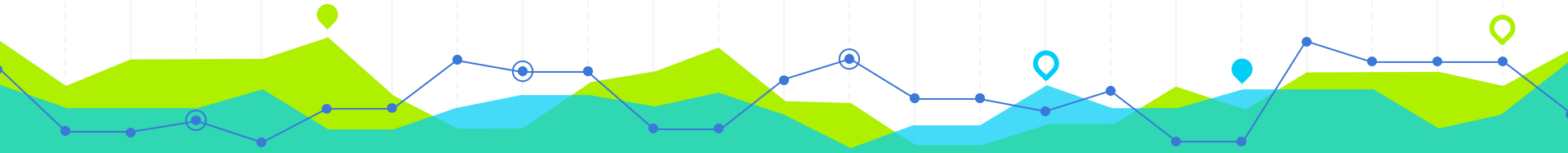
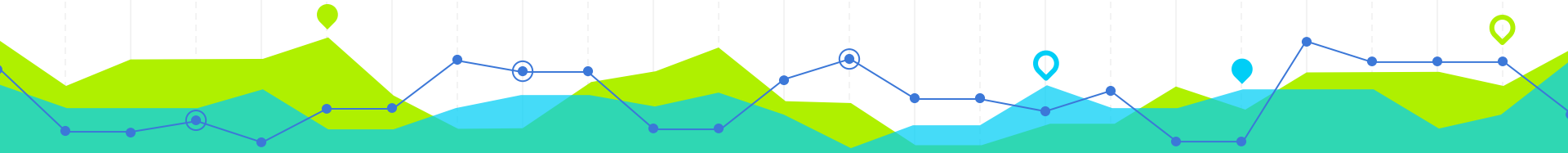


BANDUNG TRAFFIC FLOW PREDICTION WITH MACHINE LEARNING



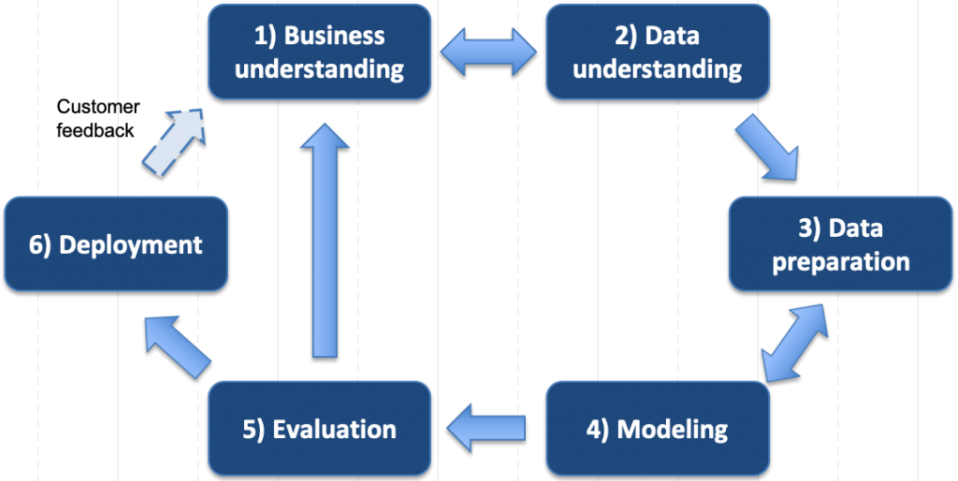
METODOLOGI

- ▶ METODOLOGI
- BUSINESS UNDERSTANDING
- DATA UNDERSTANDING
- DATA PREPARATION
- FEATURE ENGINEERING
- MODELLING
- EVALUATION

AI Based Model Framework

CRISP – DM

CRoss Industry Standard Processing for Data Mining



Source : OCR Case Structure - (duke.edu)

Generic Task and Output of The CRISP-DM AI Framework

Business Understanding

Business Objectives
Background
Business Objectives
Business Success Criteria

Data Understanding

Data Description
Describe Data
Data Description Report
Exploration Data Analysis
Data Exploration Report
Visualization for Business Insight

Data Preparation

Handling Missing Value
Simple Imputer
Clean Data
Drop Useless Data

Feature Engineering

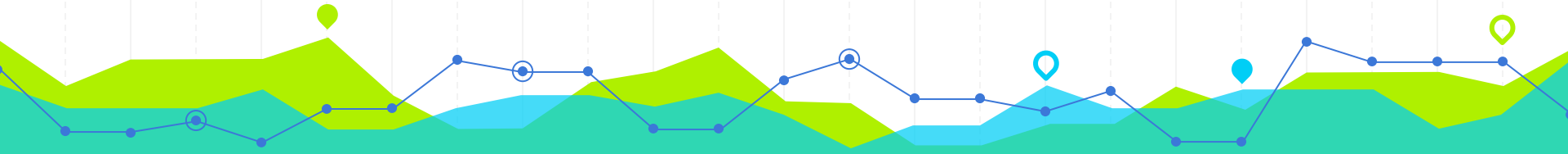
Feature Transformation
Scaling Numerical Data
Encoding Categorical Data
Feature Selection
Multicolinieearity

Modelling

Tunning Optimal Hyperparameter
Gridsearch CV
Modelling with 7 Classification Models
Logistic Regression
Decision Tree
XGBoost
AdaBoost
Random Forest
Classification Report
Feature Importance

Evaluation

Evaluate Best Model
Conclusions



BUSINESS UNDERSTANDING

METODOLOGI

► BUSINESS UNDERSTANDING

DATA UNDERSTANDING

DATA PREPARATION

FEATURE ENGINEERING

MODELLING

EVALUATION

Business Goal

1 Latar Belakang / Konteks Bisnis

Kota Bandung adalah salah satu kota di Indonesia dengan kemacetan sebagai masalah utama yang melekat. Pada case kali ini, akan diprediksi Traffic Flow di jalan-jalan Kota Bandung pada Jam-jam tertentu. Data yang digunakan diambil dari https://opendata.jabarprov.go.id/id/dataset. Yang mana hasil ini diharapkan dapat bermanfaat bagi pemerintah kota dalam mengatasi kemacetan di Kota Bandung.

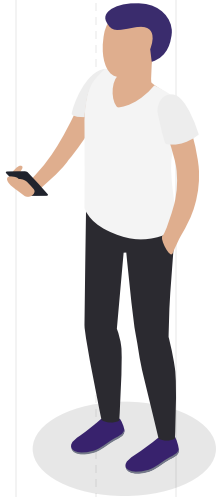
2 Tujuan Bisnis

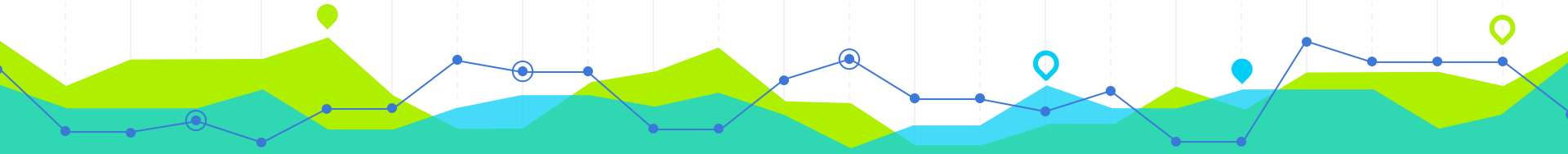
Mengurangi tingkat kemacetan pada jam-jam tertentu di Kota Bandung.

Use Case dan Deskripsi

Traffic Flow Prediction

- Usecase utama pada project ini adalah memprediksi tingkat kemacetan pada wilayah dan jam-jam tertentu di kota Bandung.
- Menetapkan fitur-fitur pada data traffic yang ditargetkan untuk menjadi parameter pemerintah kota dalam memberikan solusi kemacetan.
- Membuat mesin klasifikasi untuk menentukan apakah daerah tertentu di jam tertentu apakah jam levelnya 1,2,3 atau 4.





DATA UNDERSTANDING

METODOLOGI

BUSINESS UNDERSTANDING

► DATA UNDERSTANDING

DATA PREPARATION

FEATURE ENGINEERING

MODELLING

EVALUATION

Data Source yang akan Digunakan

Trainset

https://opendata.jabarprov.go.id/id/dataset

Data Exploration

Data yang akan dianalisa

id	time	kemendagri_kabupaten_kode	kemendagri_kabupaten_nama	street	type	avg_location	total_records	date
5930351	2022-07-06 00:00:00.000	32.73	KOTA BANDUNG	Batununggal Indah 2	WEATHERHAZARD	[107.62634049999998, -6.962361499999998]	120	2022-07-06
5930352	2022-07-06 00:00:00.000	32.73	KOTA BANDUNG	Cibaduyut Raya	ROAD_CLOSED	[107.59526000000004, -6.9472129999999925]	60	2022-07-06
5930353	2022-07-06 00:00:00.000	32.73	KOTA BANDUNG	Gerbang Tol Gede Bage	ROAD_CLOSED	[107.69057999999981, -6.959956500000006]	120	2022-07-06
5930354	2022-07-06 00:00:00.000	32.73	KOTA BANDUNG	Jenderal Ahmad Yani	WEATHERHAZARD	[107.65976500000006, -6.902228999999994]	60	2022-07-06
5930355	2022-07-06 00:00:00.000	32.73	KOTA BANDUNG	KH Wahid Hasyim	ROAD_CLOSED	[107.58966274999973, -6.945608999999994]	240	2022-07-06

time	kemendagri_kabupaten_kode	kemendagri_kabupaten_nama	street	jam_level	median_length	median_delay_seconds	median_regular_speed
2022-07-06 07:00:00.000	32.73	KOTA BANDUNG	Terusan Buah Batu	4	1922.0	657.0	15.770000
2022-07-06 07:00:00.000	32.73	KOTA BANDUNG	Jenderal AH Nasution	3	1819.0	421.0	17.939999
2022-07-06 07:00:00.000	32.73	KOTA BANDUNG	Jenderal AH Nasution	4	1064.0	586.0	14.520000
2022-07-06 07:00:00.000	32.73	KOTA BANDUNG	N11 Soekarno-Hatta	4	919.0	558.5	15.095000
2022-07-06 07:00:00.000	32.73	KOTA BANDUNG	Terusan Buah Batu	3	2024.0	599.0	15.830000

Data Exploration

Data demerger kemudian terbagi menjadi data numerik dan data kategorik.

	median_length	median_delay_seconds	median_regular_speed	median_seconds	median_speed	jam_sibuk
1851	2010.0	713.0	24.36	903.0	7.26	0
1172	545.0	487.0	8.71	807.0	2.43	0
4314	2010.0	679.0	16.51	870.0	8.30	0
1037	805.0	697.0	9.88	802.0	3.85	0
1022	673.0	962.0	5.82	1043.0	2.21	0

	time	street	type	avg_location	geometry	hari	jam
2022-07-06 07:00:00.000	Jenderal AH Nasution	JAM	[107.67224464444448, -6.905886155555552]	LINESTRING (107.678021 -6.904799, 107.677631 -...	Rabu	07:00
2022-07-06 07:00:00.000	Jenderal AH Nasution	JAM	[107.67224464444448, -6.905886155555552]	MULTILINESTRING ((107.659819 -6.902242, 107.66...	Rabu	07:00
2022-07-06 07:00:00.000	N11 Soekarno-Hatta	JAM	[107.63925460000002, -6.946320180000007]	MULTILINESTRING ((107.641468 -6.945648, 107.64...	Rabu	07:00
2022-07-06 08:00:00.000	Terusan Buah Batu	JAM	[107.63885299999995, -6.954049999999997]	MULTILINESTRING ((107.638019 -6.965016, 107.63...	Rabu	08:00
2022-07-06 08:00:00.000	Terusan Buah Batu	JAM	[107.63885299999995, -6.954049999999997]	MULTILINESTRING ((107.637998 -6.965335, 107.63...	Rabu	08:00

Data Description

Describe data

Statistik Deskriptif Data Numerik

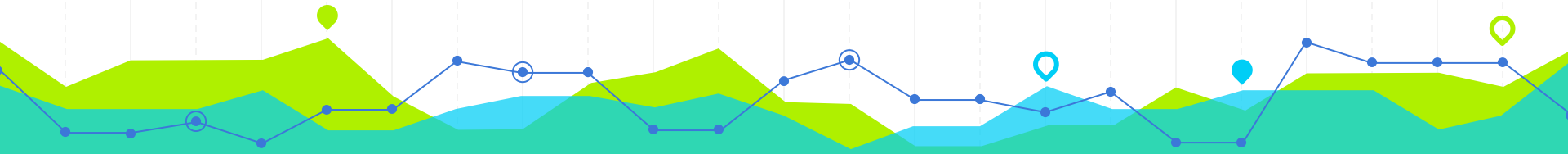
```
1 train_num = df.select_dtypes(include = 'number')
2 train_num.describe()
```

	jam_level	median_length	median_delay_seconds	median_regular_speed	median_seconds	median_speed	median_jam_level	jam_sibuk
count	6074.000000	6074.000000	6074.000000	6074.000000	6074.000000	6074.000000	6074.000000	6074.000000
mean	3.538360	1519.663072	621.125453	18.684859	765.170810	7.798781	3.538360	0.360718
std	0.593555	959.019348	305.019818	12.153065	340.149042	4.884011	0.593555	0.480248
min	1.000000	500.000000	-97.000000	2.450000	132.000000	0.880000	1.000000	0.000000
25%	3.000000	841.000000	421.625000	12.490000	539.125000	4.840000	3.000000	0.000000
50%	4.000000	1293.000000	585.000000	16.500000	721.000000	6.845000	4.000000	0.000000
75%	4.000000	1868.250000	779.000000	20.850000	940.875000	9.493750	4.000000	1.000000
max	4.000000	13201.000000	4368.500000	85.770000	4295.500000	53.490000	4.000000	1.000000

Statistik Deskriptif Data Kategorik

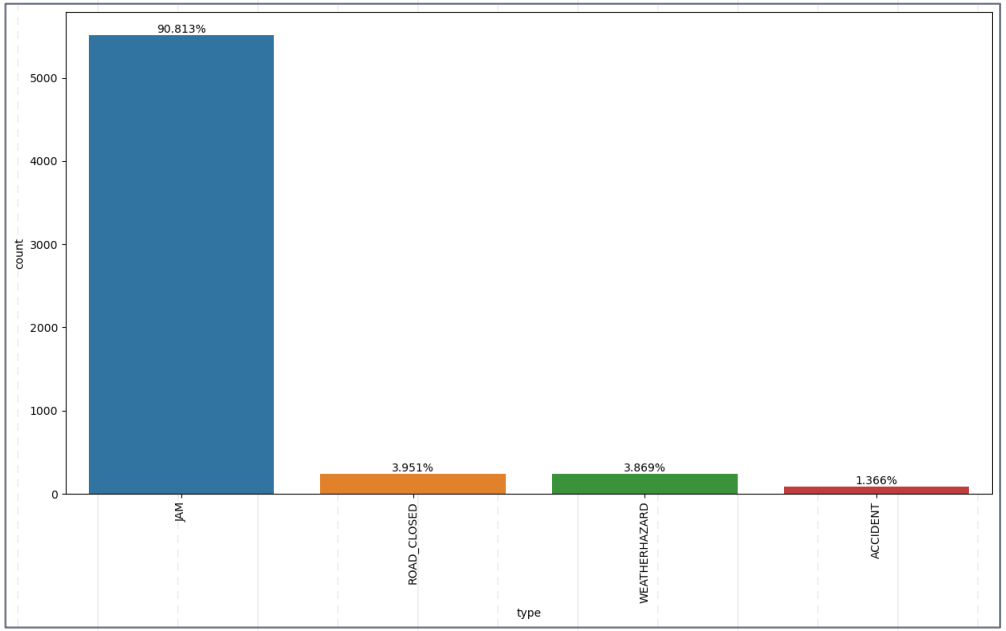
```
1 train_obj = df.select_dtypes(include = 'object')
2 train_obj.describe()
```

	time	street	type	avg_location	geometry	hari	jam
count	6074	6074	6074	6074	6074	6074	6074
unique	740	187	4	3679	5552	7	24
top	2022-08-19 17:00:00.000	N11 Soekarno-Hatta	JAM	[107.59526000000004, -6.9472129999999925]	LINESTRING (107.58148 -6.882219, 107.581484 -6...	Jumat	17:00
freq	74	648	5516	105	7	1438	896



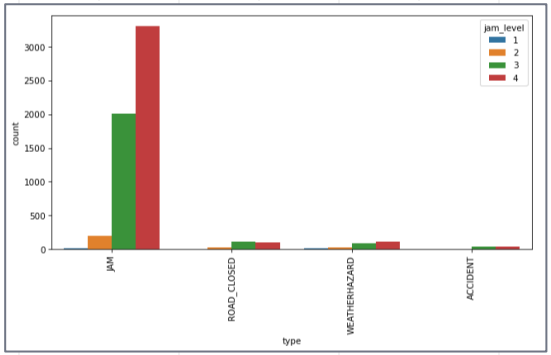
EXPLORATORY DATA ANALYSIS

Kondisi data

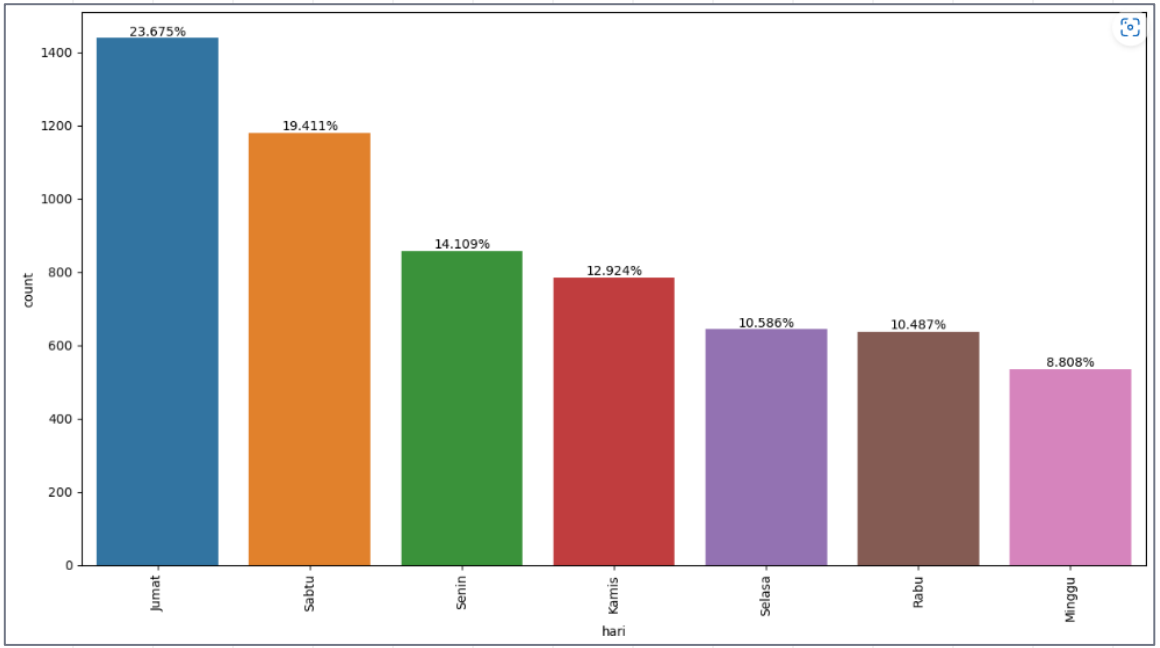


Insight

1. Dari data, kemacetan mayoritas dikarenakan aktivitas kendaraan yang padat.
2. Tingkat kemacetan yang disebabkan oleh penutupan jalan, cuaca buruk atau kecelakaan kurang dari 10%

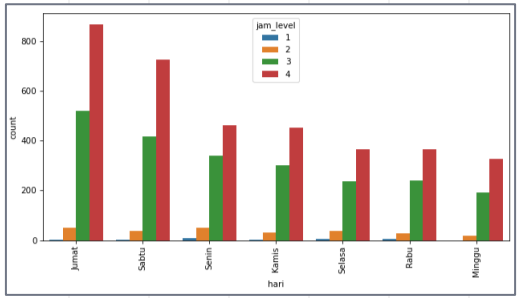


Kondisi data

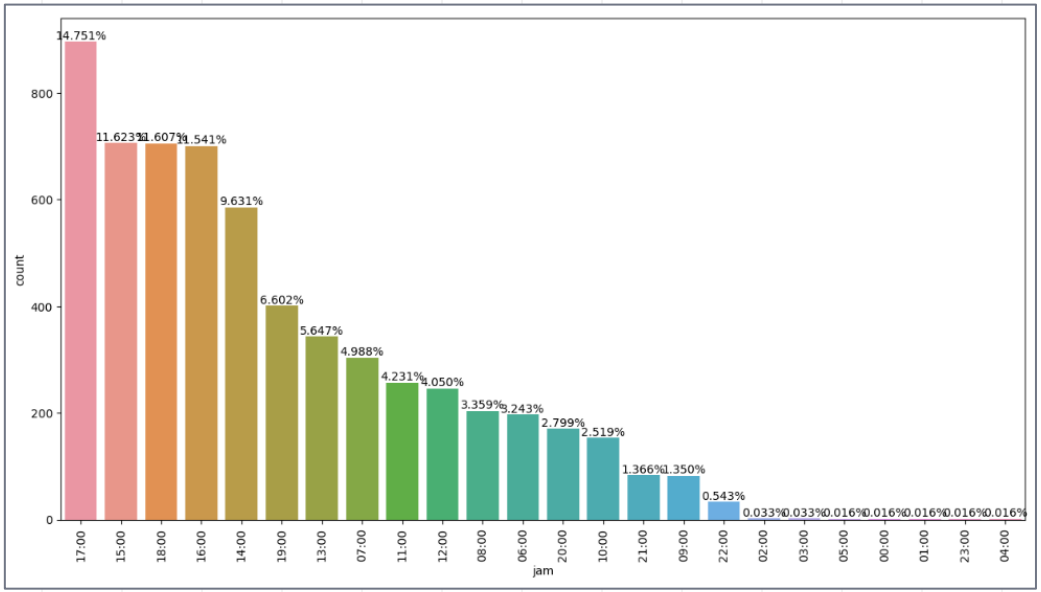


Insight

1. Data kemacetan yang tercatat mayoritas terjadi di hari Jumat, Sabtu, dan Senin.
2. Kemacetan di hari Jumat dan Senin terjadi di jam-jam sibuk kerja.
3. Kemacetan di hari Sabtu didominasi pada jam-jam malam (Saturday Night).

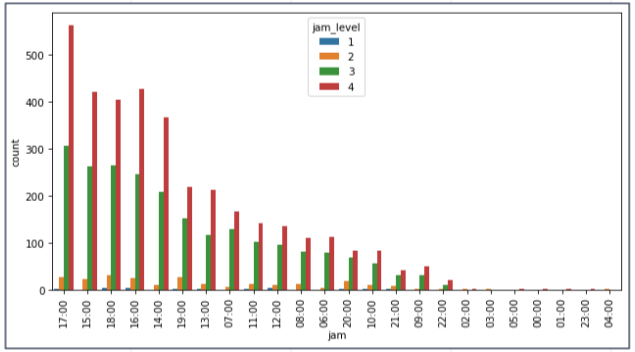


Kondisi data

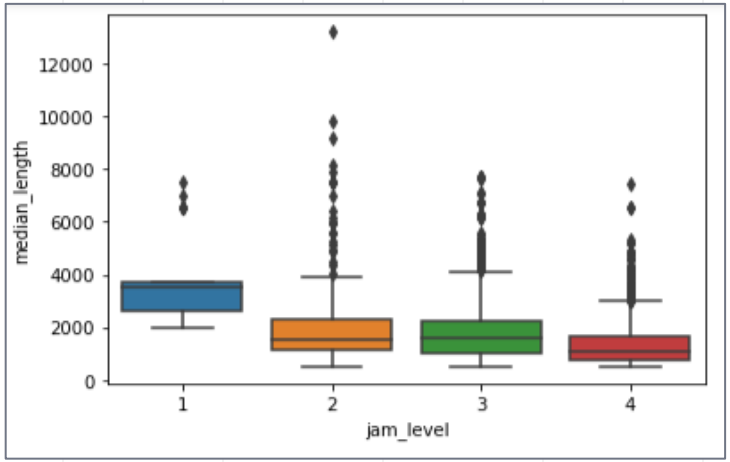


Insight

1. Jam-jam sibuk didominasi di pukul 17.00, 15.00,18.00, dan 16.00
2. Jam-jam sibuk terjadi di jam-jam kerja (terutama jam pulang)

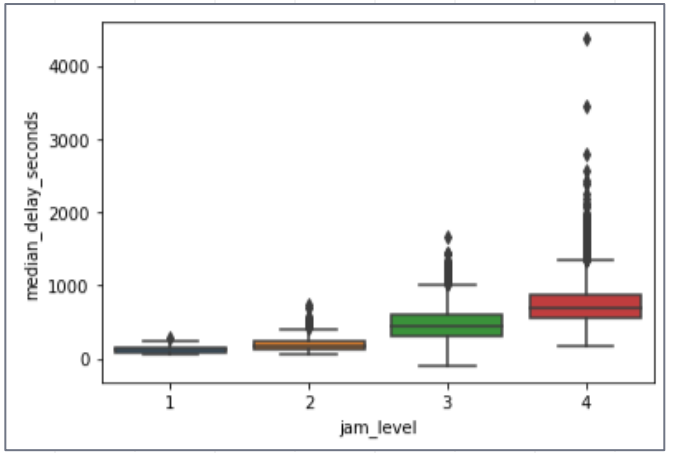


EDA



Insight

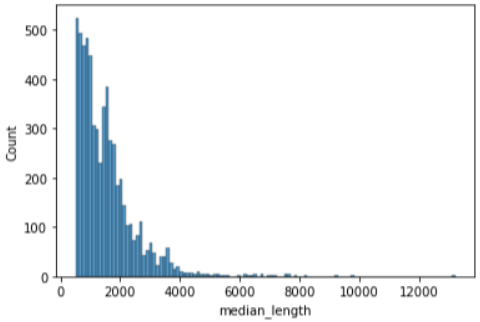
1. Kepadatan yang terjadi memiliki jarak yang pendek di kisaran 1,5 km

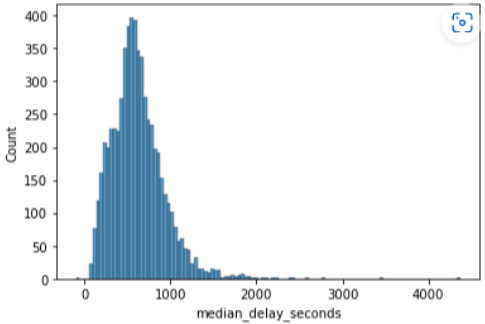


Insight

1. Waktu tunggu kemacetan terparah cukup tinggi di kisaran 20 menit

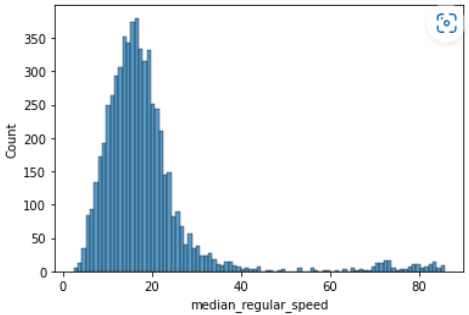
EDA – Distribusi Fitur

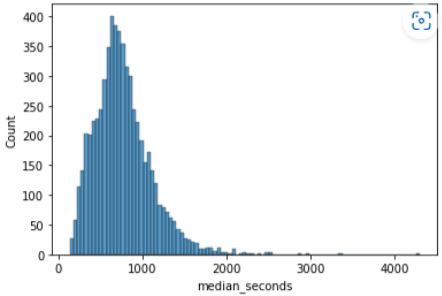


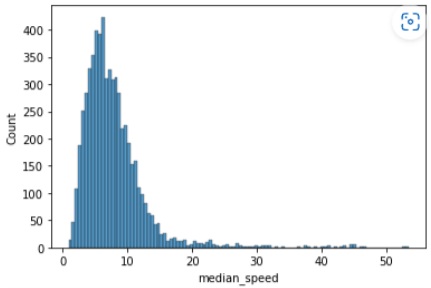


Insight

Distribusi dari Panjang kemacetan, waktu tunggu, dan kecepatan traffic semua skew kanan.







DATA PREPARATION

METODOLOGI

BUSINESS UNDERSTANDING

DATA UNDERSTANDING

► DATA PREPARATION

FEATURE ENGINEERING

MODELLING

EVALUATION

Data Cleaning

Tidak ada missing value data

```
missing=pd.DataFrame(X_train.isna().sum())
missing['Jumlah Missing Value']=missing
missing = missing.drop([0], axis=1)
percent = pd.DataFrame(X_train.isnull().sum() * 100 / len(X_train))
percent['Persentase Missing Value']=percent
percent = percent.drop([0], axis=1)
percentmissing=pd.concat([missing, percent], axis=1)
percentmissing.transpose()
```

	type	median_length	median_delay_seconds	median_regular_speed	median_seconds	median_speed	hari	jam	jam_sibuk
Jumlah Missing Value	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Persentase Missing Value	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Data Cleaning

Drop Negative Data

```
for column in train.select_dtypes(include = 'number').columns:
  print('Jumlah negative value pada kolom', column, ':', len(train.select_dtypes(include = 'number')[train[column] < 0][colu
```

Pada data numerik, terdapat satu kolom yang terdapat nilai **negatif**, sehingga akan dibuang pula **tiap row** yang mengandung nilai **negative** tersebut.

FEATURE ENGINEERING

METODOLOGI

BUSINESS UNDERSTANDING

DATA UNDERSTANDING

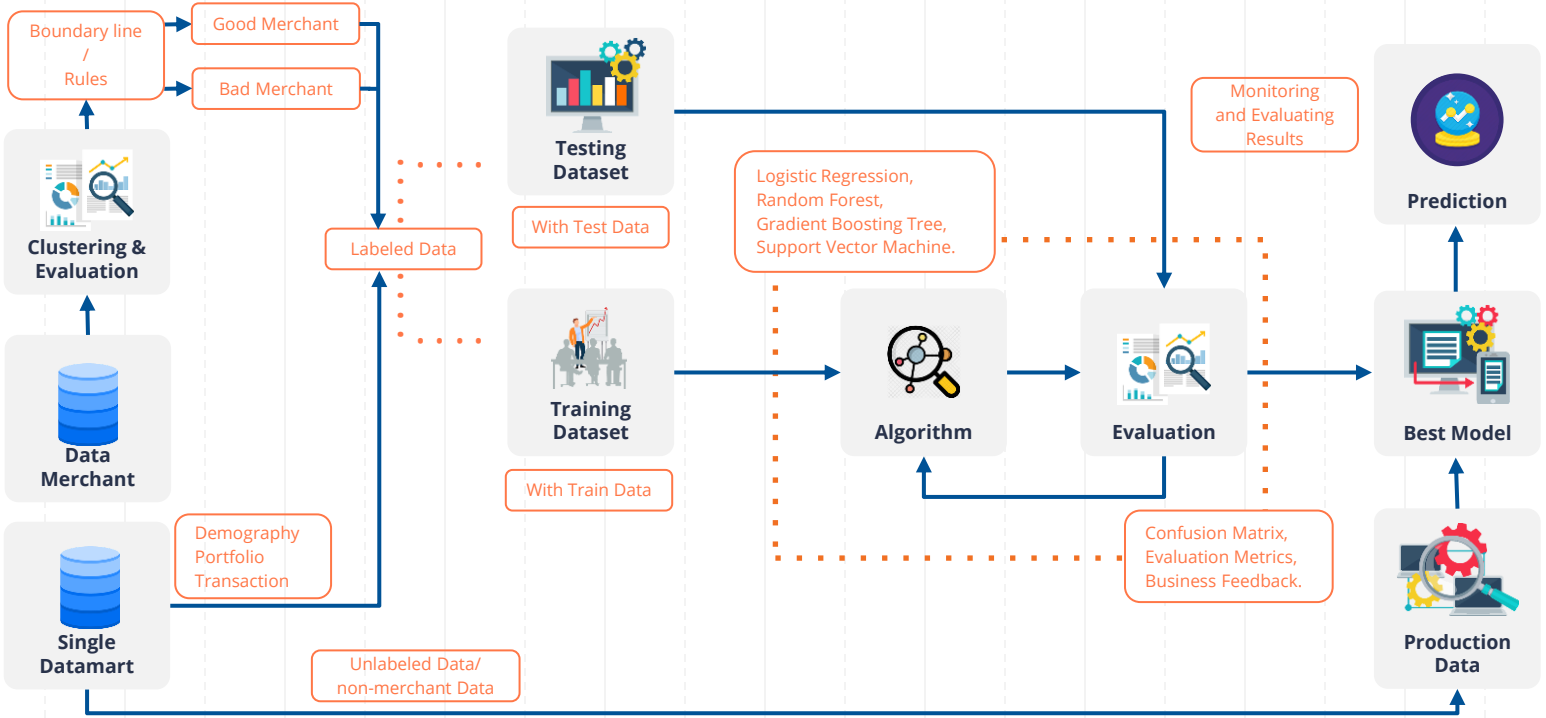
DATA PREPARATION

► FEATURE ENGINEERING

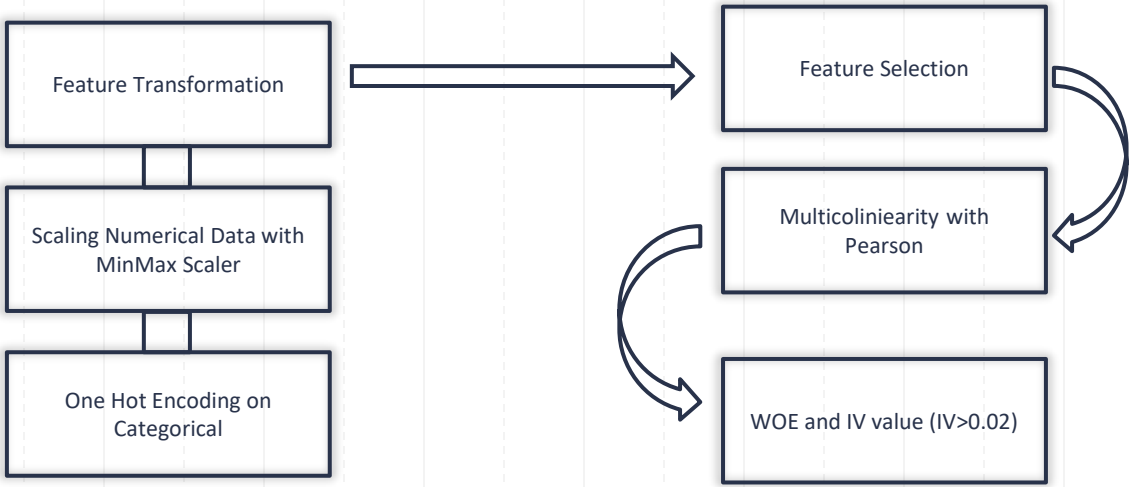
MODELLING

EVALUATION

Machine Learning Development Life Cycle



Feature Engineering



Feature Transformation

```
1 from sklearn.preprocessing import MinMaxScaler
2 mmscaler = MinMaxScaler()
3 mmscaler.fit(train_num)
```

```
MinMaxScaler()
```

```
1 train_num_scaled = pd.DataFrame(mmscaler.transform(train_num))
2 train_num_scaled.columns = train_num.columns
3 train_num_scaled
```

Scaling Numerical Feature with MinMax Scaler

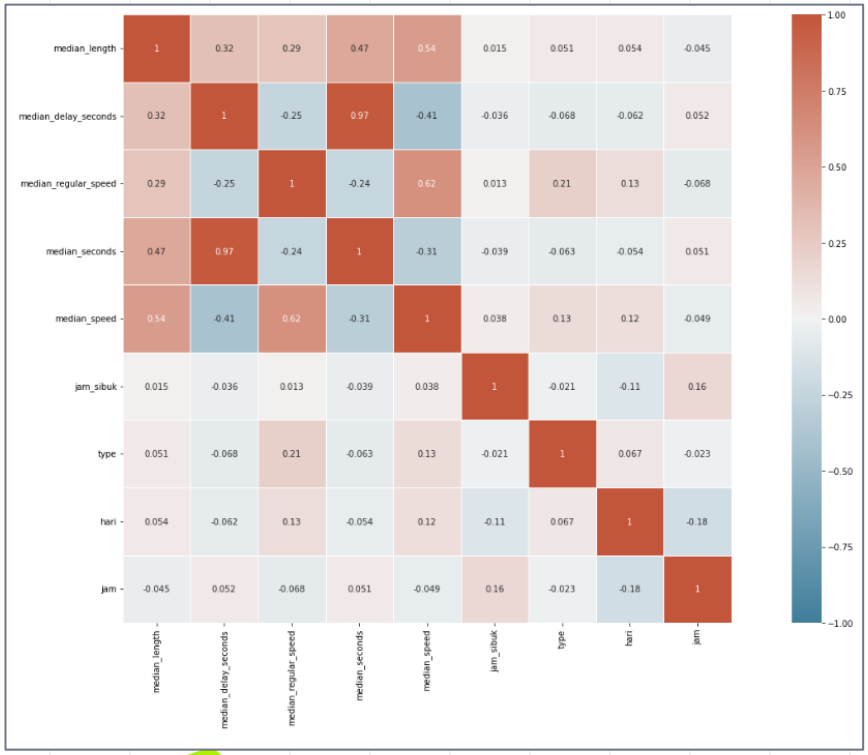
```
1 from sklearn.preprocessing import OneHotEncoder
2 ohe = OneHotEncoder(handle_unknown='ignore')
3 ohe.fit(train_obj)
```

```
OneHotEncoder(handle_unknown='ignore')
```

```
1 train_obj_ohe = pd.DataFrame(ohe.transform(train_obj).toarray())
2 train_obj_ohe.columns = ohe.get_feature_names(train_obj.columns)
3 train_obj_ohe
```

Encoding Categorical Feature with One Hot Encoder

Feature Selection
Multicolinieearity

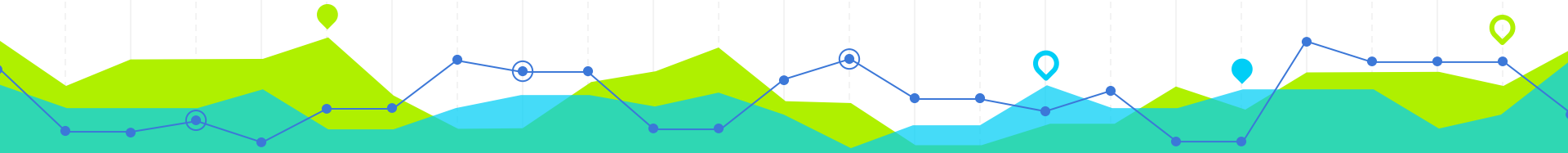


Dilakukan cek korelasi Pearson antar fitur, dengan threshold 0.7, lalu didapatkan 8 fitur tersisa.

```
feat_cols=['median_length', 'median_delay_seconds', 'median_regular_speed',
       'median_seconds', 'median_speed', 'jam_sibuk', 'type', 'hari', 'jam']
import remove_col as rc
dfc=train_transformed[feat_cols]
rc.remove_collinear_features(dfc, 0.7).columns
```

```
median_seconds | median_delay_seconds | 0.97
```

```
Index(['median_length', 'median_delay_seconds', 'median_regular_speed',
       'median_speed', 'jam_sibuk', 'type', 'hari', 'jam'],
      dtype='object')
```



MODELLING

METODOLOGI

BUSINESS UNDERSTANDING

DATA UNDERSTANDING

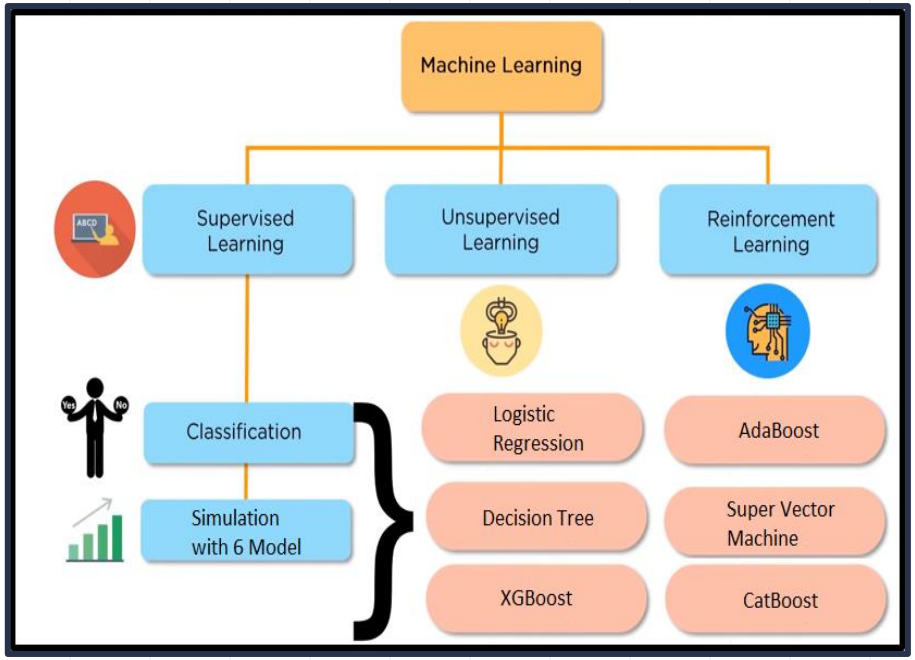
DATA PREPARATION

FEATURE ENGINEERING

► MODELLING

EVALUATION

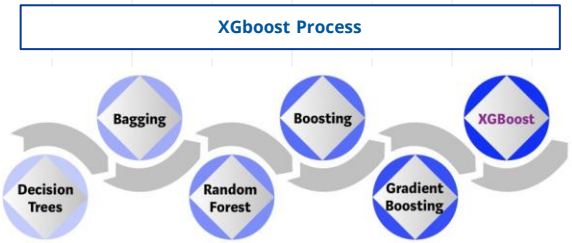
Modelling with Machine Learning

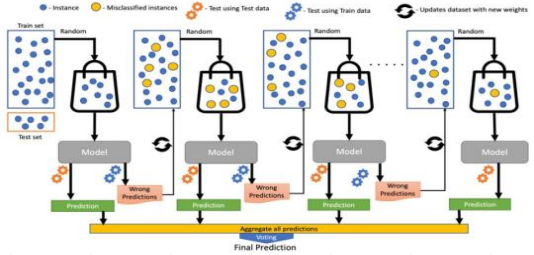


Dipilih 1 dari 6 model dengan performance yang paling baik

XGBoost Machine Learning

XGBoost is a **decision-tree-based ensemble** Machine Learning algorithm that uses a **gradient boosting framework**.





The two reasons to use XGBoost:

1. **Execution Speed**
2. **Model Performance**

Tree — base learner that generates weak rules at each iteration

Boosting

XGBoost

Ensemble Learning

decisions from multiple machine learning models are combined to reduce errors and improve prediction when compared to a single ML model.

Ensemble Learning:

- ❑ Bagging (parallel ensemble)
- ❑ Boosting (Sequential ensemble)

creates new weak learners (models) and sequentially combines their predictions to improve the overall performance of the model

$$F_i(x) = F_{i-1}(x) + f_i(x)$$

F(i) is current model, F(i-1) is previous model and f(i) represents weak model

Gradient Boosting

boosting algorithm where errors are minimized by a gradient descent algorithm and produce a model in the form of weak prediction models e.g. decision trees

$$w = w - \eta \nabla w$$

$$\nabla w = \frac{\partial L}{\partial w} \quad where\ L\ is\ loss$$

Optimized Gradient Boosting algorithm through parallel processing, tree-pruning, handling missing values, and regularization to avoid overfitting/bias

XGBoost with Hyper Tuning Parameter Classification Report

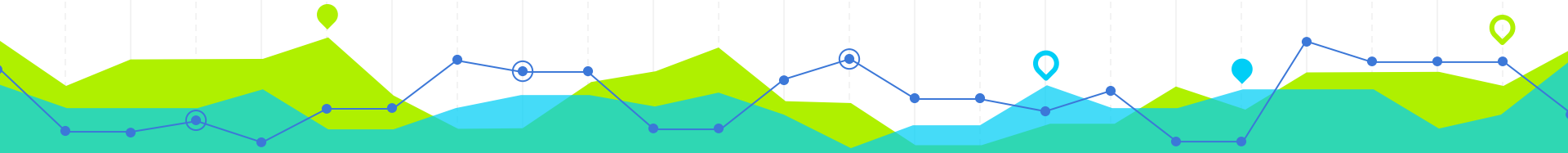
XGBoost Test Evaluation

	precision	recall	f1-score	support	pred	AUC
1	1.000000	0.800000	0.888889	5.0	4.0	0.998926
2	0.869565	0.816327	0.842105	49.0	46.0	0.995143
3	0.909953	0.855234	0.881745	449.0	422.0	0.976711
4	0.919246	0.959270	0.938832	712.0	743.0	0.982736
avg / total	0.914141	0.914403	0.913629	1215.0	1215.0	0.991927

XGBoost With Tuning Test Evaluation

	precision	recall	f1-score	support	pred	AUC
1	1.000000	0.600000	0.750000	5.0	3.0	0.998430
2	0.860000	0.877551	0.868687	49.0	50.0	0.993183
3	0.932039	0.855234	0.891986	449.0	412.0	0.974861
4	0.920000	0.969101	0.943912	712.0	750.0	0.981825
avg / total	0.922358	0.921811	0.920891	1215.0	1215.0	0.991209

Dipilih 1 dari 6 model dengan performance yang paling baik yaitu XGBoost, selain hasilnya yang cukup konsisten dalam memprediksi setiap label, dia juga memiliki hasil evaluasi yang konsisten antara train dan test nya.



EVALUATION

METODOLOGI

BUSINESS UNDERSTANDING

DATA UNDERSTANDING

DATA PREPARATION

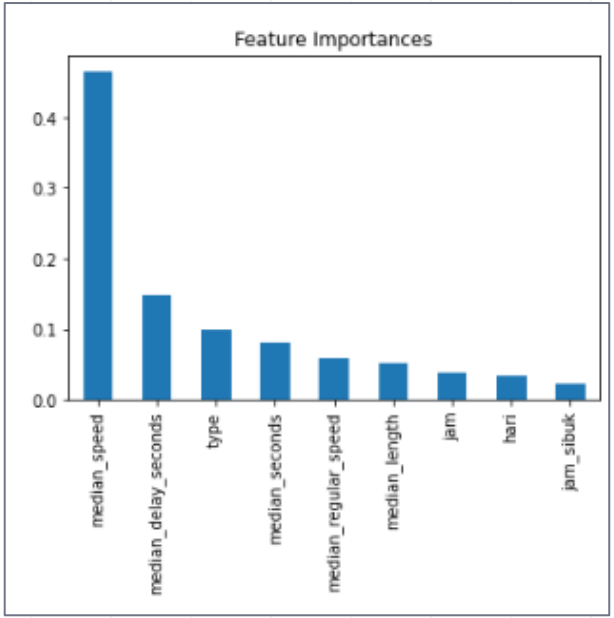
FEATURE ENGINEERING

MODELLING

► EVALUATION

Bandung Traffic Flow Analysis with XGBoost

- Waktu tunggu dan kecepatan berpengaruh penting terhadap tingkat kemacetan di Kota Bandung.
- Tipe kepadatan juga berpengaruh penting dalam memprediksi tingkat kemacetan



Terima Kasih