Improving Employee Retention by Predicting Employee Attrition Using Machine Learning

Supported by:
Rakamin Academy
Career Acceleration School
www.rakamin.com

Rakamin Academy

Created by:
Haolia
haoliaaul@gmail.com
https://www.linkedin.com/in/haolia/

"Geophysical engineering graduates who diligently want to fulfil a role where intellectual, integrity, and curiosity are highly valued. Motivated, able to research, design, implement new features and learn various software. Skill handling problems with unique ways to develop innovative solutions. Proficient using Python, SQL, Tableau and other statistical tools for data multi purposes. Looking for opportunities in data analyst, data science, data engineer and Business Intelligence. "

Exploratory Data Analysis

```
df.describe()
```

✓ 0.8s Python

	EnterpriseID	SkorSurveyEngagement	SkorKepuasanPegawai	JumlahKeikutsertaanProjek	JumlahKeterlambatanSebulanTerakhir	JumlahKetidakhadiran
count	287.000000	287.000000	287.000000	287.000000	287.000000	287.000000
mean	105923.324042	3.101045	3.888502	1.167247	0.411150	10.439024
std	4044.977599	0.836388	0.913060	2.285537	1.273018	6.829769
min	100282.000000	1.000000	1.000000	0.000000	0.000000	1.000000
25%	101269.000000	3.000000	3.000000	0.000000	0.000000	5.000000
50%	106069.000000	3.000000	4.000000	0.000000	0.000000	10.000000
75%	110514.500000	4.000000	5.000000	0.000000	0.000000	15.000000
max	111703.000000	5.000000	5.000000	8.000000	6.000000	55.000000

Data Cleansing

```python
# Feature Cleansing
df["keikutsertaanproject_boolean"] = df["JumlahKeikutsertaanProjek"].map(lambda x: 1 if x!=0 else 0)
df["isResign"] = df["isResign"].map(lambda x: 1 if x == 1 else 0)

# Date Time Feature
df["lama_bekerja"] = df["Tahun_Resign"].map(lambda x: 0 if x == "-" else x).astype(int) - df["Tahun_Hiring"].astype(int)
df["lama_bekerja"] = df["lama_bekerja"].map(lambda x: 0 if x < 0 else x)
df["usia_hired"] = df["Tahun_Hiring"].astype(int) - df["TanggalLahir"].map(lambda x: int(x[:4])).astype(int)
df["jarak_penilaian_tahun"] = df["TanggalPenilaianKaryawan"].map(lambda x: int(x[:4])).astype(int) - df["Tahun_Hiring"].astype(int)

df[["usia_hired","lama_bekerja","jarak_penilaian_tahun"]].describe()
```

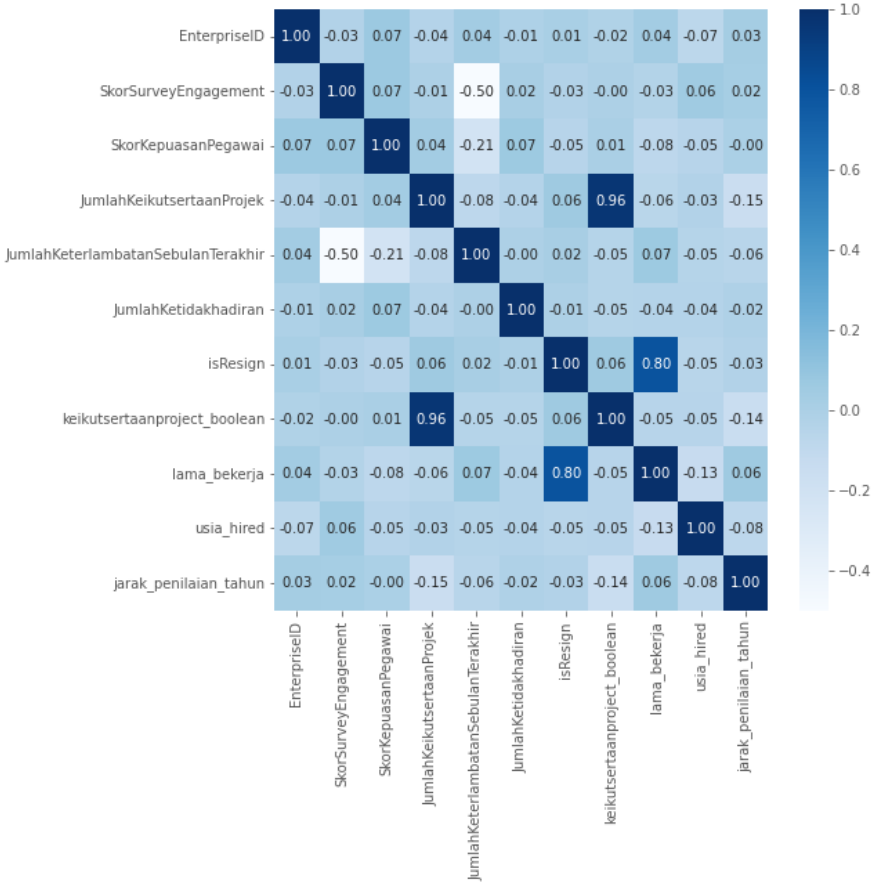
Python

	usia_hired	lama_bekerja	jarak_penilaian_tahun
count	287.000000	287.000000	287.000000
mean	34.080139	1.254355	5.933798
std	8.854922	2.353715	2.339791
min	19.000000	0.000000	1.000000
25%	28.000000	0.000000	4.000000
50%	32.000000	0.000000	6.000000
75%	39.000000	1.000000	7.000000
max	63.000000	9.000000	14.000000

Feature selection

Berdasarkan nilai korelasi ((corr memiliki nilai +/-) >= 0.05) dan agar hasil tidak diskriminikatif maka feature yang akan digunakan yaitu:

- StatusKepegawaian
- Pekerjaan
- JenjangKarir
- PerformancePegawai
- HiringPlatform
- SkorKepuasanPegawai
- JumlahKeikutsertaanProjek
- TingkatPendidikan
- AlasanResign
- umur
- Lamakerja
- is_resign



Feature engineering

Feature engineering

```python
# diubah agar menjadi group yang lebih sedikit
df_use['DivisionPekerjaan'] = np.select([(df_use['Pekerjaan'].str.contains('Software')),
                                         (df_use['Pekerjaan'].str.contains('Data') | df_use['Pekerjaan'].str.contains('Machine'))],
                                         ['Software division','Data division'],'Product division')

df_use['GroupPlatform'] = np.select([(df_use['HiringPlatform'] == 'Indeed'),
                                     (df_use['HiringPlatform'] == 'LinkedIn')],
                                    ['Indeed','LinkedIn'],'Others')

df_use['GroupAlasan'] = np.select([(df_use['AlasanResign'] == 'masih_bekerja'),
                                   (df_use['AlasanResign'].str.contains('karir'))],
                                  ['masih_bekerja','masalah_karir'],'masalah_kenyamanan')
```

Python

Modelling Machine Learning

Split data train dan testing

Modelling yang akan dilakukan pada projek ini yaitu modeling klasifikasi dengan split data test yaitu 30:70

Modelling

Modeling Klasifikasi yang dilakukan pada projek ini menggunakan algoritma K-Nearest Neighbor, Logistic Regression, Decision Tree, Random Forest, dan Gradient Boosting, yang hasil dari Accuracy dan Precision terbaik yaitu menggunakan algoritma Logistic Regresion yaitu Acc = 98,84 % dan Prec = 98,33 %

Hyperparameter tuning

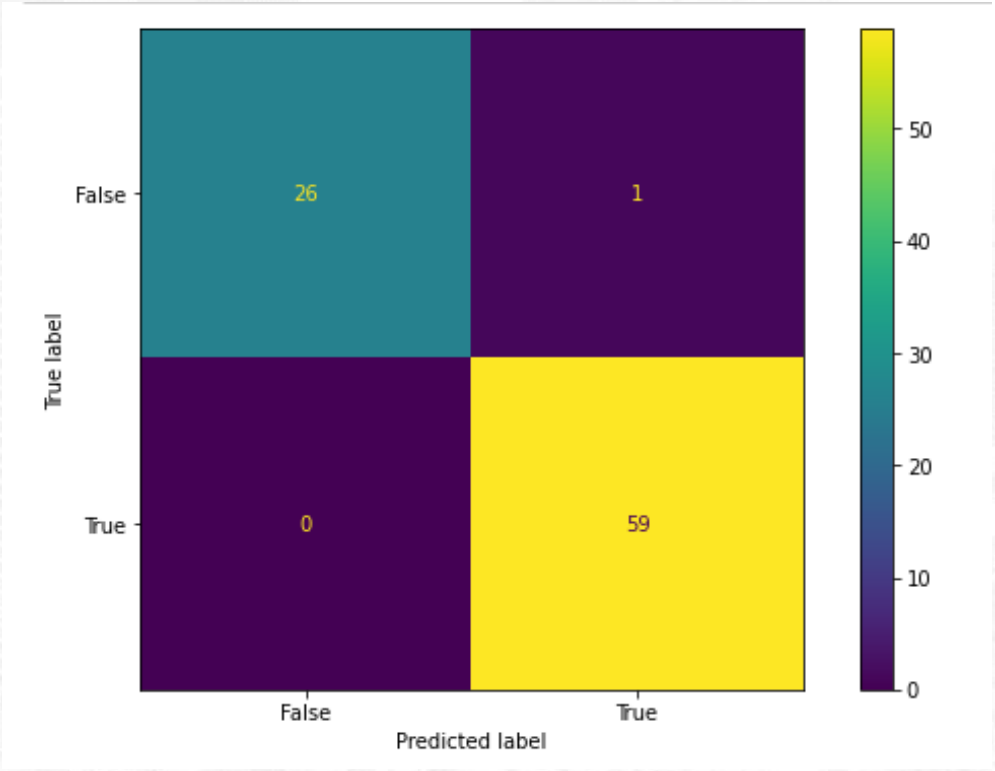
Dicoba hyperparameter tuning pada algoritma terbaik yaitu Logistic regresion tetapi hasilnya turun untuk Acc = 71 % dan Prec = 71% yang berarti cara yang terbaik model yaitu tanpa menggunakan Hyperparameter

Evaluation Model

Confusion Matrix

Jadi hasil akurasi dari model menggunakan algoritma Logistic Regression yaitu :
Accurasi = TT + FF / n_data = 59 + 26 /86 = 98,84%

dari hasil tersebut dapat disimpulkkan bahwa model tersebut dapat mendukung perusahaan untuk mengklasifikasi Karyawan yang kemungkinan resign



Evaluation Model

Feature Importance

Dapat dilihat pada grafik feature Importance bahwa fitur-fitur yang paling berpengaruh 7 terbesar pada model yaitu:

1. GroupAlasan_masih_bekerja
2. GroupAlasan_masalah_kenyamanan
3. LamaBekerja
4. GroupAlasan_masalah_karir
5. Umur
6. Performance_mapped
7. Kepegawaian_mapped