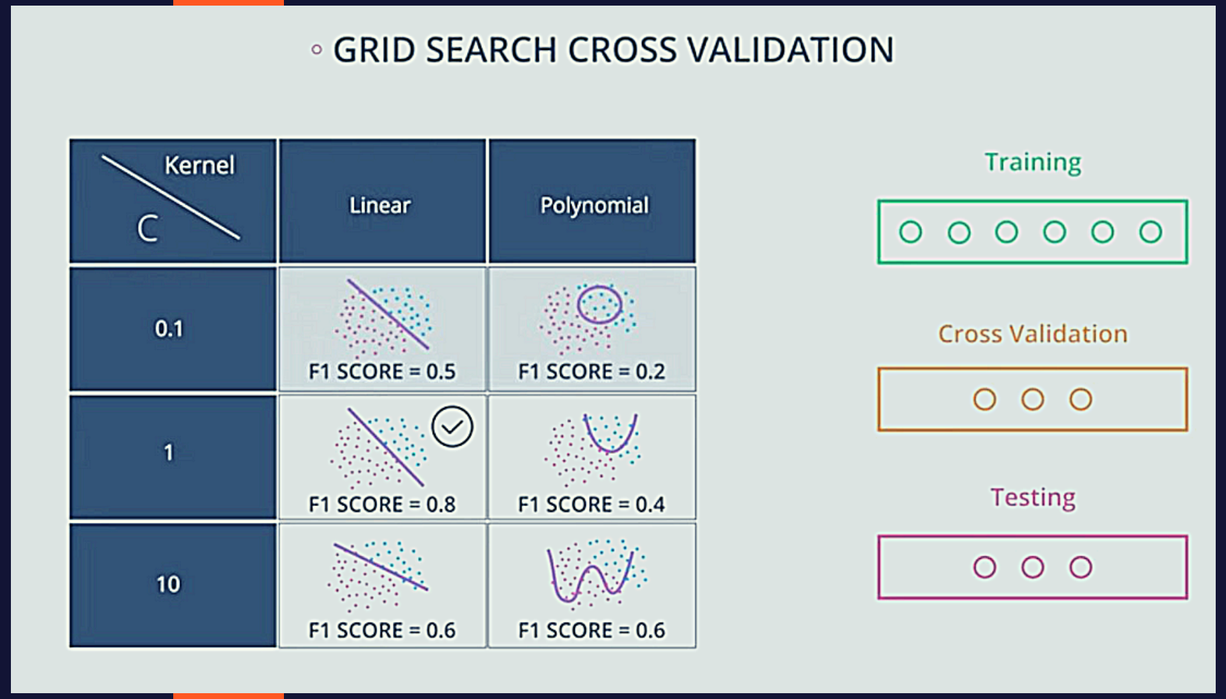
- Pecahkan data kita menjadi ***K-buckets***.
- Kemudian, rata-ratakan hasilnya untuk mendapatkan model akhir

```
from sklearn.model_selection import KFold

X = np.array([[1, 2], [3, 4], [1, 2], [3, 4]])
y = np.array([1, 2, 3, 4])

kf = KFold(n_splits=10, shuffle = True)
```



Grid Search CV

Menemukan parameter terbaik untuk menggunakannya sebagai model kita




```
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import make_scorer
from sklearn.metrics import f1_score

scorer = make_scorer(f1_score)

parameters = {'max_depth':[2,4,6,8],
'min_samples_leaf':[2,4,6,8]}

grid_obj = GridSearchCV(clf, parameters,
scoring=scorer)
grid_fit = grid_obj.fit(X, y)
```

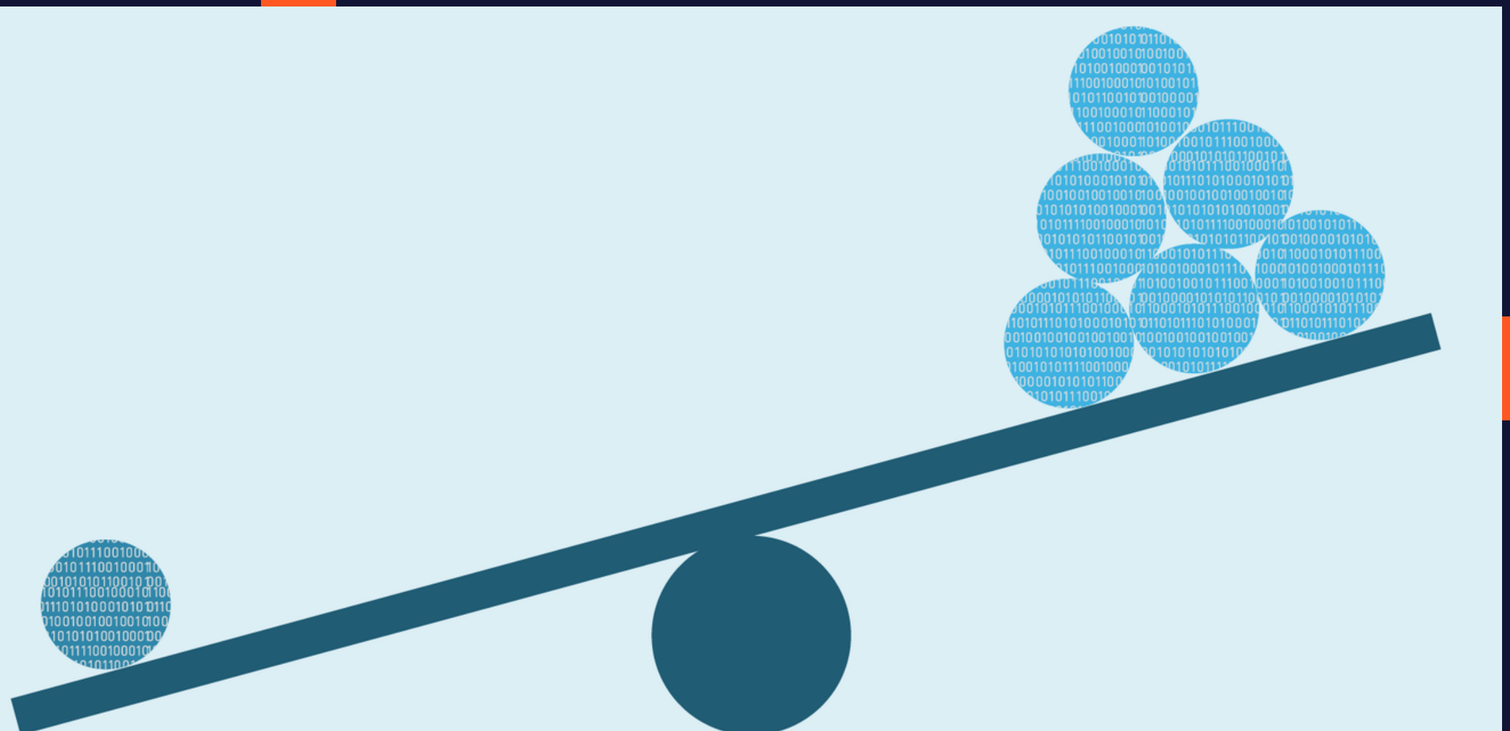
DigitalSkola

ADVANCED MACHINE LEARNING

DigitalSkola

IMBALANCED DATA

TARGET VARIABLE YANG MEMILIKI JUMLAH DATA YANG TIDAK SEIMBANG 50:50

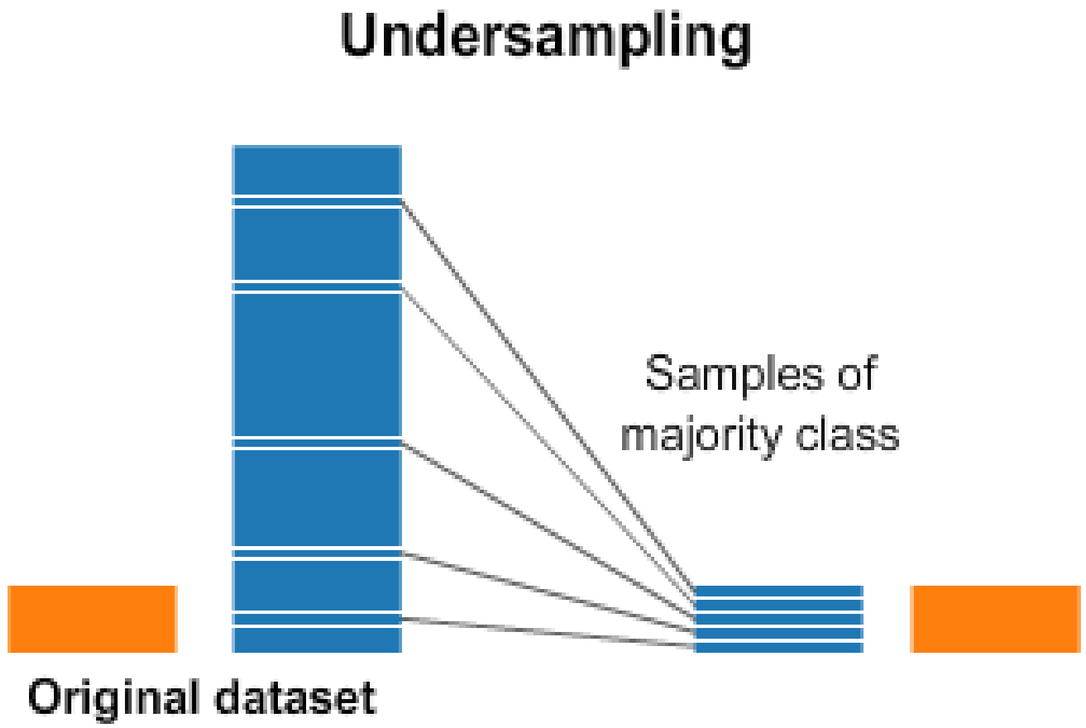


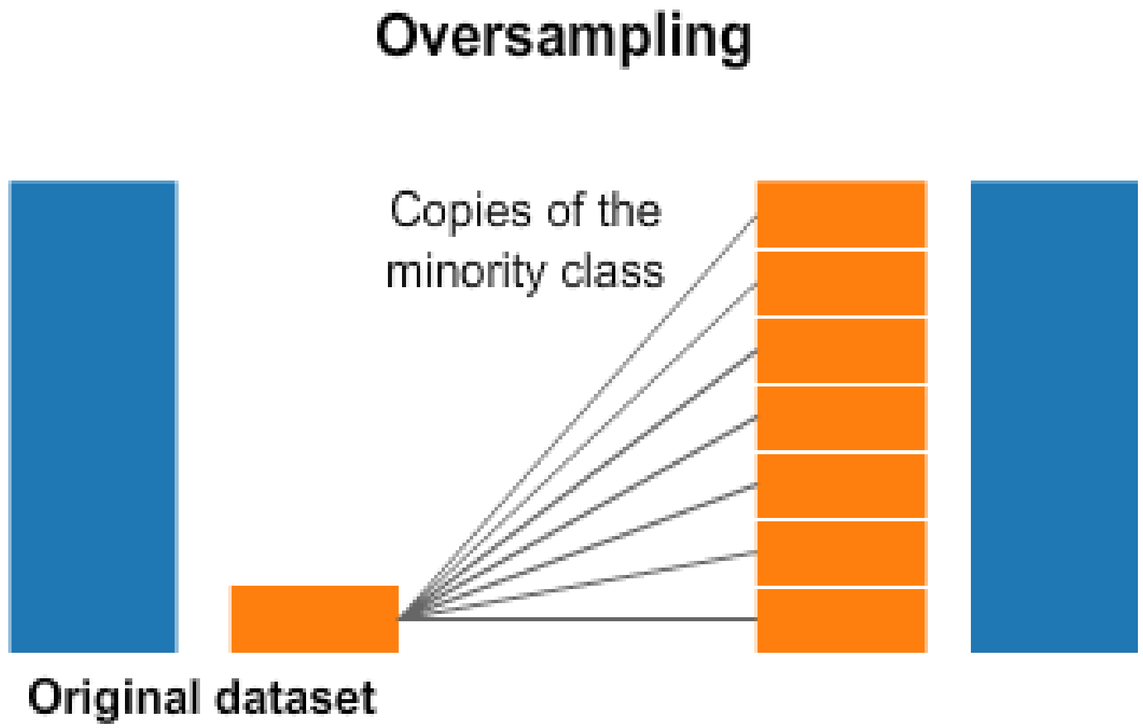
DigitalSkola

Handling Imbalance Data

TERDAPAT DUA HAL DALAM MEMBUAT DATA MENJADI BALANCE:

1. **Oversampling**: menambahkan data pada target variable yang sedikit
2. **Undersampling**: mengurangi data pada target variable yang banyak

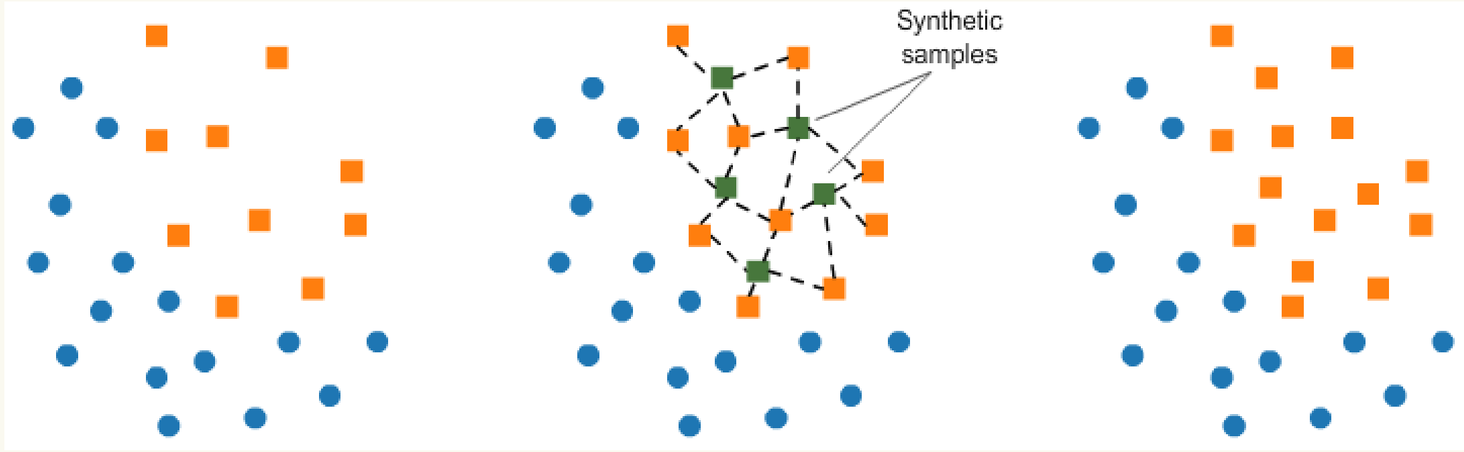




SMOTE
Synthetic Minority Oversampling TEchnique

```
from imblearn.over_sampling import SMOTE

smote = SMOTE(ratio='minority')
X_sm, y_sm = smote.fit_sample(X, y)

plot_2d_space(X_sm, y_sm, 'SMOTE over-sampling')
```

UNTUK OVERSAMPLING, LIBRARY **SMOTE** DAPAT DIGUNAKAN UNTUK PENAMBAHAN DATA SINTESIS.

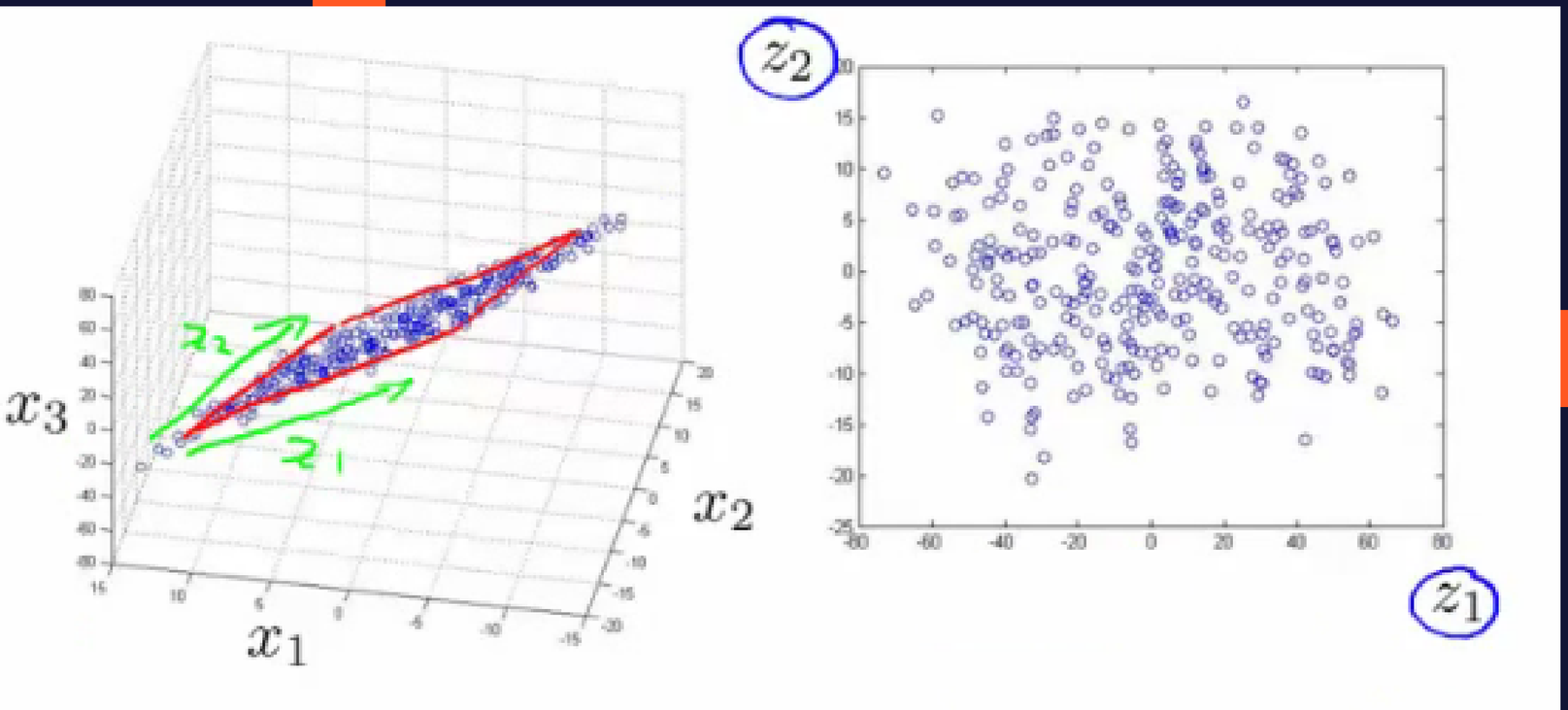


ORIGINAL DATASET	GENERATING SAMPLES	RESAMPLED DATASET

DigitalSkola

DIMENSIONALITY REDUCTION

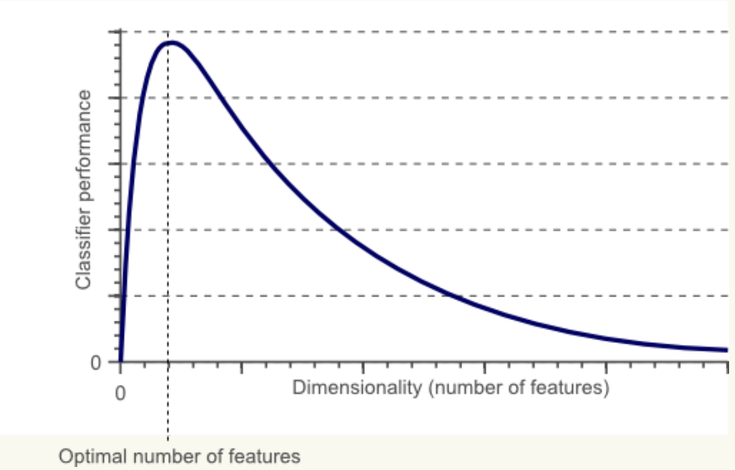
MENGURANGI DIMENSI SUATU DATASET



DigitalSkola

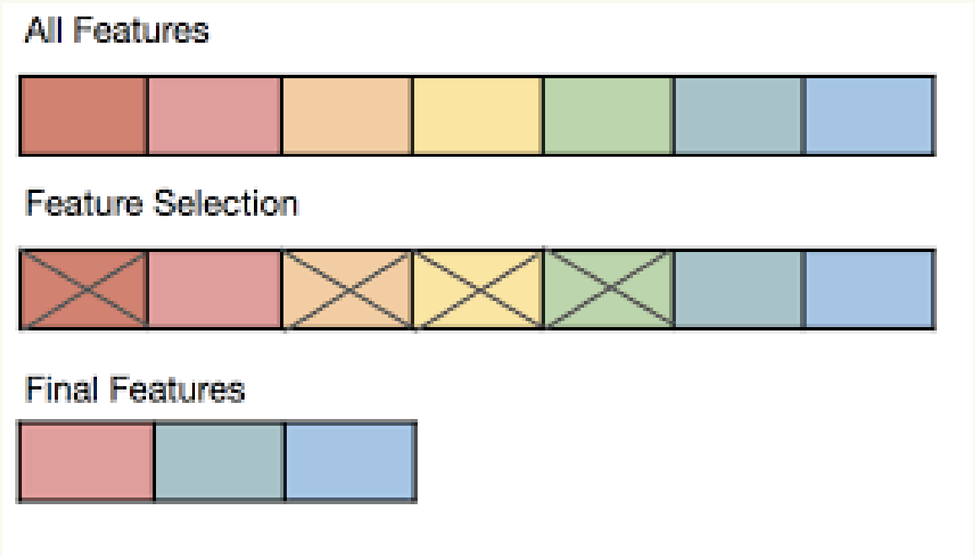
Benefit of Dimensionality Reduction

- Mengurangi misleading data yang membuat akurasi model meningkat
- Mengurangi dimensi, mengurangi komputasi
- Mengurangi feature yang redundant

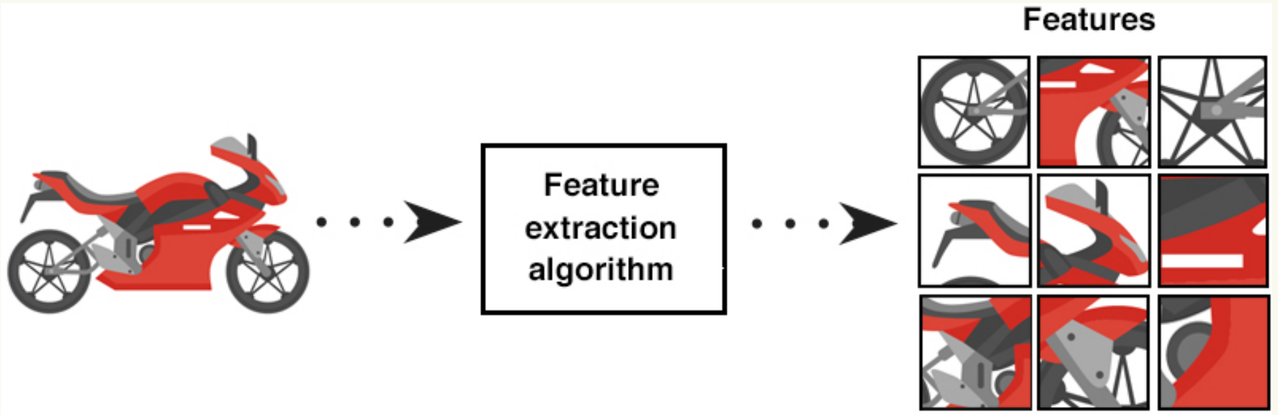


TERDAPAT **DUA TEKNIK** DI DALAM DIMENSIONALITY REDUCTION:

FEATURE SELECTION



FEATURE EXTRACTION



DigitalSkola

Binary Classification Examples

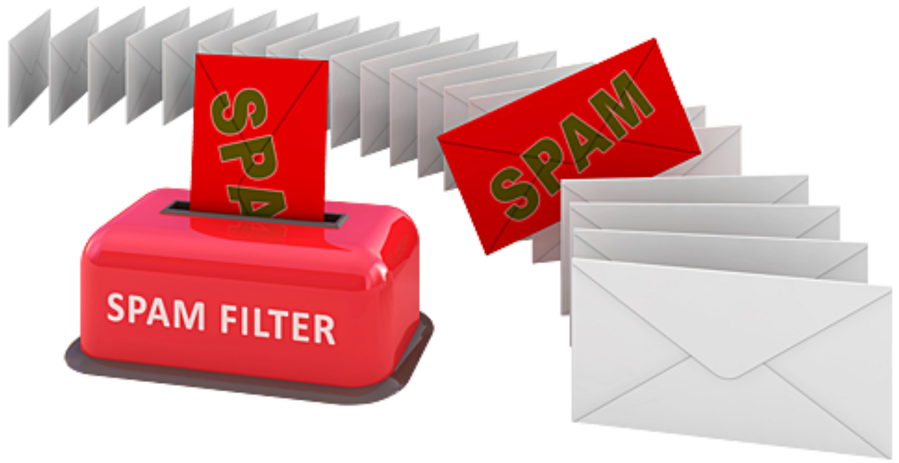
Objective

Detect spam or not spam mail

Target Variable

is_spam

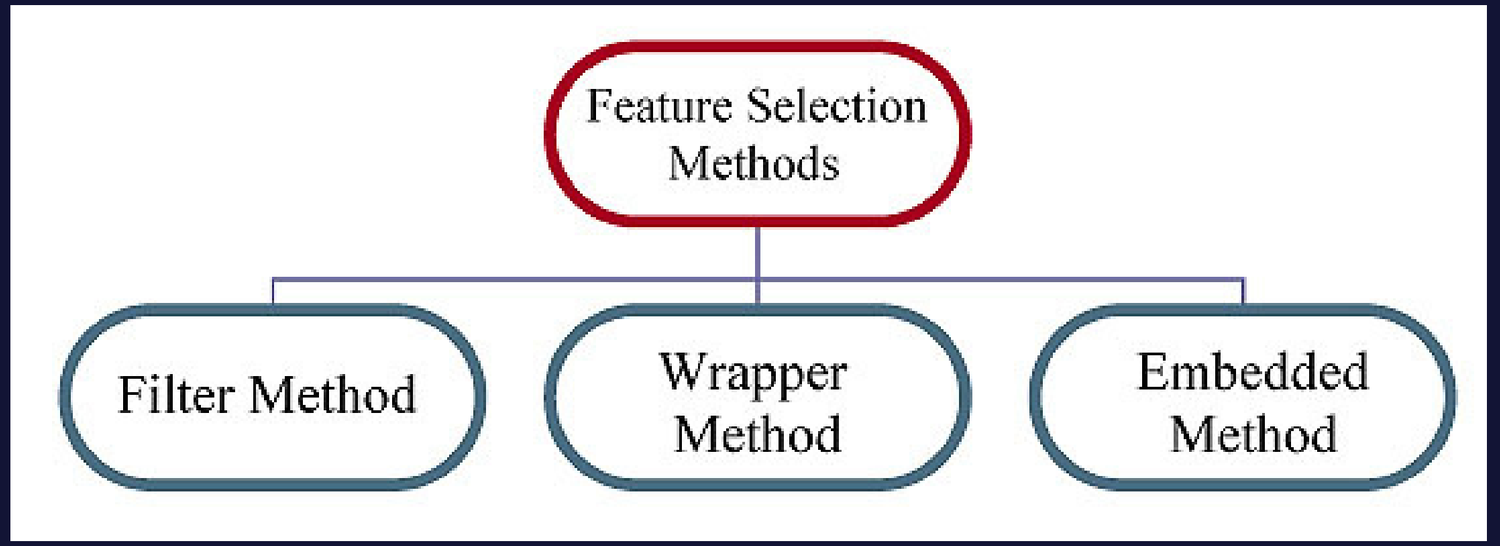
- 1 (spam)
- 0 (not spam)



Feature Selection

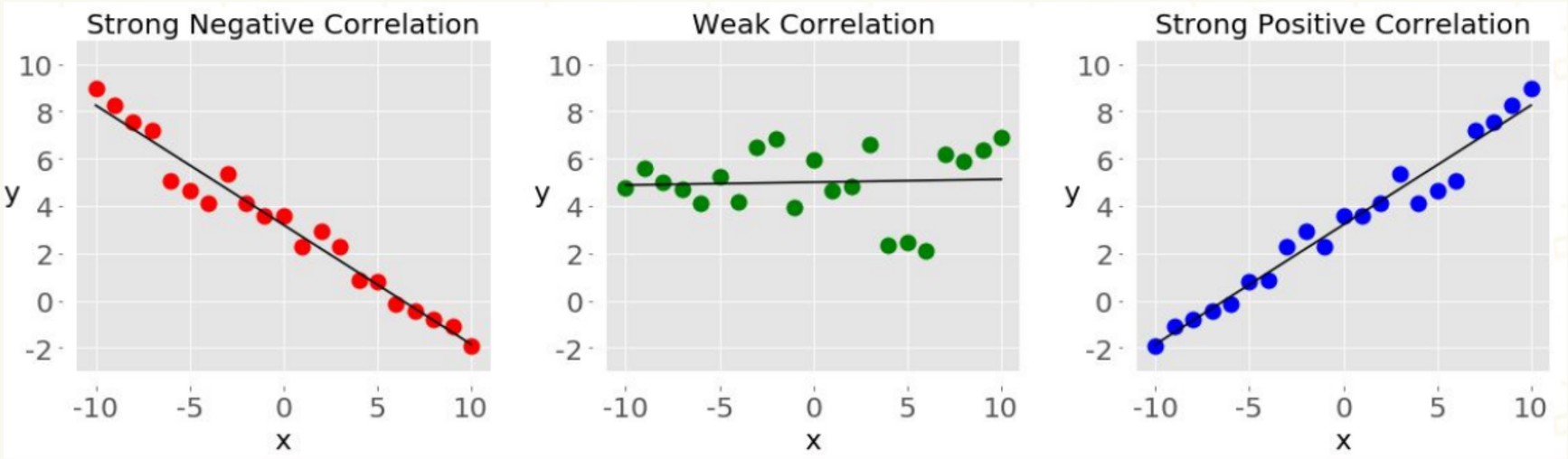
PROSES MEMILIH SUBSET DARI FITUR-FITUR RELEVAN DARI SELURUH FITUR YANG ADA DI DATASET. BEBERAPA HAL KEUNTUNGAN FEATURE SELECTION:

- Mengurangi waktu komputasi
- Mengurangi data yang tidak relevan
- Meningkatkan akurasi



Filter Method

METODE FILTER DIGUNAKAN DENGAN MELIHAT FITUR-FITUR YANG MEMILIKI KORELASI YANG TINGGI.



Pada filter metode ini kita menggunakan metode statistik yaitu **ANOVA** yang digunakan untuk menganalisis varians untuk menentukan jika rata-rata dari lebih dari dua populasi adalah sama

Filter Method

```
from sklearn.feature_selection import SelectKBest, f_classif
```

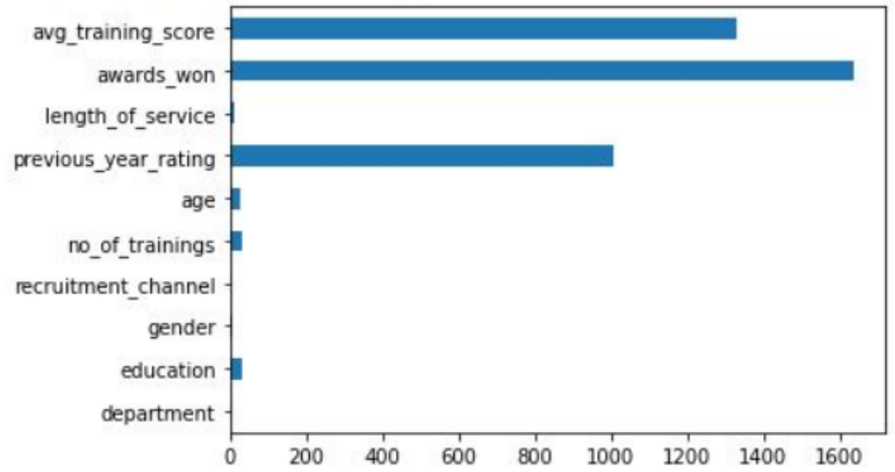
```
filter = SelectKBest(f_classif, k=5)
filter.fit(X_train, y_train)

X_train_new = filter.transform(X_train)
X_test_new = filter.transform(X_test)

print("Before feature selection", X_train.shape)
print("After feature selection", X_train_new.shape)
```

Features = 5
Selected Features = avg_training_score, awards_worn, previous_year_rating, education, no_of_trainings

Baseline ML (Logistic Regression)	91.79%
Logistic Regression + Anova	92.03%

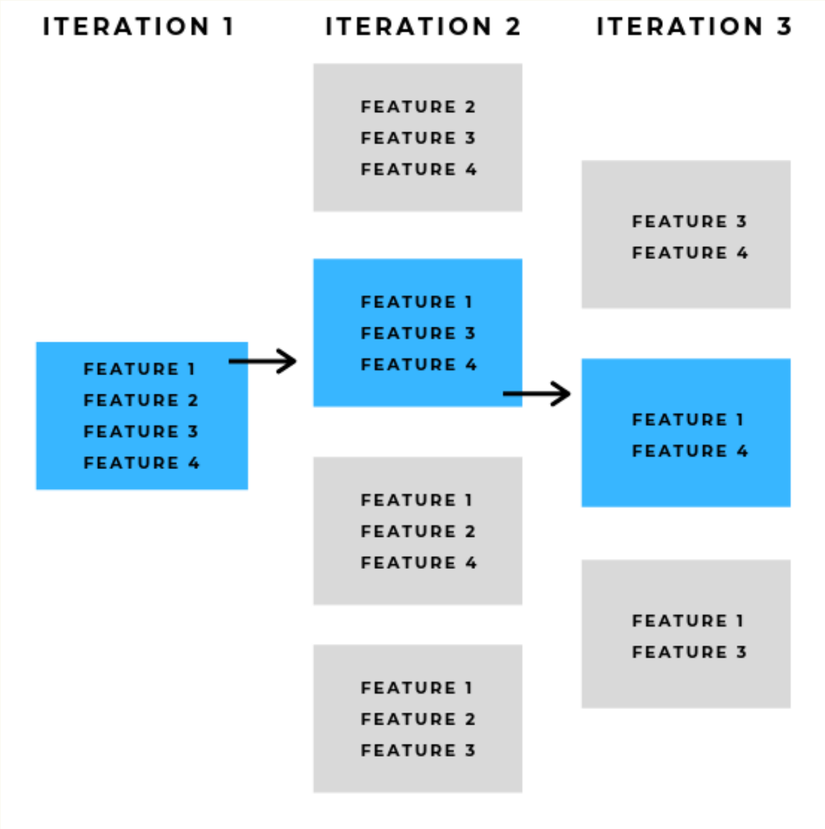
```
Before feature selection (37104, 10)
After feature selection (37104, 5)
```

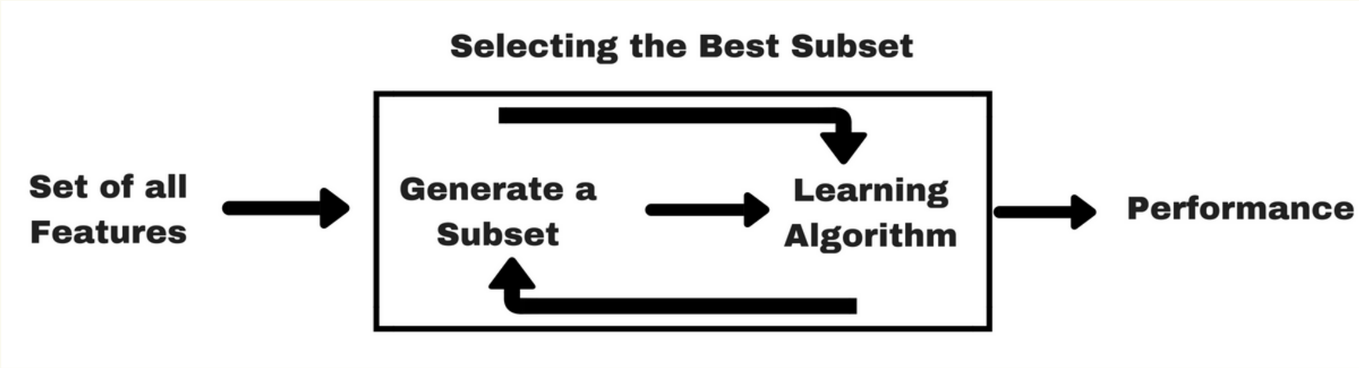
```
Score of features [1.58485364e-01 3.26275563e+01 6.63837086e+00 1.47834343e+00
 3.00369928e+01 2.54122347e+01 1.00475625e+03 1.02120870e+01
 1.63478931e+03 1.33002169e+03]
```



Wrapper Method

METODE WRAPPER DIGUNAKAN UNTUK MENEMUKAN KOMBINASI VARIABEL YANG TERBAIK. SALAH SATU METODE WRAPPER ADALAH **RFE (*RECURSIVE FEATURE ELIMINATION*)**.





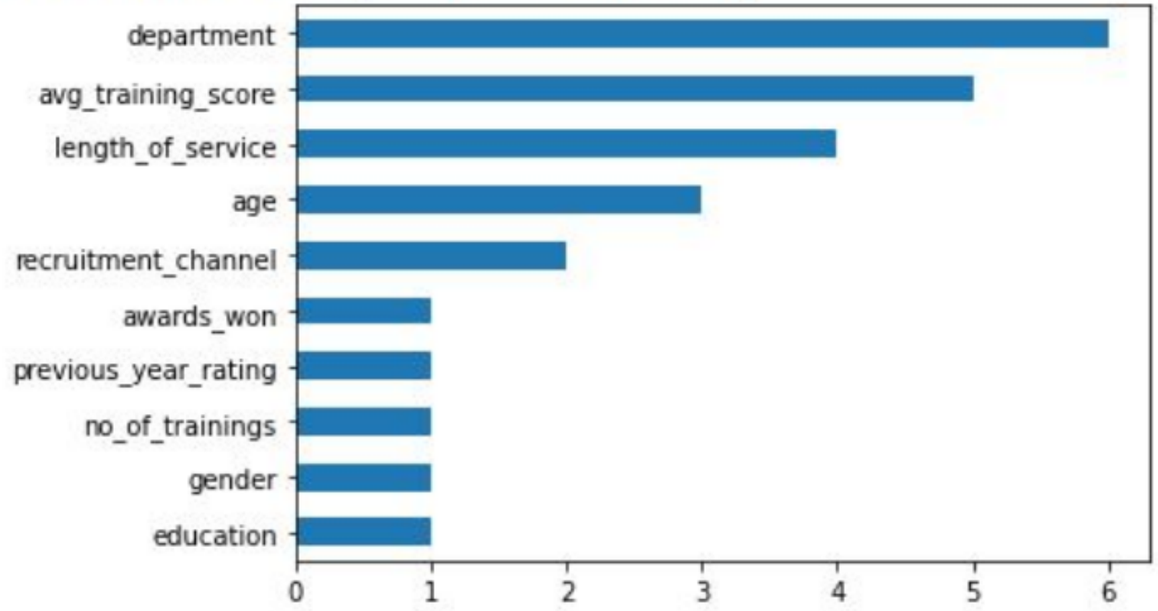
Wrapper Method

```
from sklearn.feature_selection import RFE

wrapper = RFE(clf, n_features_to_select=5)
wrapper.fit(X_train, y_train)

X_train_wrapper = wrapper.transform(X_train)
X_test_wrapper = wrapper.transform(X_test)
```

Features = 5
Selected Features = avg_training_score, awards_worn, previous_year_rating, education, no_of_trainings

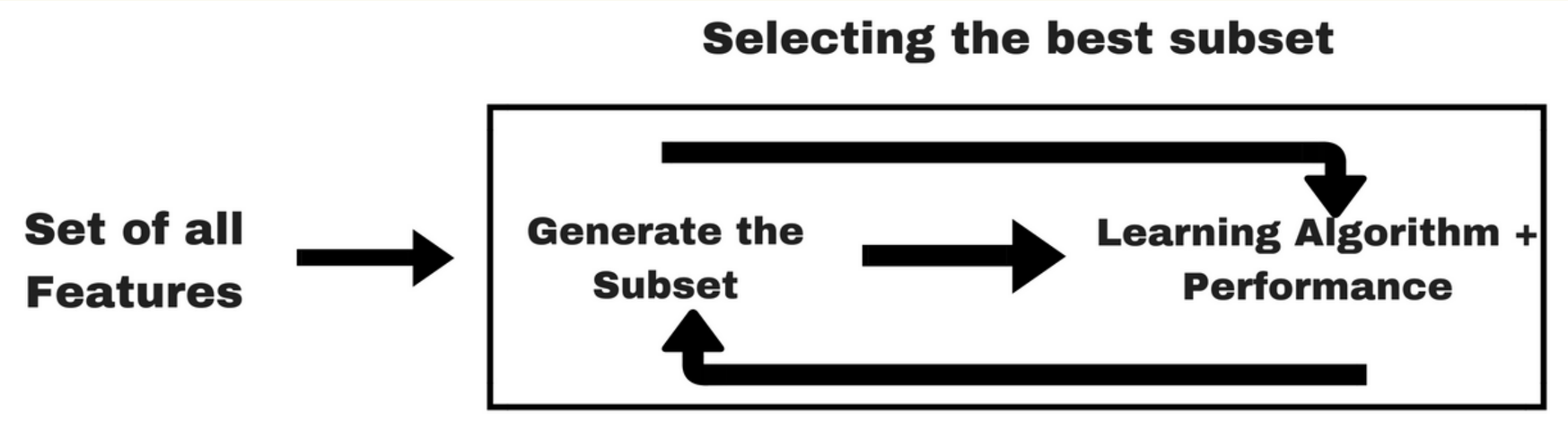
Baseline ML (Logistic Regression)	91.79%
Logistic Regression + Anova	92.03%
Logistic Regression + RFE	91.83%

```
Before feature selection (37104, 10)
After feature selection (37104, 5)

Score of features [6 1 1 2 1 3 1 4 1 5]
```



Embedded Method

METODE EMBEDDED INI DIGUNAKAN UNTUK MEMILIH FITUR-FITUR MANA SAJA YANG DIGUNAKAN DARI HASIL PERFORMA ALGORITMA *MACHINE LEARNING* MODEL.



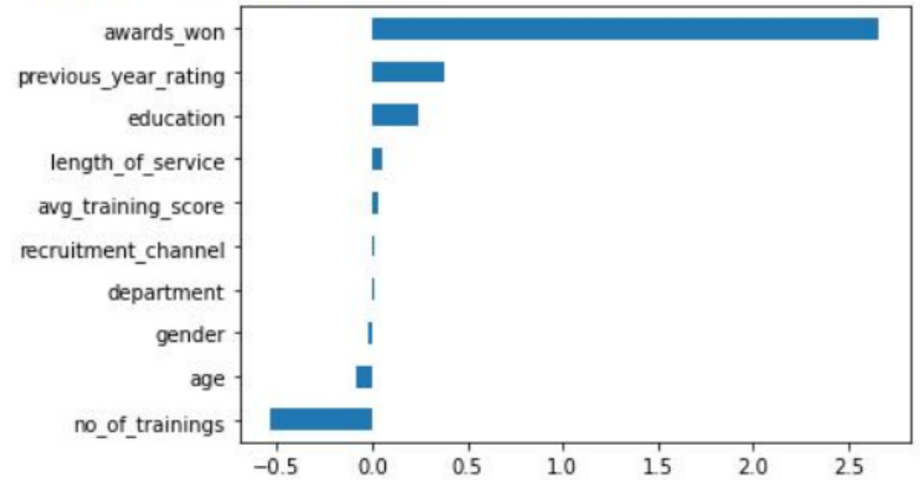
Embedded Method

```
from sklearn.feature_selection import SelectFromModel

clf = LogisticRegression()
clf_feature = SelectFromModel(clf)

clf_feature.fit(X_train, y_train)

X_train_importance = clf_feature.transform(X_train)
X_test_importance = clf_feature.transform(X_test)
```

Features = 2
Selected Features = awards_worn, previous_year_rating

Baseline ML (Logistic Regression)	91.79%
Logistic Regression + Anova	92.03%
Logistic Regression + RFE	91.83%
Logistic Regression + Feature Importance	91.67%

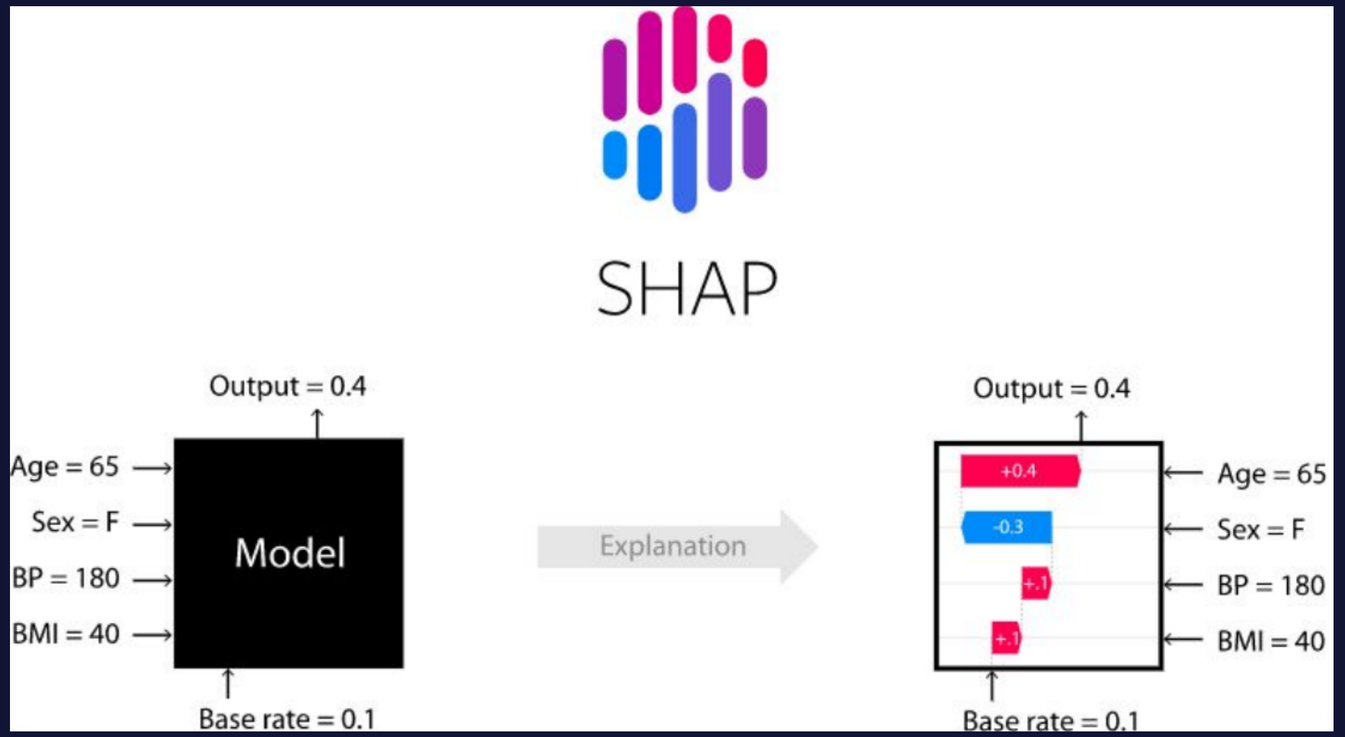
```
Coef [ 0.01008519  0.2411234  -0.0189336   0.01141018 -0.53517601 -0.08410499
  0.37662892  0.05672754  2.66393455  0.02762389]
Treshold 0.4025748276559864
```



DigitalSkola

EXPLAINABLE AI (BONUS)

SHAP (SHapley Additive exPlanations) is a game-theoretic approach to explain the output of any machine learning model. It connects optimal credit allocation with local explanations using the classic Shapley values from game theory and their related extensions



FEATURES PUSHING THE PREDICTION HIGHER ARE SHOWN IN RED, THOSE PUSHING THE PREDICTION LOWER ARE IN BLUE

THANK YOU!

By OMICRON