

# Presentasi TK KASDD - model.unfit()

---

KASDD Genap 2024/2025

# Anggota Kelompok

- Fikri Budianto (2206025306)
- Khansa Mahira (2206819413)
- Dian Fathur Rahman (2206082096)
- Gilang Fajar Pratama (2206082631)

# Outline

- Deskripsi Dataset
- Exploratory Data Analysis
- Preprocessing
- Modelling

# Deskripsi Dataset

---

# Deskripsi Dataset

- Dataset ini berisi informasi mengenai karyawan suatu perusahaan mencakup data demografi karakteristik pekerjaan kepuasan kerja pengalaman kerja dan faktor lain yang berhubungan dengan retensi dan kesejahteraan karyawan. Dataset ini dapat digunakan untuk menganalisis faktor-faktor yang memengaruhi kepuasan kerja serta kinerja karyawan.
- Dataset terdiri dari 9078 baris dan 30 kolom
- Dataset bersifat imbalanced (Attrition No: 7303, Attrition Yes: 1775)

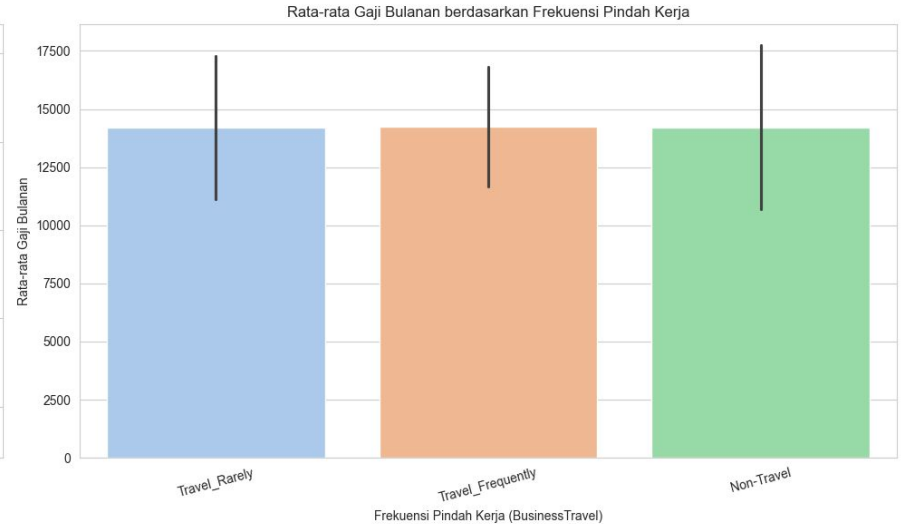
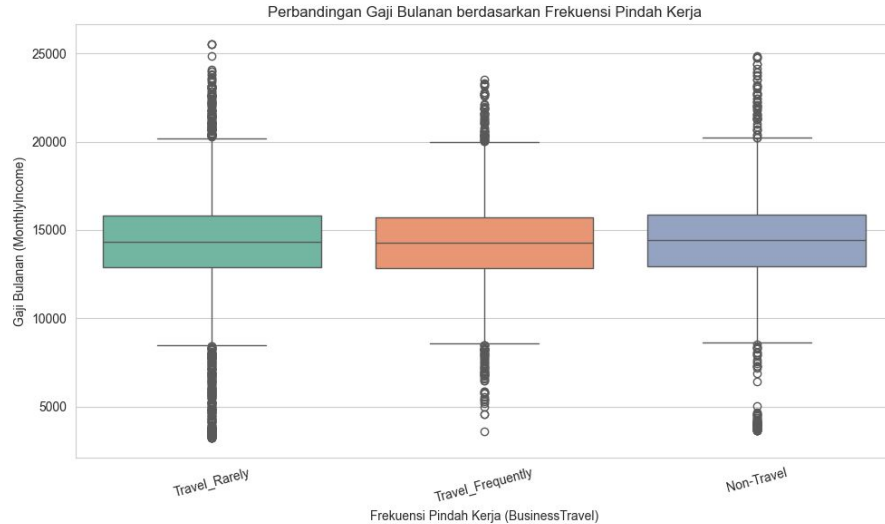
# Deskripsi Dataset

- **Kolom kategorikal** : 'ID', 'BusinessTravel', 'Department', 'Education', 'EducationField', 'EnvironmentSatisfaction', 'Gender', 'JobInvolvement', 'JobRole', 'JobSatisfaction', 'MaritalStatus', 'OverTime', 'PerformanceRating', 'RelationshipSatisfaction', 'WorkLifeBalance', 'Attrition'
- **Kolom numerikal** : 'Age', 'DistanceFromHome', 'JobLevel', 'NumCompaniesWorked', 'PercentSalaryHike', 'StandardHours', 'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear', 'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion', 'YearsWithCurrManager', 'MonthlyIncome'

# Exploratory Data Analysis

---

# Apakah karyawan yang tidak sering pindah kerja mendapatkan gaji lebih tinggi dibandingkan dengan sering berpindah-pindah kerja?

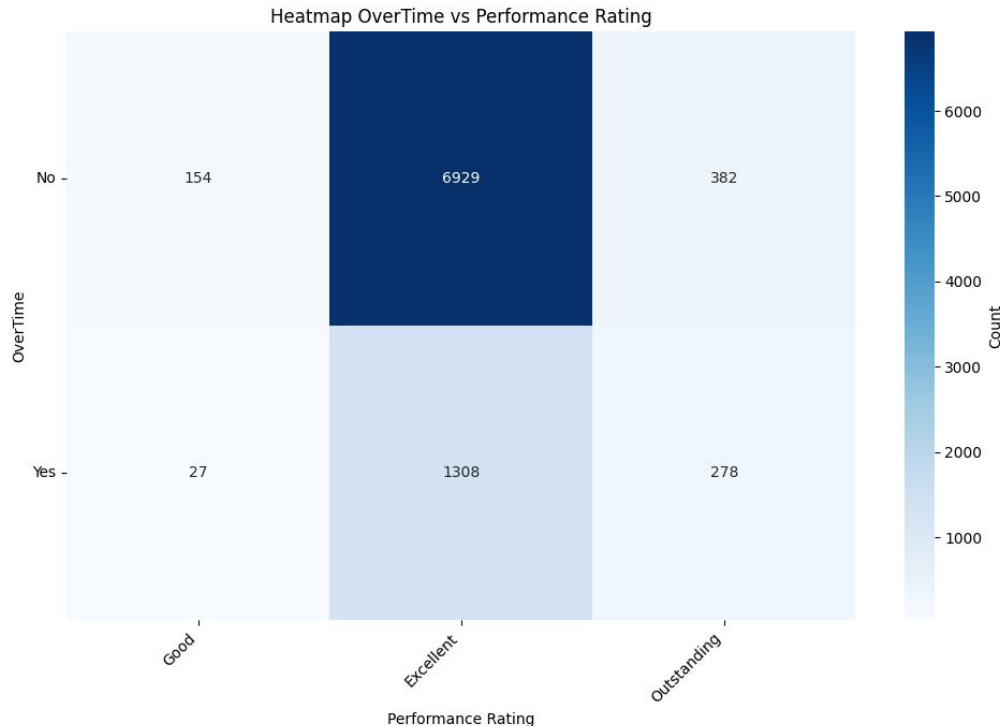


Karyawan dengan kategori Travel\_Rarely cenderung memiliki gaji lebih tinggi secara median dibandingkan dengan kategori Travel\_Frequently.

Dapat disimpulkan karyawan yang tidak sering pindah kerja (Travel\_Rarely atau Non-Travel) cenderung memiliki gaji bulanan yang lebih tinggi dibandingkan yang sering berpindah-pindah kerja (Travel\_Frequently).

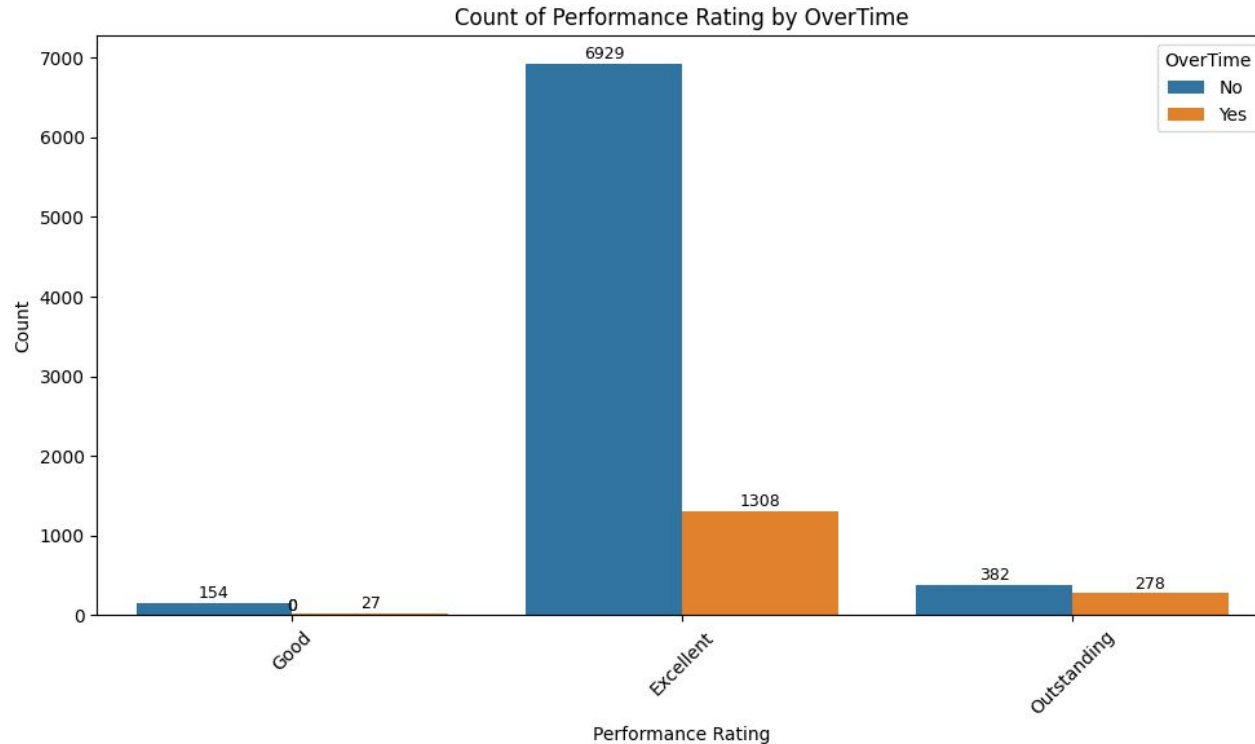


# Apakah karyawan yang bekerja secara overtime memiliki performance yang baik?



Berdasarkan heatmap, terdapat pola bahwa pekerja yang tidak overtime cenderung memiliki performa yang baik saat bekerja.

Apakah karyawan yang bekerja secara overtime memiliki performance yang baik?



## Apa ciri-ciri dari karyawan yang memiliki income tinggi?

- Membentuk dua dataset yaitu high\_income\_df dan other\_income\_df dengan menggunakan Q3 sebagai batasan untuk membentuk sub dataset

```
✓ [21] Q3_monthly_income = df_eda['MonthlyIncome'].quantile(0.75)
0s      print(f"Kuartil Ketiga (Q3) untuk MonthlyIncome: {Q3_monthly_income}\n")
```

```
⇒ Kuartil Ketiga (Q3) untuk MonthlyIncome: 15786.557499999999
```

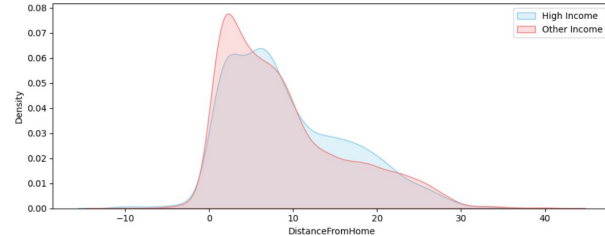
```
✓ [22] high_income_df = df_eda[df_eda['MonthlyIncome'] >= Q3_monthly_income].copy()
0s      other_income_df = df_eda[df_eda['MonthlyIncome'] < Q3_monthly_income].copy()

      print(f"Jumlah karyawan berpenghasilan tinggi (>= Q3): {len(high_income_df)}")
      print(f"Jumlah karyawan berpenghasilan lainnya (kurang dari Q3): {len(other_income_df)}\n")
```

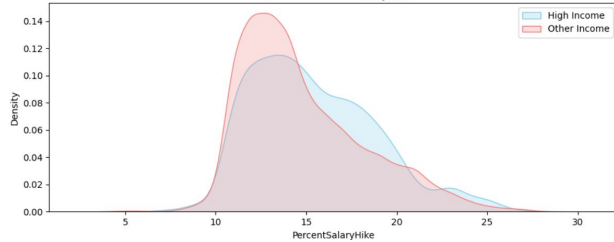
```
⇒ Jumlah karyawan berpenghasilan tinggi (>= Q3): 2270
    Jumlah karyawan berpenghasilan lainnya (kurang dari Q3): 6808
```

# Apa ciri-ciri dari karyawan yang memiliki income tinggi?

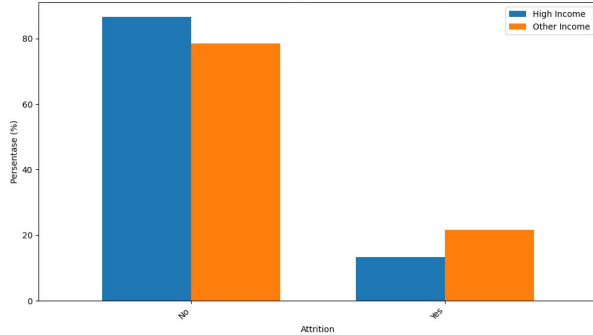
Distribusi: DistanceFromHome



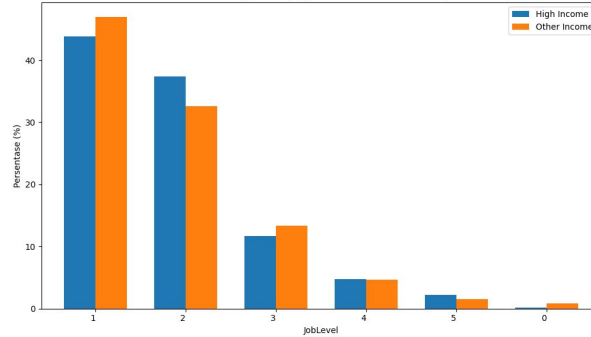
Distribusi: PercentSalaryHike



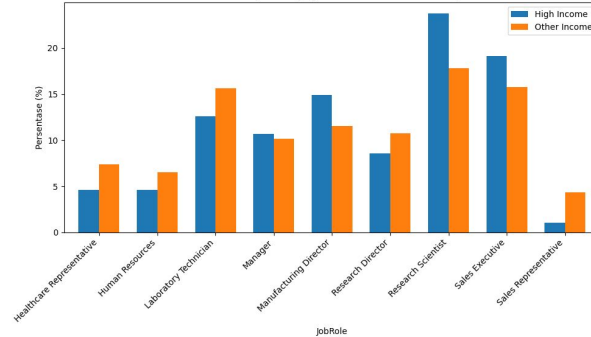
Distribusi: Attrition (High Income vs. Other Income)



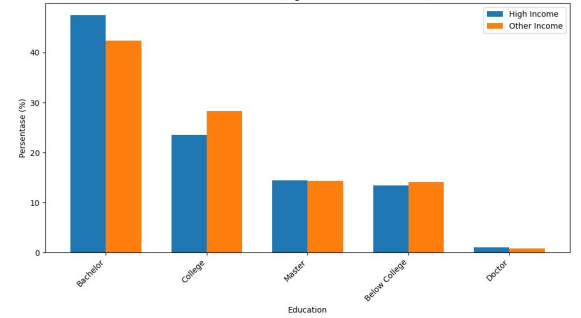
Distribusi: JobLevel (High Income vs. Other Income)



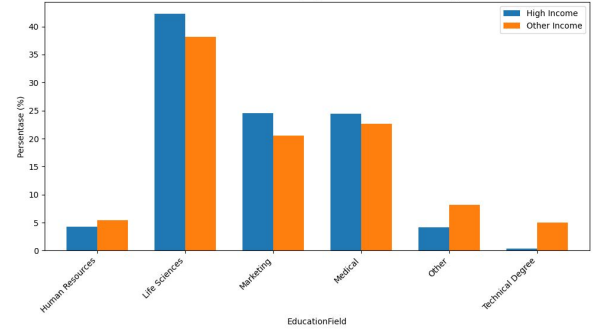
Distribusi: JobRole (High Income vs. Other Income)



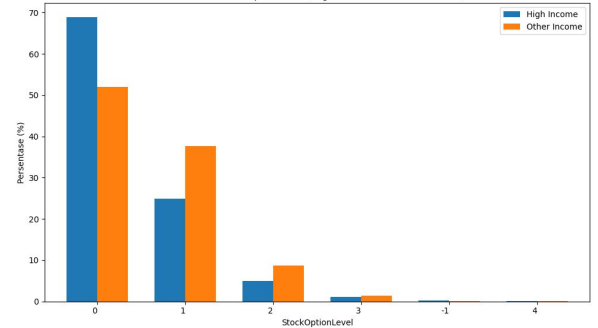
Distribusi: Education (High Income vs. Other Income)



Distribusi: EducationField (High Income vs. Other Income)



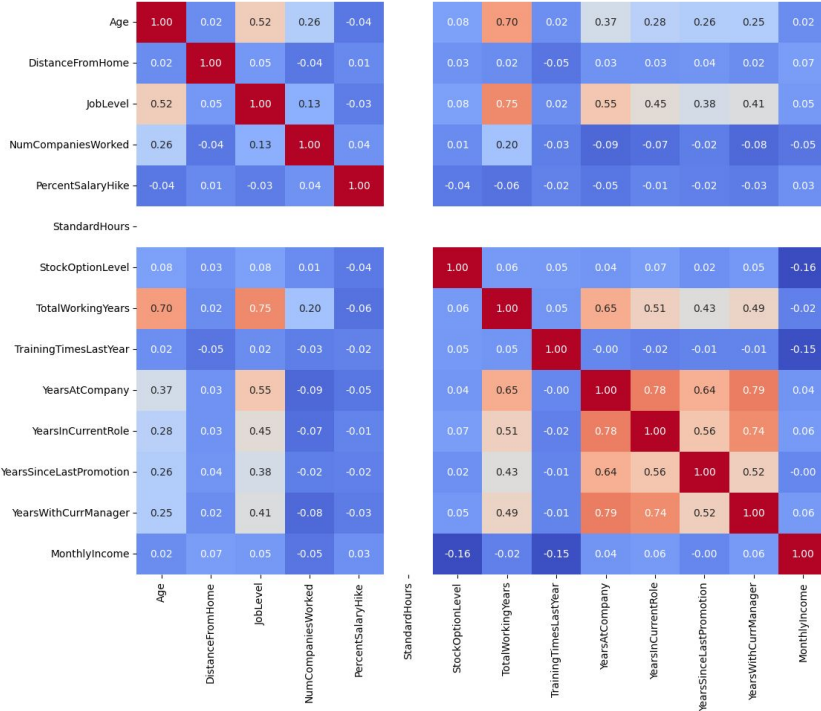
Distribusi: StockOptionLevel (High Income vs. Other Income)



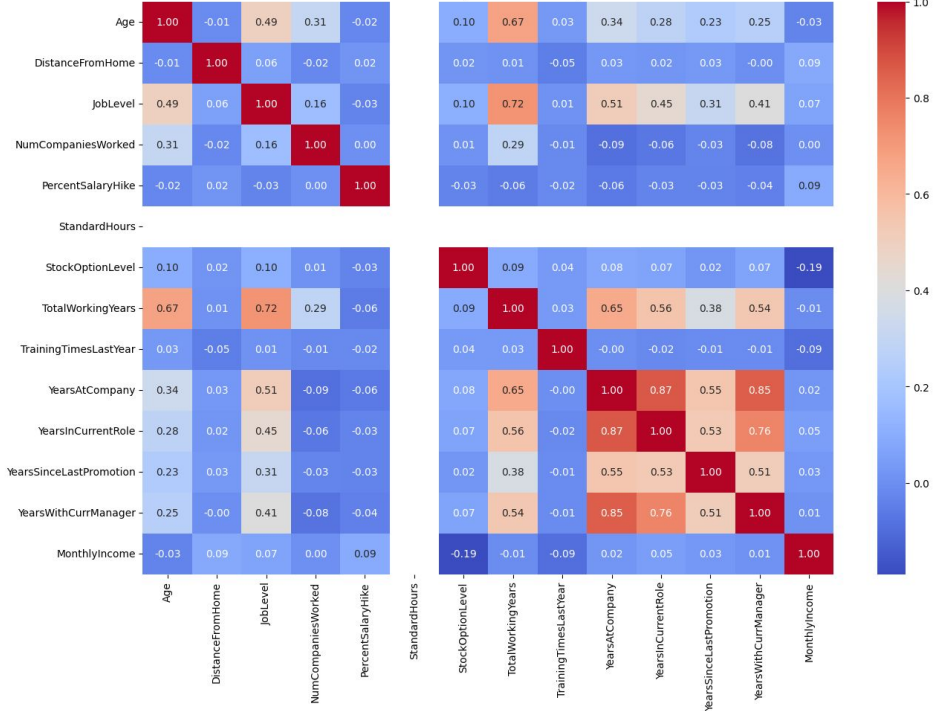
# Apa ciri-ciri dari karyawan yang memiliki income tinggi?

- Korelasi dilakukan dengan data train sebenarnya

Correlation Heatmap of All Features Pearson



Correlation Heatmap of All Features Spearman

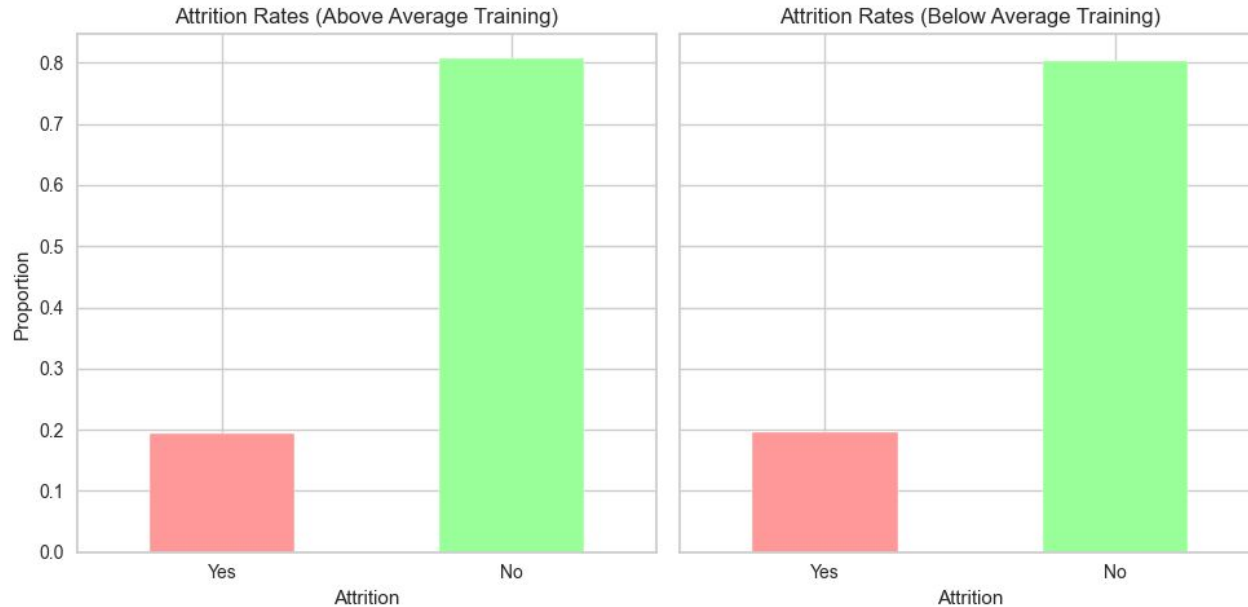


## Apa ciri-ciri dari karyawan yang memiliki income tinggi?

Karyawan yang memiliki pendapatan tinggi cenderung memiliki karakteristik sebagai berikut.

- JobLevel yang tinggi (2, 4, 5)
- Pendidikan tinggi (Bachelor, Master, atau Doctor)
- Persentase kenaikan gaji (PercentSalaryHike) yang lebih tinggi
- Bidang studi (EducationField) umumnya Life Sciences, Marketing, atau Medical
- Umumnya bekerja di departemen Research & Development (R&D) atau Sales
- Cenderung tidak ingin meninggalkan perusahaan (Attrition 'No' tinggi)
- Jarak tempuh dari rumah ke kantor yang lebih jauh
- Opsi saham rendah (StockOptionLevel 0)
- Kepuasan lingkungan kerja (EnvironmentSatisfaction) cenderung tinggi atau sangat tinggi, walaupun WorkLifeBalance dominan pada kategori Good

Apakah karyawan yang memiliki jumlah pelatihan di atas rata-rata cenderung memiliki tingkat pengunduran diri yang lebih tinggi?

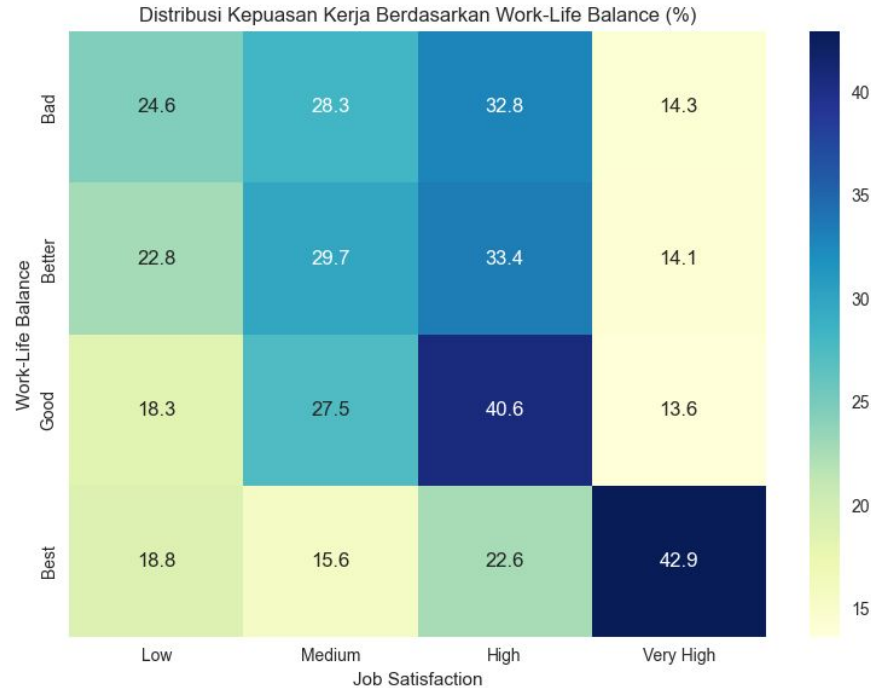


Apakah karyawan yang memiliki jumlah pelatihan di atas rata-rata cenderung memiliki tingkat pengunduran diri yang lebih tinggi?

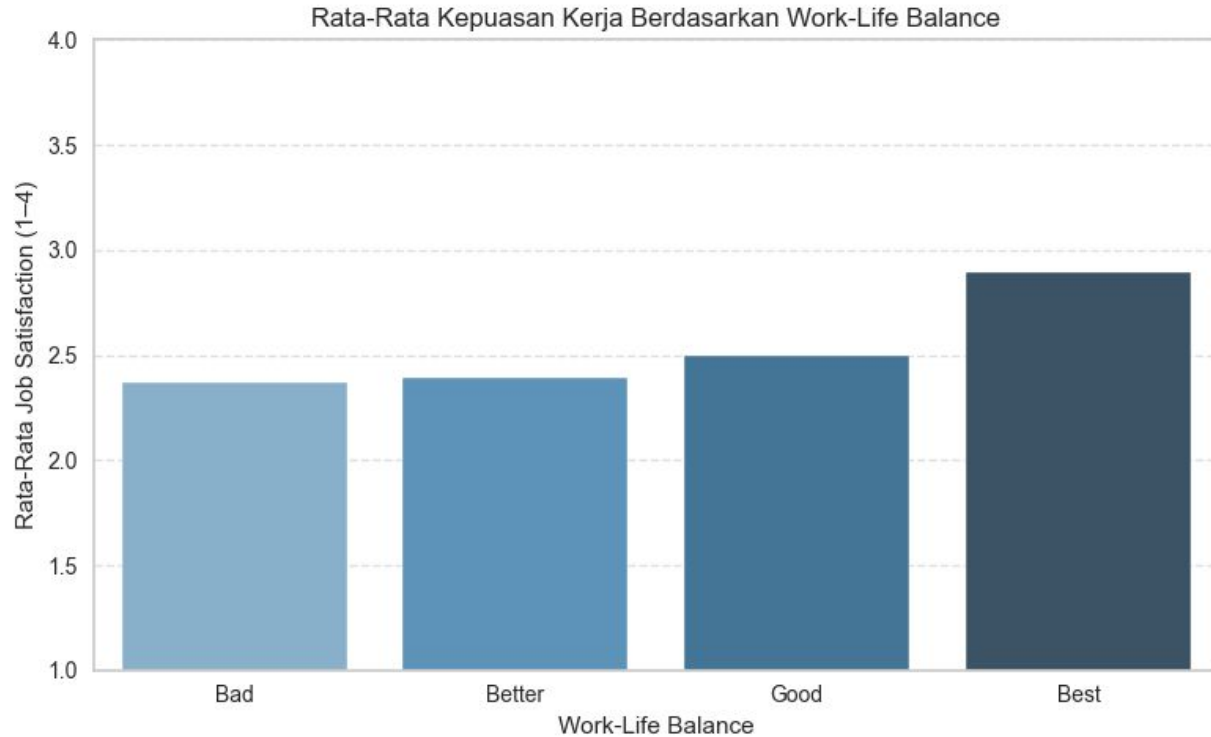
Karyawan dengan jumlah pelatihan di atas rata-rata yang mengundurkan diri proporsinya tidak jauh berbeda dibandingkan karyawan dengan jumlah pelatihan di bawah rata-rata yang mengundurkan diri. Hal ini menunjukkan bahwa dugaan karyawan dengan tingkat pelatihan lebih tinggi memiliki tingkat pengunduran diri yang lebih tinggi tidak benar.



Apakah Work-Life Balance yang baik berkorelasi dengan tingkat kepuasan kerja yang tinggi?



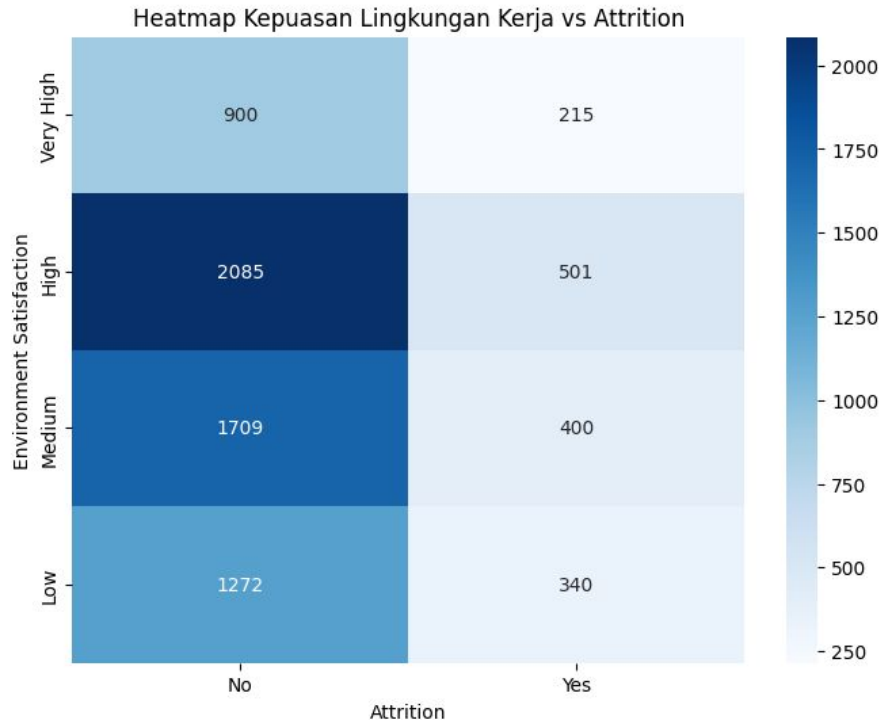
Apakah Work-Life Balance yang baik berkorelasi dengan tingkat kepuasan kerja yang tinggi?



Apakah Work-Life Balance yang baik berkorelasi dengan tingkat kepuasan kerja yang tinggi?

Berdasarkan heatmap, terlihat bahwa semakin tinggi work-life balance seseorang maka semakin tinggi juga tingkat kepuasan kerjanya. Selain itu, berdasarkan bar plot, semakin tinggi work-life balance maka nilai rata-rata kepuasan kerjanya juga meningkat. Kedua visualisasi ini menunjukkan bahwa work-life balance yang baik berkorelasi dengan tingkat kepuasan kerja.

Apakah kepuasan terhadap lingkungan kerja berpengaruh terhadap kemungkinan karyawan bertahan (attrition)?



Terdapat pengaruh bahwa semakin tinggi level kepuasan lingkungan kerja, maka karyawan akan tetap bertahan. Akan tetapi, pengaruh ini hanya sampai pada level kepuasan lingkungan kerja “High”.

# Preprocessing

---

# Preprocessing

Secara umum, langkah-langkah preprocessing yang dilakukan adalah handling missing values, handling duplicate values, handling numerical data dengan scaling, handling categorical data dengan encoding, membuat fitur baru dan feature selection di beberapa anggota kelompok, dan menguji beberapa metode berbeda untuk handle imbalanced data (SMOTE, SVM SMOTE, class\_weights, Tomek Link) untuk task klasifikasi.

# Missing Values & Duplicates

Values yang missing dihandle dengan menggunakan median untuk data numerik dan modus untuk data kategorikal.

```
▶ # Check for missing values in all columns
missing_values_all = df_classif.isnull().sum().sort_values(ascending=False)

print("🔍 Missing Values (All Columns):\n")
print(missing_values_all)
```

[274]

```
... 🔍 Missing Values (All Columns):
```

Education	1668
EnvironmentSatisfaction	1656
Department	1509
RelationshipSatisfaction	1295
WorkLifeBalance	1141
YearsAtCompany	1110
EducationField	1101
TotalWorkingYears	1046
PercentSalaryHike	572
Gender	543
YearsWithCurrManager	540
YearsSinceLastPromotion	532
JobSatisfaction	195
JobInvolvement	31
JobRole	0
MaritalStatus	0

# Handling Duplicates

Terdapat juga uji coba handling duplicates setelah melakukan drop ID

Menghilangkan data duplikat menyebabkan performa lebih buruk di klasifikasi (skor public f1 Kaggle 0,75913 turun ke 0,72926) tetapi terdapat peningkatan di regresi (R2 dari 0.76935 ke 0.77438 )

```
✓ [65] df_with_outliers_encoded = df_with_outliers_encoded.drop_duplicates()  
0s df_with_outliers_encoded.duplicated().sum()
```

⇒ np.int64(0)

```
✓ [66] df_outliers_clean_encoded.duplicated().sum()  
0s
```

⇒ np.int64(1405)

```
✓ [67] df_outliers_clean_encoded = df_with_outliers_encoded.drop_duplicates()  
0s df_outliers_clean_encoded.duplicated().sum()
```

⇒ np.int64(0)



# Handling numerical data

Scaling yang digunakan masing-masing anggota berbeda untuk menguji mana yang lebih baik. Scaling yang diuji oleh masing-masing anggota adalah Min-Max Scaling, Standard Scaler, dan Log Transformation.

Metode yang paling baik adalah menerapkan transformasi yang berbeda untuk kolom yang berbeda tergantung pada skewness kolom tersebut. Hal ini karena kedua metode ini mengurangi skewness distribusi data menjadi lebih normal. Semua model termasuk tree-based model mengalami peningkatan performa karena handling seperti ini.

# Handling numerical data

```
# Apply scaling/normalization based on column-specific methods
def apply_scaling(df, numerical_cols):
    df_copy = df.copy()

    # Columns for Log1p + StandardScaler
    log_then_standard_cols = [
        'DistanceFromHome', 'NumCompaniesWorked', 'TotalWorkingYears',
        'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion', 'YearsWithCurrManager'
    ]
    df_copy = log_transform(df_copy, [col for col in log_then_standard_cols if col in numerical_cols])
    df_copy = standard_scale(df_copy, [col for col in log_then_standard_cols if col in numerical_cols])

    # Columns for StandardScaler only
    standard_cols = ['Age', 'PercentSalaryHike', 'MonthlyIncome']
    df_copy = standard_scale(df_copy, [col for col in standard_cols if col in numerical_cols])

    # Columns for MinMaxScaler
    minmax_cols = ['TrainingTimesLastYear']
    df_copy = minmax_scale(df_copy, [col for col in minmax_cols if col in numerical_cols])

    return df_copy
```

# Handling categorical data

Diterapkan label encoding untuk data ordinal untuk mempertahankan urutan dan one-hot encoding untuk data nominal. Untuk label dilakukan 0/1 encoding.

```
def encode_categorical(df, ordinal_cols, non_ordinal_cols, mappings=None):
    """
    Apply label encoding to ordinal columns and one-hot encoding to non-ordinal columns.

    Parameters:
    df (pd.DataFrame): Input DataFrame
    ordinal_cols (list): List of ordinal column names
    non_ordinal_cols (list): List of non-ordinal column names
    mappings (dict): Optional dictionary of column names to ordered value lists

    Returns:
    pd.DataFrame: DataFrame with encoded columns
    dict: Dictionary of all encoders used
    """
    df_encoded = df.copy()
    all_encoders = {}

    # Apply label encoding to ordinal columns
    if ordinal_cols:
        df_encoded, label_encoders = label_encode(df_encoded, ordinal_cols, mappings)
        all_encoders.update(label_encoders)

    # Apply one-hot encoding to non-ordinal columns
    if non_ordinal_cols:
        df_encoded, ohe_encoders = one_hot_encode(df_encoded, non_ordinal_cols)
        all_encoders.update(ohe_encoders)

    return df_encoded, all_encoders
```

Do you want to install the recommended 'Rainbow CSV' extension from mechatroner for

# Handling categorical data

Diterapkan label encoding untuk data ordinal untuk mempertahankan urutan dan one-hot encoding untuk data nominal.

```
✓ [57] # One-Hot Encoding
0s # List of categorical nominal columns to encode
nominal_columns = [
    "Department", "EducationField", "Gender", "JobRole", "MaritalStatus"
]
```

```
✓ [60] # Label Encoding
0s # Mapping untuk kolom BusinessTravel
BusinessTravel_mapping = {
    'Non-Travel': 1,
    'Travel_Rarely': 2,
    'Travel_Frequently': 3
}

# Mapping untuk kolom Education
Education_mapping = {
    'Below College': 1,
    'College': 2,
    'Bachelor': 3,
    'Master': 4,
    'Doctor': 5
}

# Mapping untuk kolom EnvironmentSatisfaction
EnvironmentSatisfaction_mapping = {
    'Low': 1,
    'Medium': 2,
    'High': 3,
    'Very High': 4
}
```

```
# Mapping untuk kolom RelationshipSatisfaction
RelationshipSatisfaction_mapping = {
    'Low': 1,
    'Medium': 2,
    'High': 3,
    'Very High': 4
}

# Mapping untuk kolom WorkLifeBalance
WorkLifeBalance_mapping = {
    'Bad': 1,
    'Good': 2,
    'Better': 3,
    'Best': 4
}

# Mapping untuk kolom Attrition
Attrition_mapping = {
    'No': 0,
    'Yes': 1
}
```

```
# Mapping untuk kolom JobInvolvement
JobInvolvement_mapping = {
    'Low': 1,
    'Medium': 2,
    'High': 3,
    'Very High': 4
}

# Mapping untuk kolom JobSatisfaction
JobSatisfaction_mapping = {
    'Low': 1,
    'Medium': 2,
    'High': 3,
    'Very High': 4
}

# Mapping untuk kolom OverTime
OverTime_mapping = {
    'No': 0,
    'Yes': 1
}

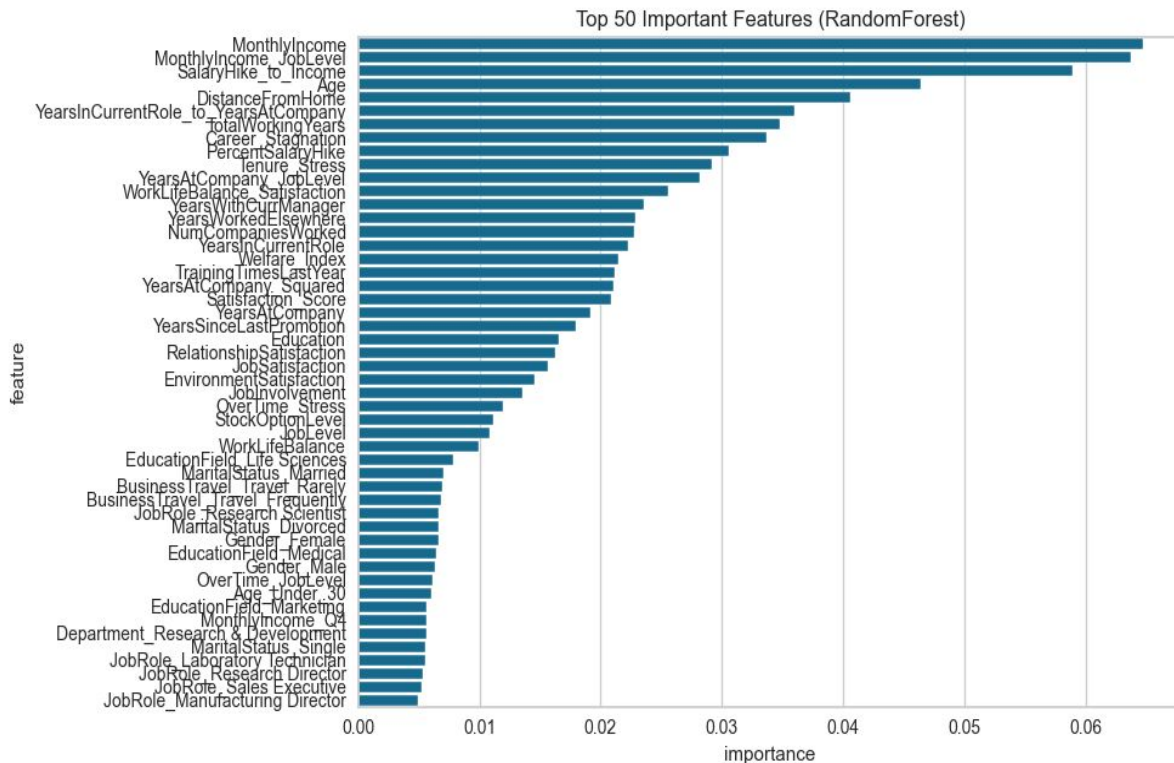
# Mapping untuk kolom PerformanceRating
PerformanceRating_mapping = {
    'Good': 1,
    'Excellent': 2,
    'Outstanding': 3
}
```

# Modelling

---

# Classification

Dipilih top 35, hal ini karena kalau dipilih semua ternyata terlalu banyak noise yang ditangkap sehingga menyebabkan ada misclassification/class noise dan juga untuk mengurangi resiko overfitting.



# Classification

Dengan menggunakan RandomizedSearchCV, scale\_pos\_weight=30 untuk XGBoost, weights = {0: 1, 1: 30} pada model lain, dan StratifiedKFold dengan K=10 diperoleh hasil sebagai berikut:





Model Performance Summary (Sorted by F1 Weighted):

	best_threshold	f1_weighted	f1_micro	f1_macro	precision	recall	accuracy	roc_auc
XGBoost	0.80	0.819759	0.841335	0.692888	0.717172	0.360000	0.841335	0.763866
BalancedRandomForest	0.70	0.819035	0.843736	0.688154	0.756927	0.338592	0.843736	0.764715
KNN	0.70	0.816789	0.846708	0.678793	0.834877	0.304789	0.846708	0.741365
ExtraTrees	0.85	0.816748	0.851052	0.673986	0.948669	0.281127	0.851052	0.741627
CatBoost	0.85	0.816730	0.840078	0.686092	0.722222	0.344225	0.840078	0.737754
RandomForest	0.60	0.816490	0.839963	0.685571	0.722420	0.343099	0.839963	0.728658
LightGBM	0.85	0.815955	0.838706	0.685570	0.709677	0.347042	0.838706	0.756879
GradientBoosting	0.55	0.813036	0.834476	0.682535	0.677524	0.351549	0.834476	0.740709
EasyEnsemble	0.60	0.724238	0.768176	0.517740	0.310912	0.117183	0.768176	0.593980
RidgeClassifier	0.60	0.720519	0.755258	0.522300	0.288194	0.140282	0.755258	0.565099
GaussianNB	0.35	0.707099	0.797096	0.443547	0.000000	0.000000	0.797096	0.495674
LogisticRegression	0.85	0.638018	0.608253	0.499010	0.213989	0.348169	0.608253	0.514137

C:\Users\Fikri Budianto\AppData\Roaming\Python\Python312\site-packages\xgboost\training.py:183: UserWarning: [07:41:38] WARNING: Parameters: { "use\_label\_encoder" } are not used.

# Classification

## KNN

	<b>knn_clf_smote_handle_outliers.csv</b> Complete · Khansa Mahira · 6h ago	<b>0.72980</b>	<input type="checkbox"/>
	<b>knn_clf_smote_no_handle_outliers.csv</b> Complete · Khansa Mahira · 6h ago	<b>0.72573</b>	<input type="checkbox"/>
	<b>submission_clf_smote_handle_outliers (1).csv</b> Complete · Khansa Mahira · 9h ago	<b>0.71528</b>	<input type="checkbox"/>
	<b>submission_clf_smote_handle_outliers.csv</b> Complete · Khansa Mahira · 10h ago	<b>0.70826</b>	<input type="checkbox"/>



# Classification

## GradientBoost



**class\_with\_outlier.csv**

Complete · FajarTama · 4h ago

**0.72311**



**smote\_gradient\_without\_outlier.csv**

Complete · FajarTama · 5h ago

**0.73832**



# Classification

## Random Forest



**hasil\_clf\_dupli\_outlier\_minmax\_adasyn.csv**

Complete · Dian Fathur Rahman · 1h ago

**0.72980**



**hasil\_klasifikasi\_RF.csv**





Complete · Dian Fathur Rahman · 2d ago

**0.73460**



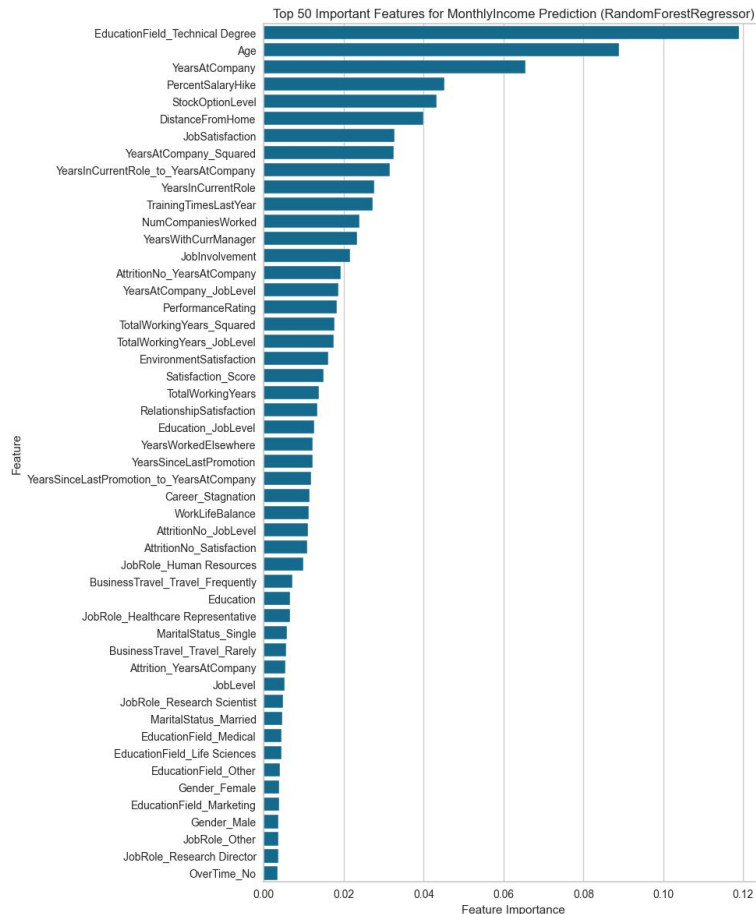
# Classification

Pengujian di Kaggle dengan berbagai macam preprocessing menghasilkan skor f1 sebagai berikut:

	<b>attrition_predictions_top1.csv</b> Complete · Fikri Budianto · 1d ago	<b>0.75913</b>	<input checked="" type="checkbox"/>
	<b>employee_test_classif_pred.csv</b> Complete · FajarTama · 10h ago	<b>0.75255</b>	<input type="checkbox"/>
	<b>submission_classification (2).csv</b> Complete · Khansa Mahira · 24d ago	<b>0.73913</b>	<input type="checkbox"/>
	<b>hasil_klasifikasi_RF.csv</b> Complete · Dian Fathur Rahman · 2d ago	<b>0.73460</b>	<input type="checkbox"/>

# Regression

Dipilih top 35, hal ini karena kalau dipilih semua ternyata terlalu banyak noise yang ditangkap sehingga menyebabkan prediksi tidak akurat dan juga untuk mengurangi resiko overfitting.



# Regression





Dengan menggunakan RandomizedSearchCV dan KFold dengan k=5 diperoleh hasil sebagai berikut:

```
warnings.warn(
```

Regression Model Performance:					
	r2		mae	mse	rmse
XGBoost	0.741351	1029.483380	2.271241e+06	1507.063706	
ExtraTrees	0.714220	1083.025093	2.509483e+06	1584.134904	
LightGBM	0.710333	1100.878121	2.543619e+06	1594.872817	
RandomForest	0.696181	1119.286940	2.667894e+06	1633.368968	
CatBoost	0.685809	1151.664400	2.758970e+06	1661.014604	
GradientBoosting	0.683735	1162.568394	2.777184e+06	1666.488444	
KNN	0.484401	1540.039719	4.527574e+06	2127.809644	
MLPRegressor	0.471174	1579.542233	4.643717e+06	2154.928515	
Lasso	0.214013	1917.252467	6.901899e+06	2627.146563	
ElasticNet	0.213927	1917.409501	6.902656e+06	2627.290672	
Ridge	0.213923	1917.519441	6.902689e+06	2627.296908	
LinearRegression	0.213914	1917.582335	6.902765e+06	2627.311336	





# Regression

Pengujian di Kaggle dengan berbagai macam preprocessing menghasilkan skor  $r^2$  sebagai berikut:

	<b>regress_pred_gradient.csv</b> Complete · FajarTama · 1h ago	<b>0.77449</b>	<input type="checkbox"/>
	<b>voting_ensemble_prediction.csv</b> Complete · Fikri Budianto · 8m ago	<b>0.77438</b>	<input checked="" type="checkbox"/>
	<b>hasil_rgr.csv</b> Complete · Dian <b>Fathur</b> Rahman · 27s ago	<b>0.74698</b>	<input type="checkbox"/>
	<b>submission_regression (5).csv</b> Complete · Khansa Mahira · 1d ago	<b>0.61566</b>	<input type="checkbox"/>

# Regression

## KNN

	<b>knn_regression_handle_outliers.csv</b> Complete · Khansa Mahira · 6h ago	<b>-0.01274</b>	<input type="checkbox"/>
	<b>knn_regression_no_handle_outliers.csv</b> Complete · Khansa Mahira · 6h ago	<b>0.61566</b>	<input type="checkbox"/>
	<b>submission_regression (8).csv</b> Complete · Khansa Mahira · 10h ago	<b>0.27770</b>	<input type="checkbox"/>
	<b>submission_regression (4).csv</b> Complete · Khansa Mahira · 1d ago	<b>0.57977</b>	<input type="checkbox"/>

# Regression

## GradientBoost



**regress\_pred\_gradient.csv**

Complete · FajarTama · 6m ago

**0.77449**



**regress\_pred\_gradient.csv**

Complete · FajarTama · 1h ago

**0.65520**





# Regression

## Random Forest



**hasil\_rgr\_dupli\_outlier.csv**

Complete · Dian Fathur Rahman · 1h ago

**0.73351**



**hasil\_rgr.csv**

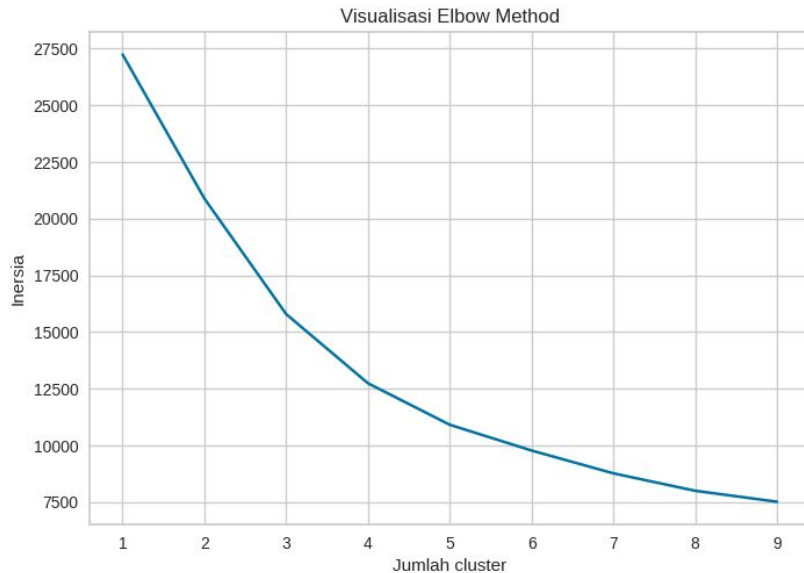
Complete · Dian **Fathur** Rahman · 27s ago

**0.74698**



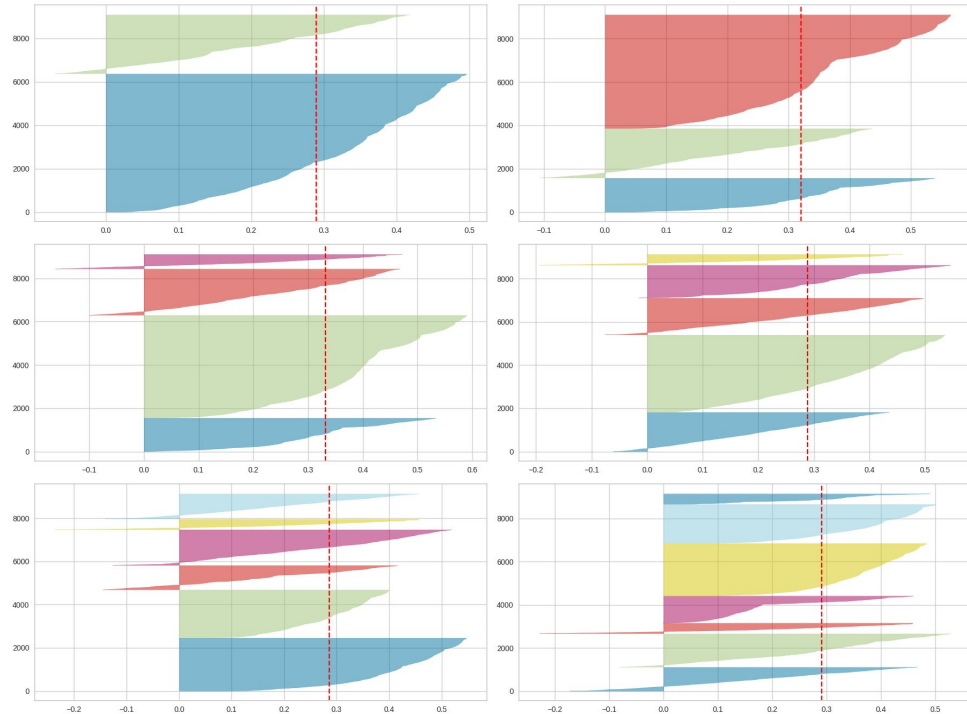
# Clustering

- Fitur yang dipilih: MonthlyIncome, PercentSalaryHike, JobLevel
- Model: K-Means



```
... Compare nilai silhouette coefficient
=====
Untuk k = 2, rata-rata silhouette_coefficient adalah: 0.2893
Untuk k = 3, rata-rata silhouette_coefficient adalah: 0.3206
Untuk k = 4, rata-rata silhouette_coefficient adalah: 0.3315
Untuk k = 5, rata-rata silhouette_coefficient adalah: 0.2887
Untuk k = 6, rata-rata silhouette_coefficient adalah: 0.2856
Untuk k = 7, rata-rata silhouette_coefficient adalah: 0.2907
Untuk k = 8, rata-rata silhouette_coefficient adalah: 0.2988
Untuk k = 9, rata-rata silhouette_coefficient adalah: 0.2803
```

Visualisasi Silhouette Coefficient untuk beberapa nilai k



# Clustering

- $k_{\text{optimal}} = 3$

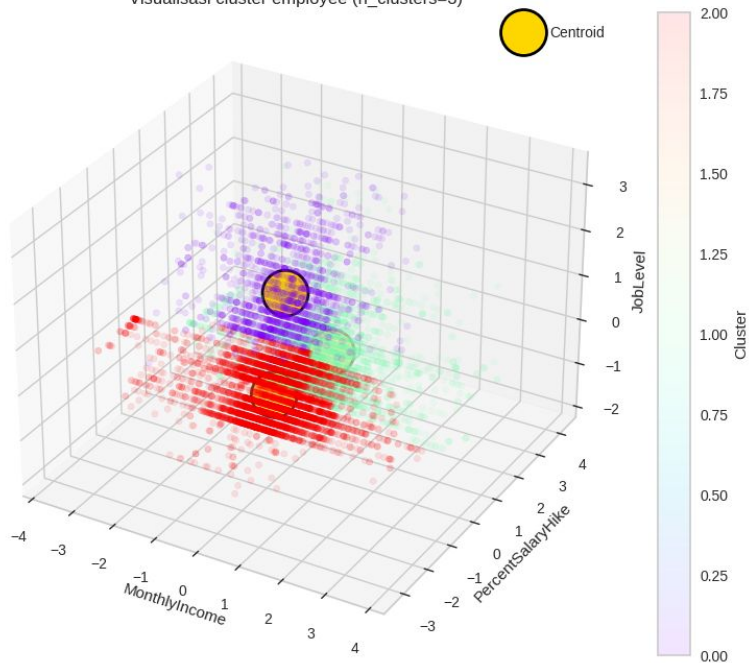
```
# Posisi Centroid
print("Posisi centroid: ")
kmeans.cluster_centers_

[252]

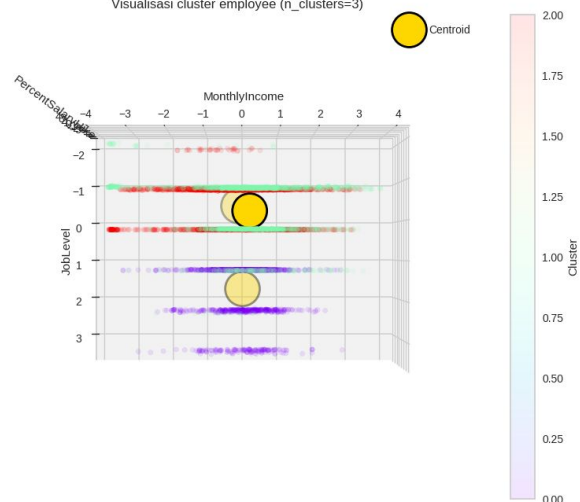
... Posisi centroid:

... array([[ 0.01430524, -0.23231313,  1.73356016],
           [ 0.21071019,  1.34587176, -0.25566358],
           [-0.09566955, -0.51462023, -0.40565471]])
```

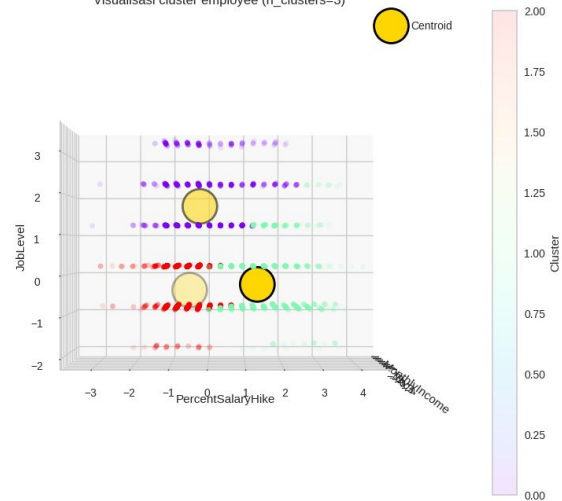
Visualisasi cluster employee (n\_clusters=3)



Visualisasi cluster employee (n\_clusters=3)

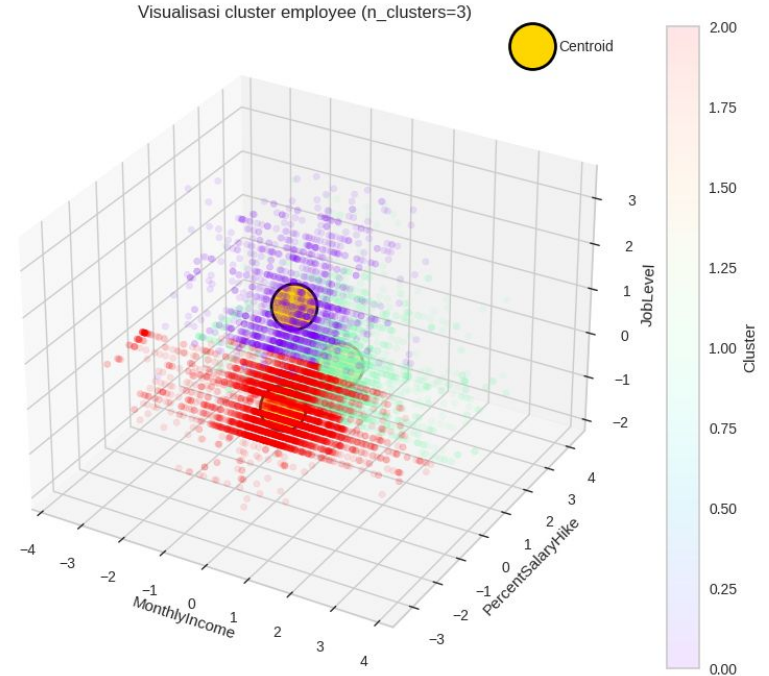


Visualisasi cluster employee (n\_clusters=3)



# Clustering

- Cluster merah  
JobLevel rendah, PercentSalaryHike rendah, dan MonthlyIncome yang relatif kecil.  
Karyawan entry-level atau posisi junior yang belum lama bekerja atau kurang berprestasi dalam kenaikan gaji.
- Cluster hijau  
MonthlyIncome yang sedikit lebih tinggi  
JobLevel yang relatif rendah ke menengah  
PercentSalaryHike yang tinggi.  
Karyawan dengankinerja yang sangat baik atau keahlian khusus di bidang yang sangat dibutuhkan perusahaan.
- Cluster ungu  
JobLevel yang tinggi  
MonthlyIncome dan PercentSalaryHike yang moderat  
(lebih tinggi dari cluster merah tetapi lebih rendah dari cluster hijau).  
Karyawan senior atau manajerial dengan level pekerjaan yang tinggi tetapi tidak mengalami kenaikan gaji cepat



Terima Kasih

---