Tugas Kelompok KASDD

RESIGNMENT INTENTION

BE AI

BE AI

ANGGOTA

Adella Rakha Amadea - 2006463616
Mohammad Bryan Mahdavikhia - 2006463124
Izzan Nufail Arvin - 2006462922
Enzo Hering Fahrezzy - 2006596850

OVERVIEW DATA

Dataset Resignation Intention merupakan dataset dengan dimensi 30 kolom x 1470 baris yang mendeskripsikan mengenai karyawan dan apakah suatu karyawan (employee) akan resign atau tidak.

Dataset ini dapat mengeksplor tentang karakteristik data karyawan yang resign pada perusahaan ini dan banyak eksplorasi data yang bisa kami eksplor, seperti karyawan mana yang loyal. Selain itu, kami juga bisa memprediksi karyawan yang akan resign dan berapa lama karyawan akan bertahan pada perusahaan ini melalui data tersebut.

PROGRES KELOMPOK

SOAL NOMOR 1

A

Visualisasikan karakteristik karyawan yang resign dari perusahaan tersebut!

B

Apakah karyawan memilih untuk resign setelah mendapatkan promosi?

C

Departemen manakah yang memiliki karyawan loyal paling banyak?

D

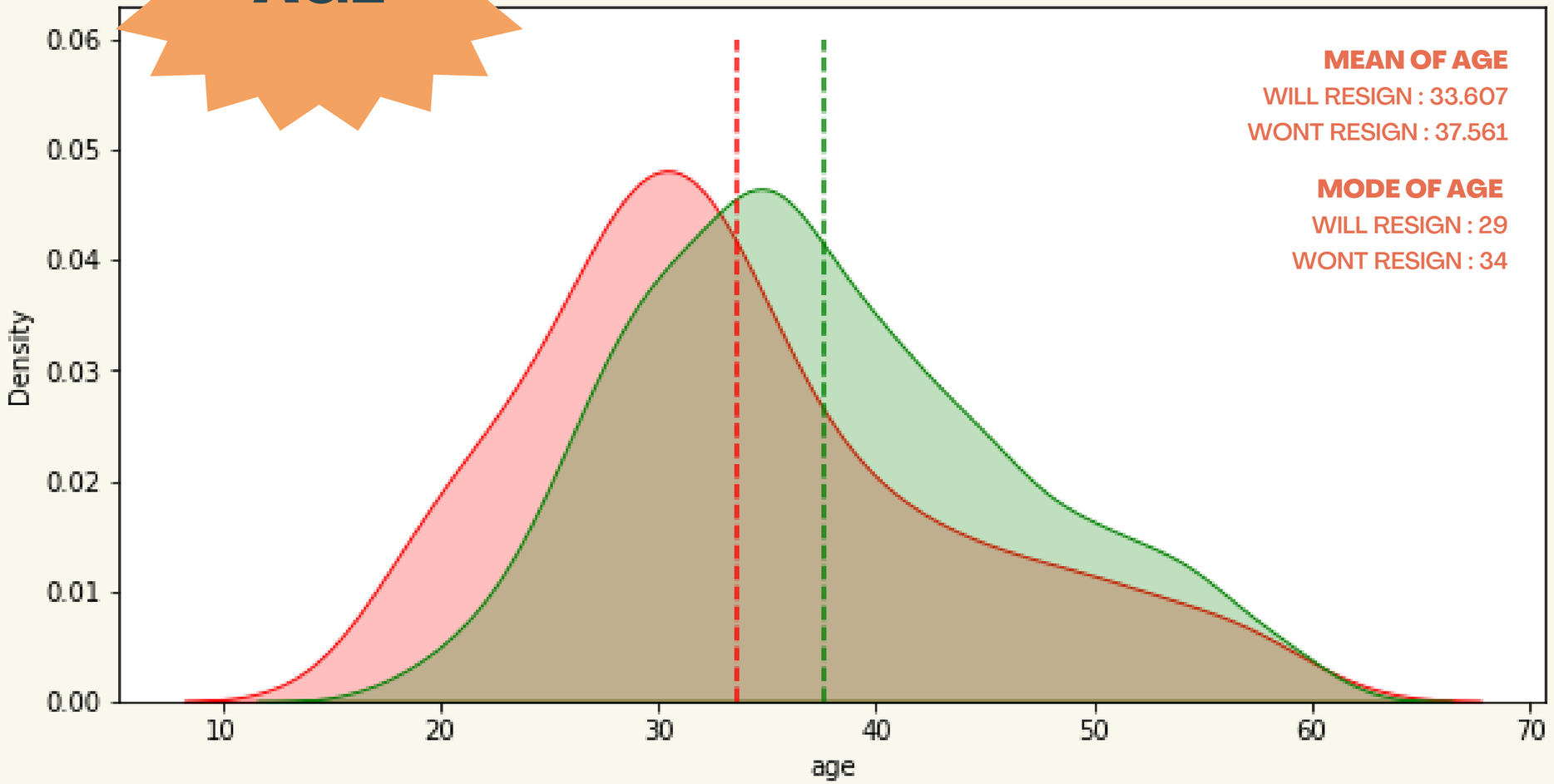
Lakukan analisis korelasi antar atribut, visualisasikan atribut-atribut yang memiliki korelasi. Jika ada, sampaikan pendapat anda mengenai keterkaitan atribut tersebut

A

Visualisasikan karakteristik karyawan yang resign dari perusahaan tersebut!

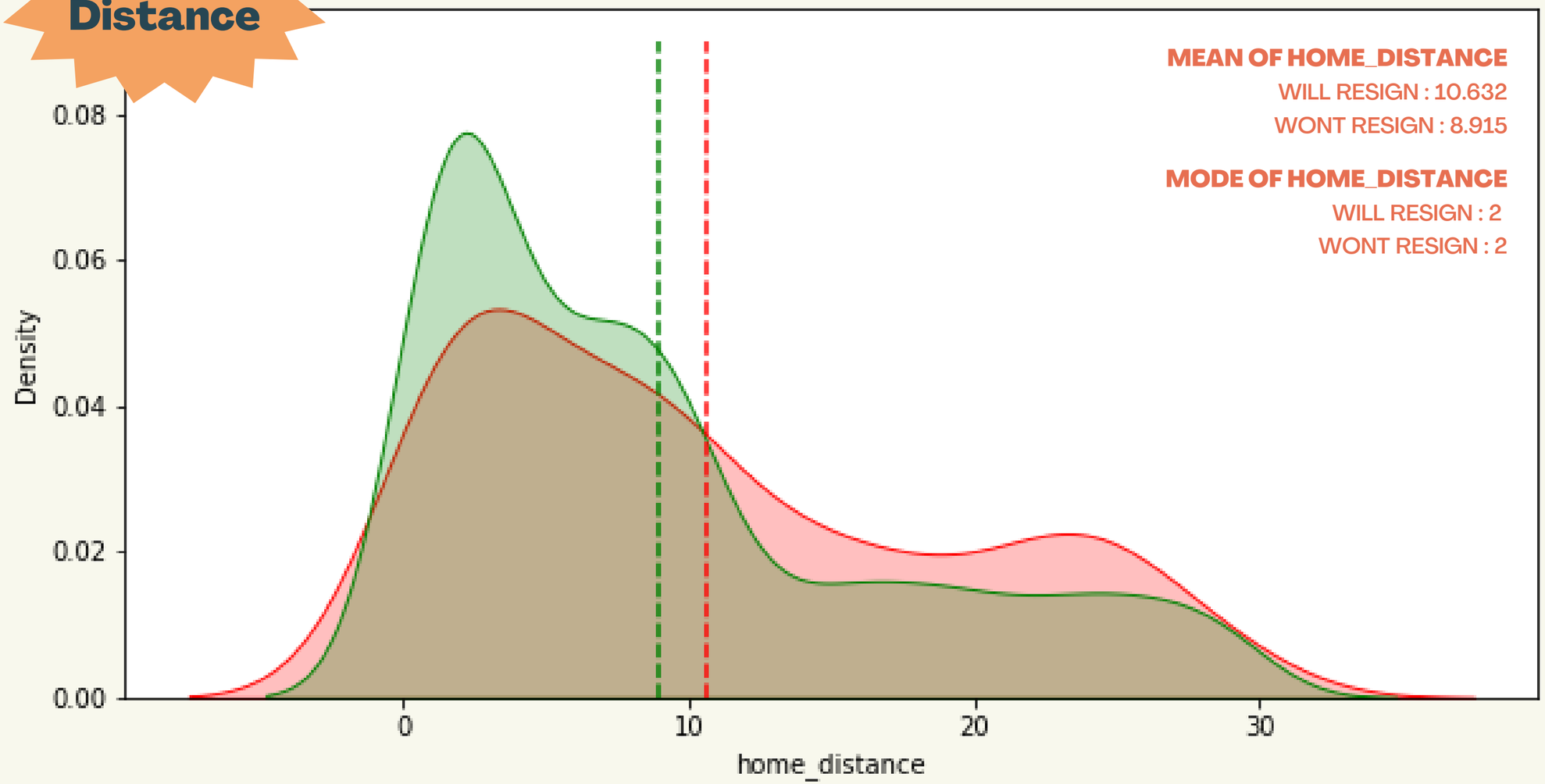
A

AGE



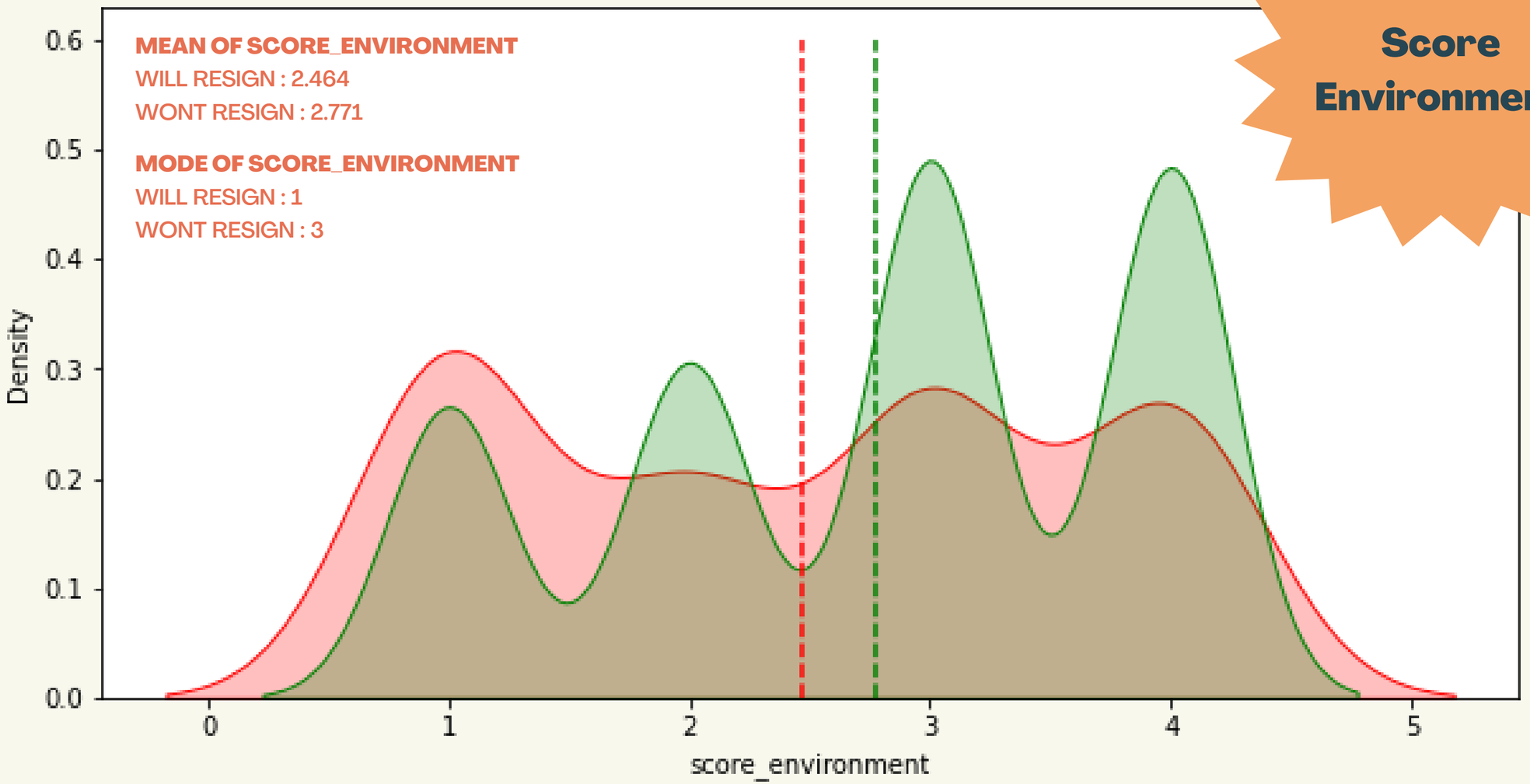
Karyawan yang resign mempunyai rata-rata umur 33.61 tahun dengan kebanyakan berumur 29 tahun

Home Distance



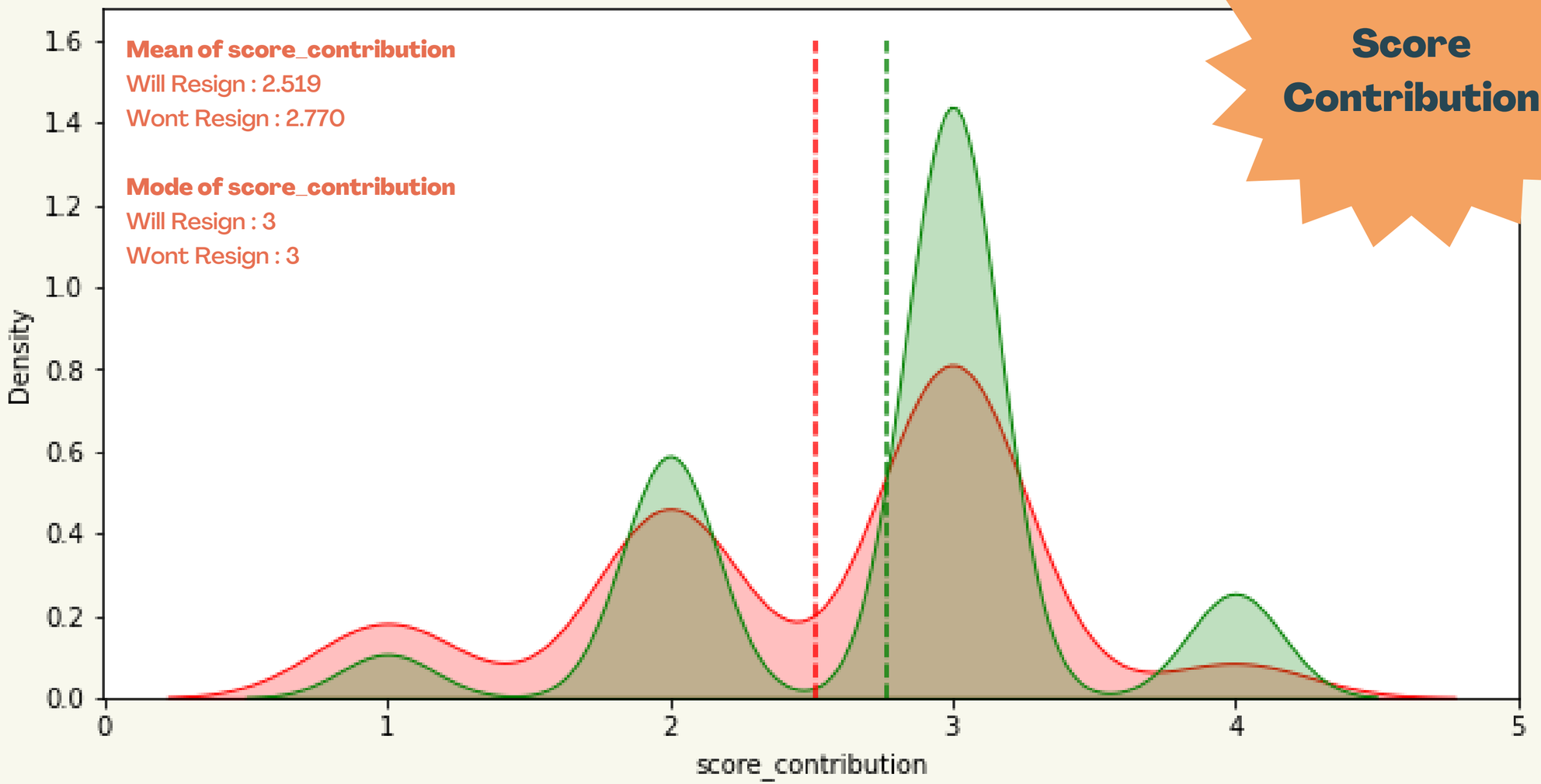
Karyawan yang resign mempunyai rata-rata jarak rumah ke kantor sejauh 10.63 km dengan kebanyakan berjarak 2 km

Score Environment



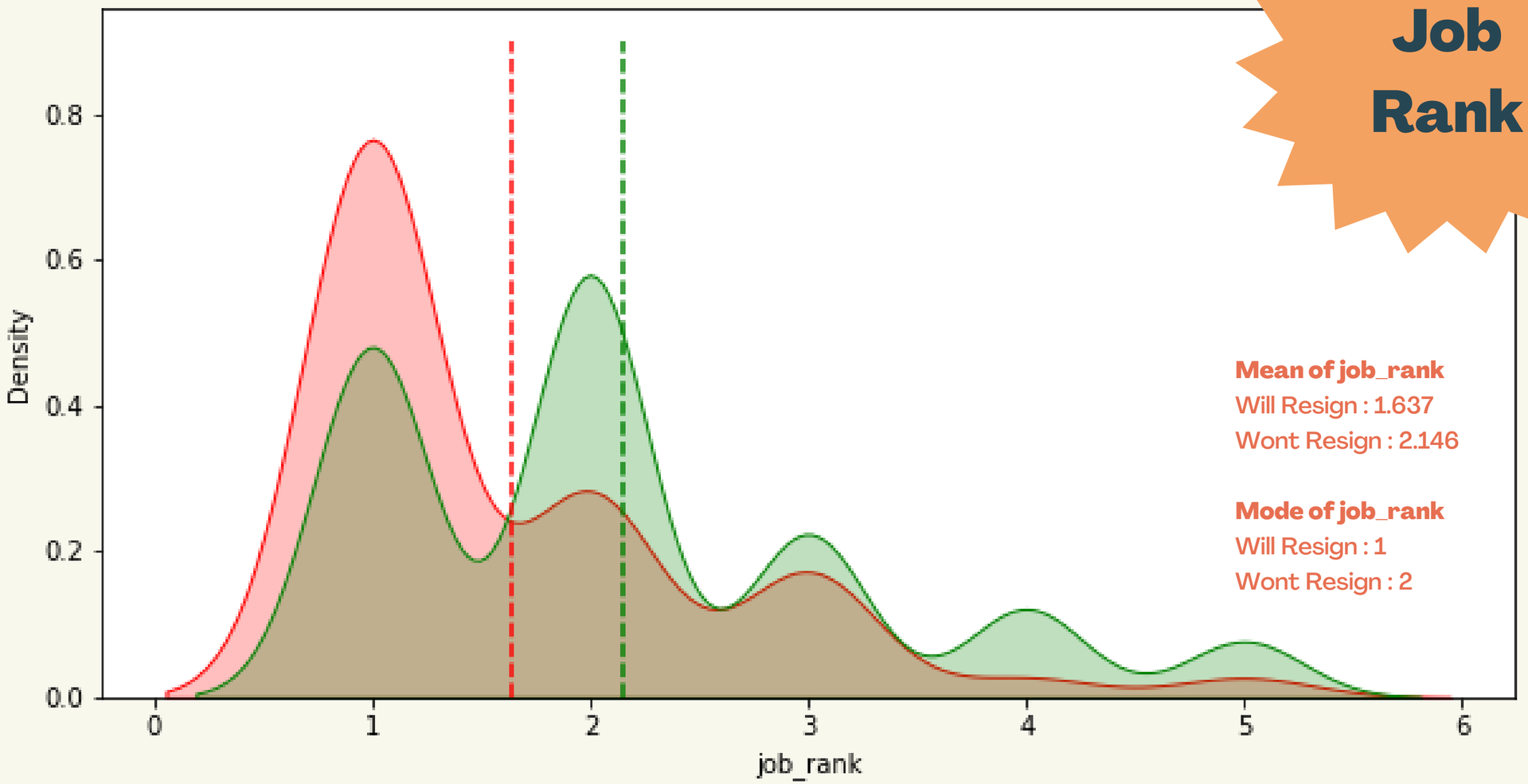
Rata-rata nilai kepuasan karyawan yang resign terhadap lingkungan kerja adalah 2.46 atau jika kita floor adalah 2 (rendah) dan kebanyakkan bernilai 1 (sangat rendah)

Score Contribution



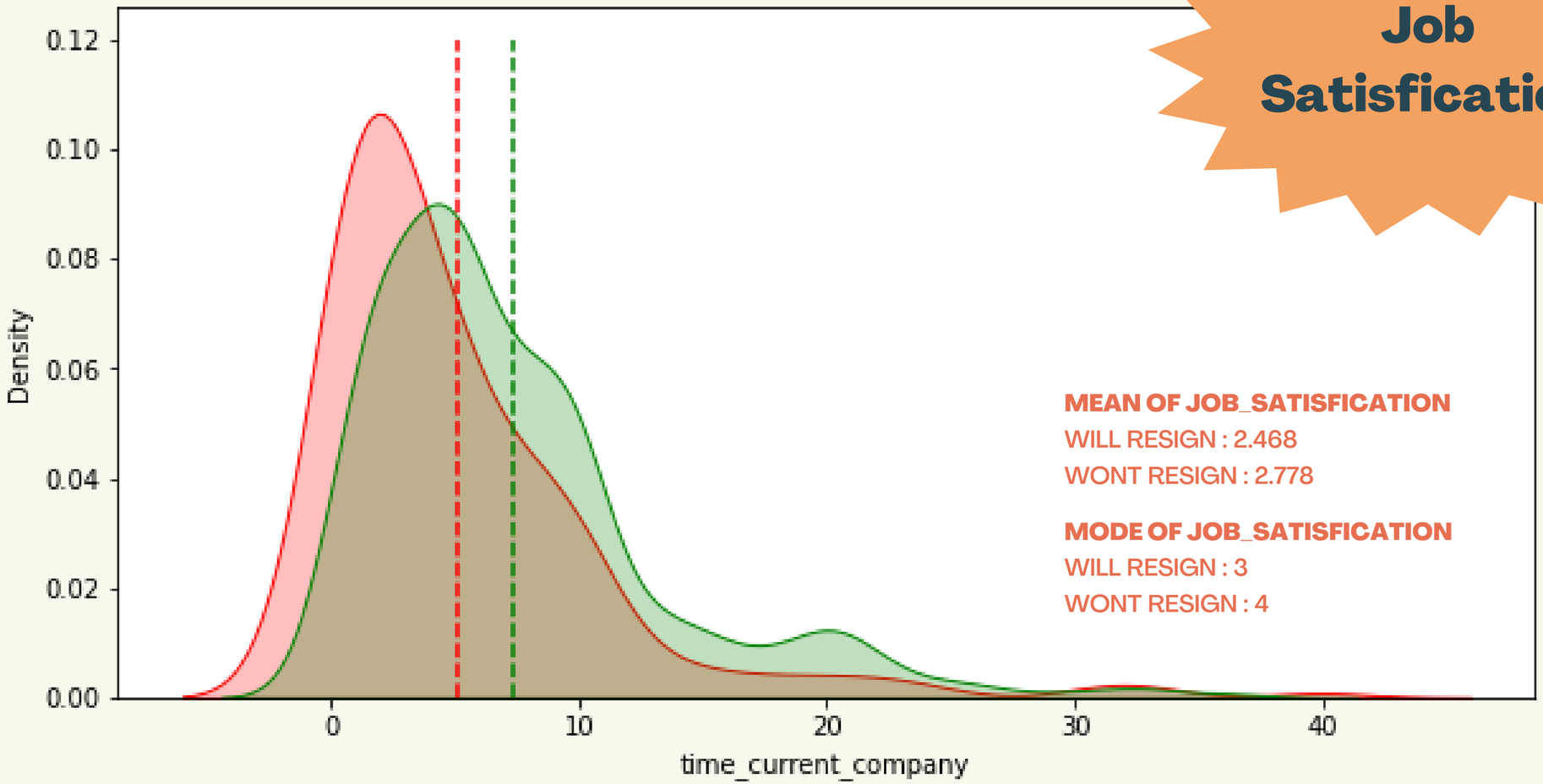
Rata-rata skor distribusi karyawan yang resign adalah sebesar 2.519 atau jika dibulatkan 3 (tinggi) dan kebanyakan sebesar 3 (tinggi).

Job Rank



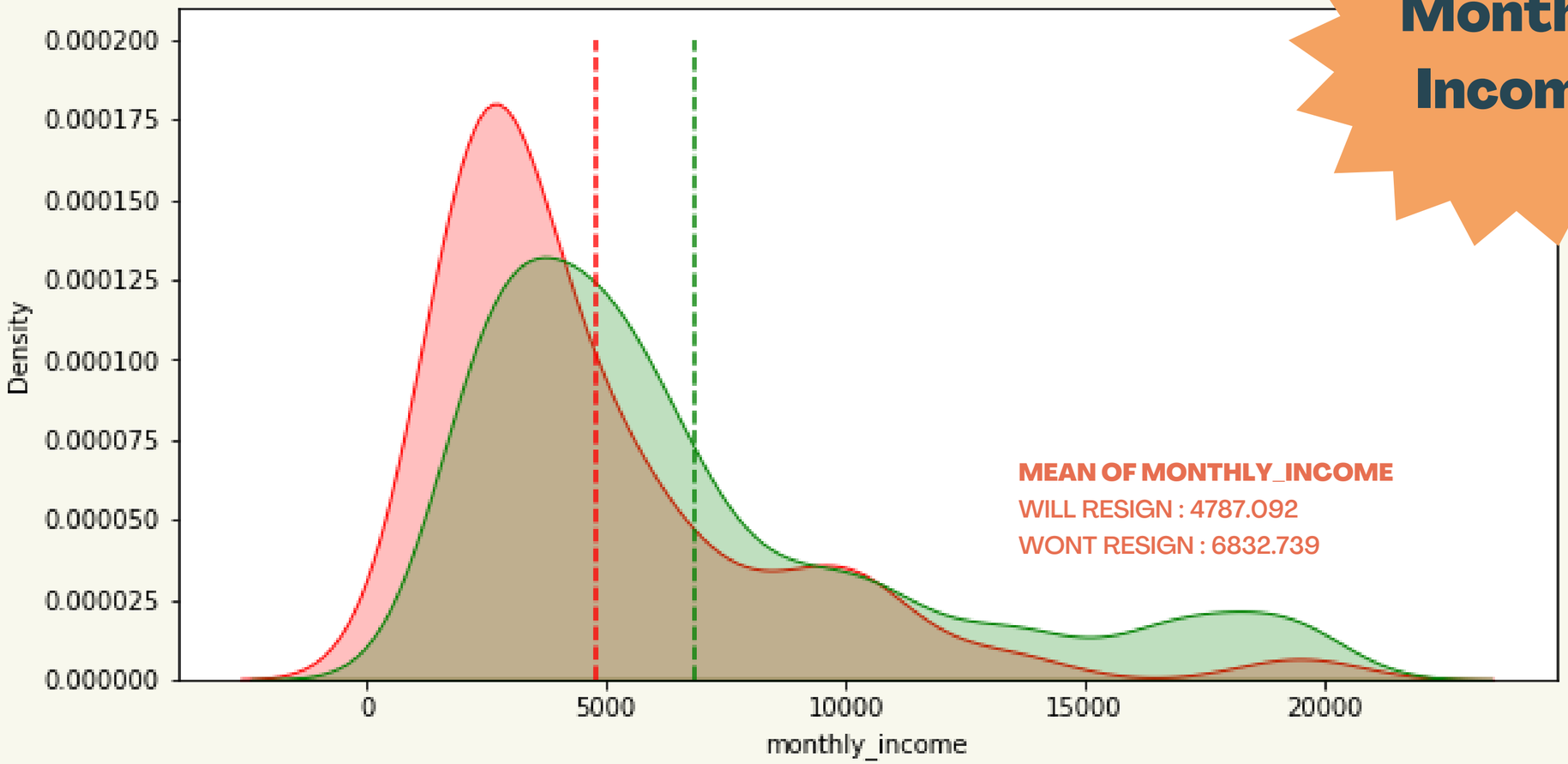
Rata-rata tingkat jabatan karyawan yang resign adalah 1.637 atau dibulatkan menjadi 2 (associate) dan kebanyakkan tingkat jabatannya adalah 1 (Entry Level)

Job Satisfication



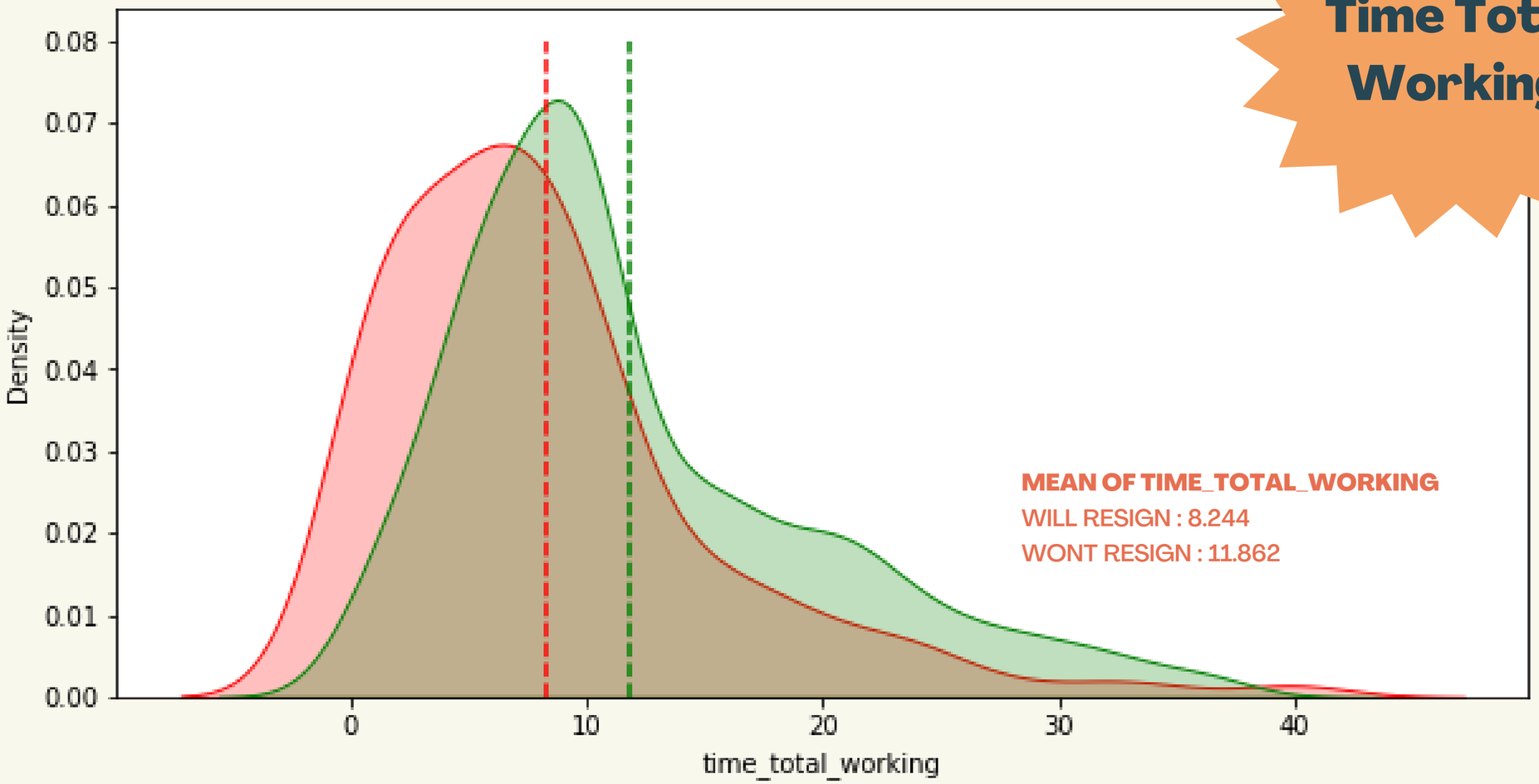
Rata-rata nilai kepuasan karyawan yang resign terhadap pekerjaannya adalah 2.468 dan kebanyakan nilai kepuasan karyawan adalah 3 (tinggi) terhadap pekerjaannya.

Monthly Income



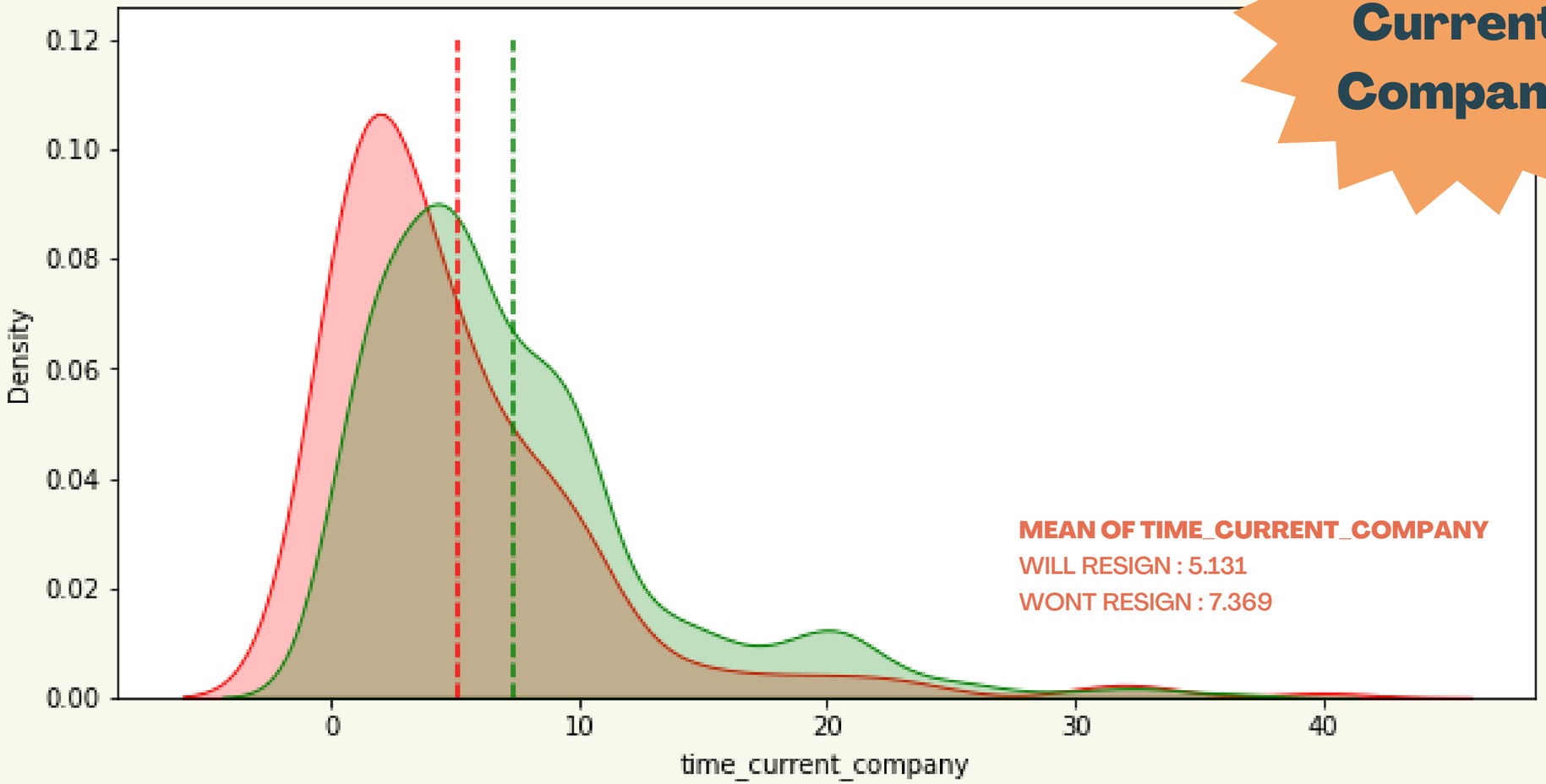
Karyawan yang resign memiliki karakteristik rata-rata gaji per bulan sebesar 4787.092 USD

Time Total Working



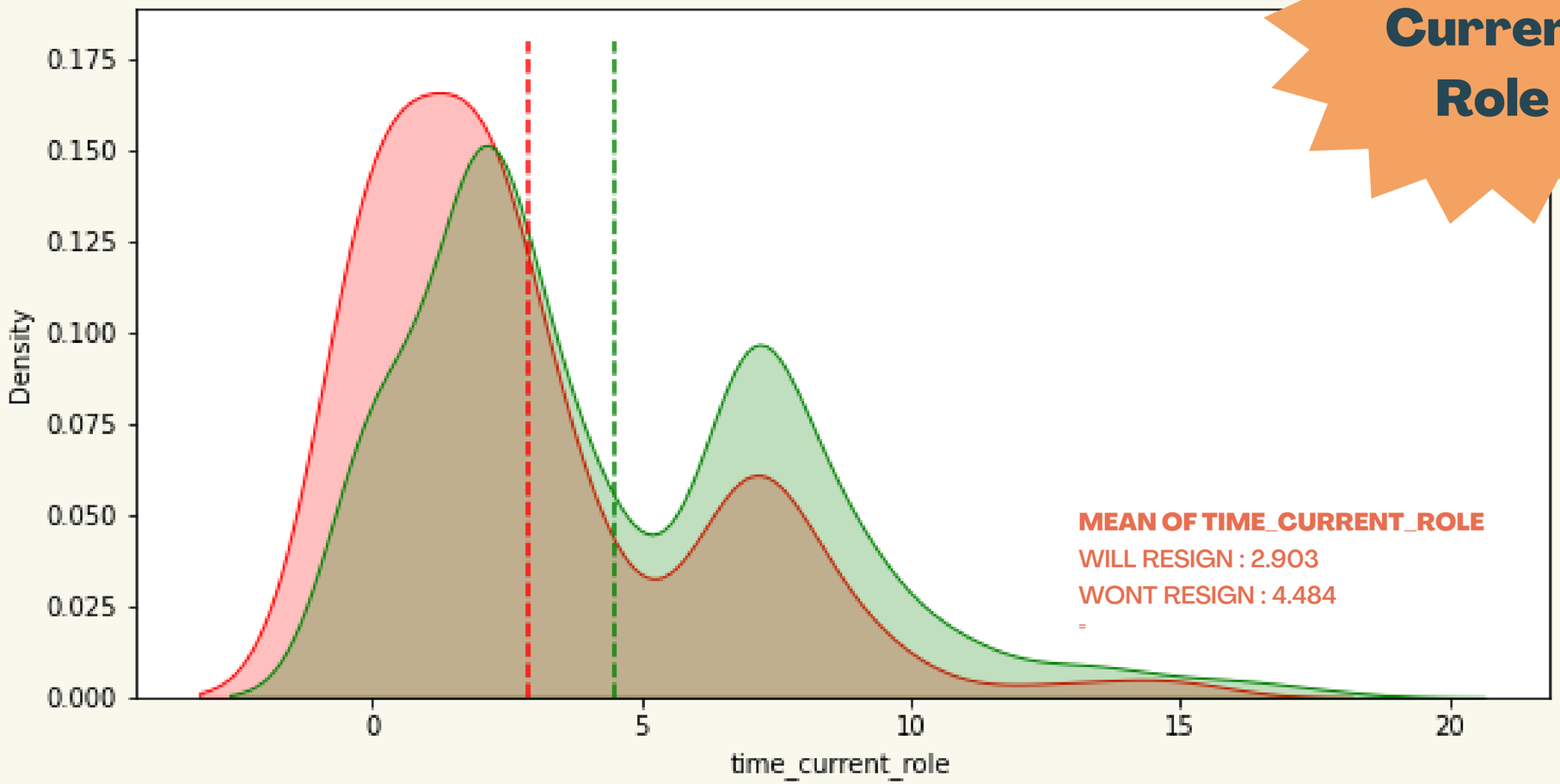
Karakteristik dari karyawan yang resign berdasarkan jumlah waktu pengalaman kerjanya memiliki rata-rata 8.244.

Time Current Company

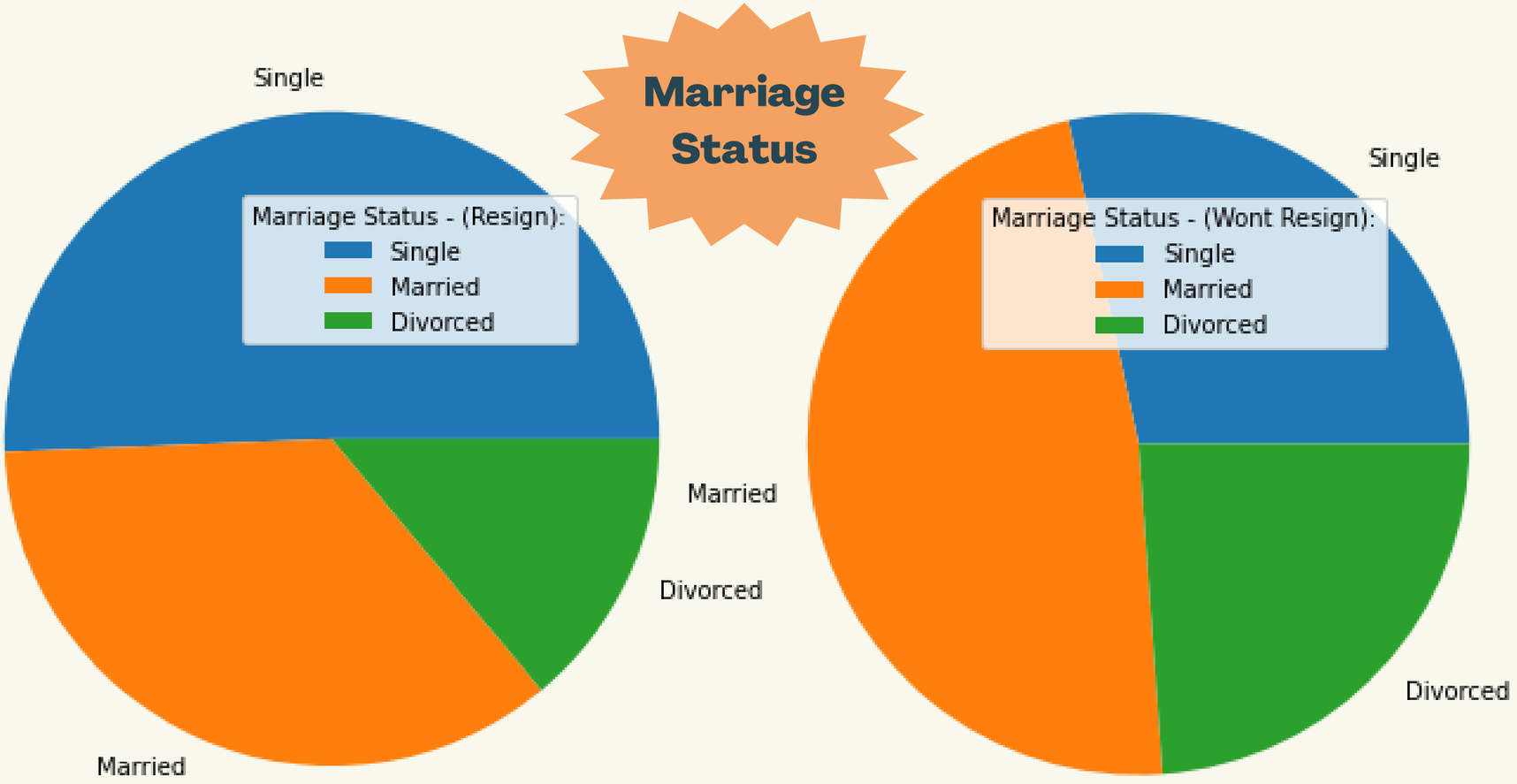


Rata-rata lama kerja karyawan yang resign di perusahaan adalah 5.131 Tahun.

Time Current Role

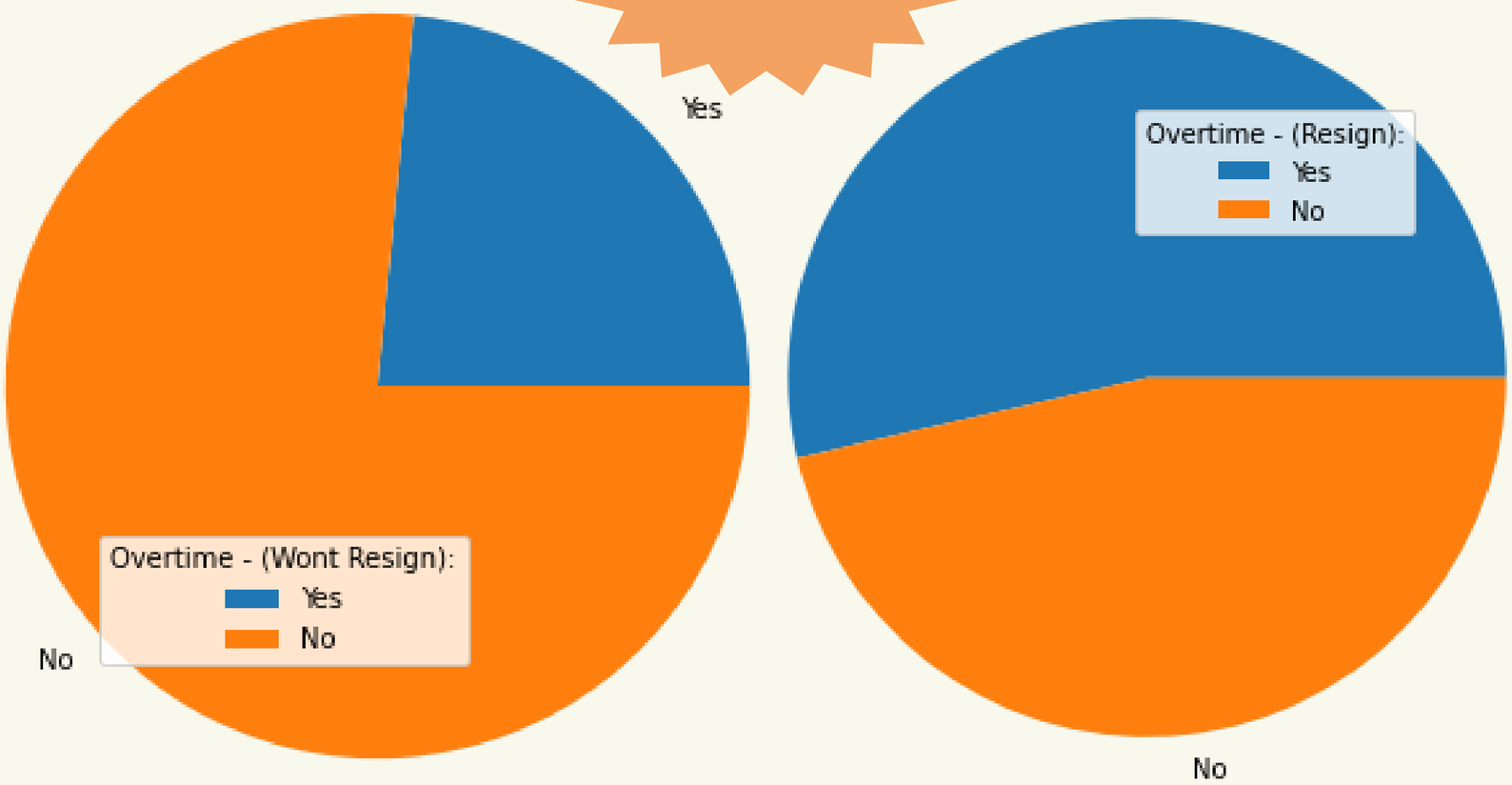


Rata-rata lama kerja karyawan yang resign di peran saat ini adalah 2.903 Tahun.



Dari data yang diperoleh, Karyawan yang resign kebanyakkan didominasi oleh karyawan yang status pernikahannya masih single.

Overtime



Dari data yang diperoleh, Karyawan yang resign kebanyakkan didominasi oleh karyawan yang mengalami overtime.

KESIMPULAN

Maka karyawan resign
memiliki karakteristik :

1. Mempunyai rata-rata umur 33.61 tahun dengan kebanyakkan berumur 29 tahun
2. Mempunyai rata-rata jarak rumah ke kantor sejauh 10.63 km dengan kebanyakan berjarak 2 km
3. Mempunyai rata-rata nilai kepuasan karyawan terhadap lingkungan kerja adalah 2.46 atau jika kita floor adalah 2 (rendah) dan kebanyakan bernilai 1 (sangat rendah)
4. Mempunyai rata-rata distribusi skor sebesar 2,519 dan kebanyakan sebesar 3

KESIMPULAN

Maka karyawan resign
memiliki karakteristik :

5. Memiliki tingkat jabatannya adalah kebanyakan yang memiliki jabatan 1 (Entry Level) dengan rata-rata jabatannya 1.637
6. Memiliki rata-rata nilai kepuasan karyawan terhadap pekerjaannya adalah 2.468 dan kebanyakan nilai kepuasan karyawan adalah 3 (tinggi) terhadap pekerjaannya.
7. Memiliki rata-rata gaji per bulan sebesar 4787.092 USD
8. Memiliki rata-rata jumlah waktu pengalaman kerjanya sebesar 8.244
9. Memiliki rata-rata lama kerja karyawan di perusahaan adalah 5.131 Tahun.
10. Memiliki rata-rata lama kerja karyawan di peran saat ini adalah 2.903 Tahun.

B

Apakah karyawan memilih untuk resign setelah mendapatkan promosi?

B

B

APAKAH KARYAWAN MEMILIH UNTUK RESIGN SETELAH MENDAPATKAN PROMOSI?

```
df2_promotion = df2[(df2.time_last_promotion == 0)]
df2_promotion = df2_promotion[(df2_promotion.time_current_company > 0)]
df2_promotion = df2_promotion[(df2_promotion.job_rank > 1)]

df2_promotion_yes = df2_promotion[(df2_promotion.resign == "Yes")]
df2_promotion_no = df2_promotion[(df2_promotion.resign == "No")]

print("RESIGN      :", len(df2_promotion_yes))
print("WONT RESIGN :", len(df2_promotion_no))
```

```
RESIGN      : 29
WONT RESIGN : 275
```

Kami masih belum bisa handle karyawan yang mendapatkan promosi dibawah satu tahun kerja.

Memilih karyawan yang memiliki waktu last promotion == 0 tahun, memiliki jumlah tahun kerja di perusahaan > 0 tahun, dan rank jabatannya diatas 1

Asumsi :

1. Waktu last promotion == 0, sebagai threshold penanda karyawan baru dipromosi.
2. Jumlah lama di perusahaan >0, untuk mengurangi kemungkinan karyawan yang memiliki job rank tersebut sejak awal masuk (tanpa promosi).
3. Job_rank >1, sebagai penanda bahwa karyawan sudah pernah mendapatkan promosi.

Membagi data menjadi karyawan yang resign dan karyawan yang tidak resign

RESULT

Tidak, karena berdasarkan threshold dan filter yang ditentukan, dapat dilihat bahwa karyawan yang akan resign lebih sedikit dibandingkan karyawan yang tidak akan resign

Departemen manakah yang memiliki karyawan loyal paling banyak?

C

C

C

DEPARTEMEN MANAKAH YANG MEMILIKI KARYAWAN LOYAL PALING BANYAK?

```
loyal_treshold = df2["time_current_company"].mean()

df2_marketing = df2[(df2.division == "Marketing")]
df2_hr = df2[(df2.division == "Human Resource")]
df2_hat = df2[(df2.division == "Health and Technology")]

loyal_count_marketing = len(df2_marketing[(df2_marketing.time_current_company >= loyal_treshold)])
loyal_count_hr = len(df2_hr[(df2_hr.time_current_company >= loyal_treshold)])
loyal_count_hat = len(df2_hat[(df2_hat.time_current_company >= loyal_treshold)])

print("Departemen Marketing :", loyal_count_marketing)
print("Departemen HR        :", loyal_count_hr)
print("Departemen H & T     :", loyal_count_hat)

print("================================")

def topercentage(dec):
  return str(round(dec * 100, 2)) + "%"

print("Departemen Marketing :", topercentage(loyal_count_marketing / len(df2_marketing)))
print("Departemen HR        :", topercentage(loyal_count_hr / len(df2_hr)))
print("Departemen H & T     :", topercentage(loyal_count_hat / len(df2_hat)))
```

RESULT

Kami menetapkan loyal threshold yaitu mean dari time_current_company seluruh karyawan sebagai batas loyalitas karyawan

Kemudian kami menghitung berapa karyawan yang memiliki time_current_company diatas threshold tersebut

Hasil perhitungan tersebut dapat dilihat pada pie chart disamping

```
Departemen Marketing : 176
Departemen HR        : 20
Departemen H & T     : 332
================================
Departemen Marketing : 39.46%
Departemen HR        : 31.75%
Departemen H & T     : 34.55%
```

D

Lakukan analisis korelasi antar atribut, visualisasikan atribut-atribut yang memiliki korelasi. Jika ada, sampaikan pendapat anda mengenai keterkaitan atribut tersebut!

D

D

LAKUKAN ANALISIS KORELASI ANTAR ATRIBUT, VISUALISASIKAN ATRIBUT-ATRIBUT YANG MEMILIKI KORELASI. JIKA ADA, SAMPAIKAN PENDAPAT ANDA MENGENAI KETERKAITAN ATRIBUT TERSEBUT

```
from sklearn.preprocessing import LabelEncoder

# Lakukan encoding untuk data kategorikal

df_encoded = df2.copy()

# Label encoding untuk data kategorikal boolean
le = LabelEncoder()
df_encoded["resign"] = le.fit_transform(df_encoded["resign"])
df_encoded["gender"] = le.fit_transform(df_encoded["gender"])
df_encoded["over_time"] = le.fit_transform(df_encoded["over_time"])

# One hot encoding untuk data kategorikal non boolean namun tidak ordinal
df_encoded = pd.get_dummies(df_encoded)
```

```
df_corr = df_encoded.corr(method='pearson')
f,ax = plt.subplots(figsize=(30, 30))
sns.heatmap(df_corr, annot=True, linewidths=.5, fmt= '.1f',ax=ax)
```

```
corr = df_encoded.corr(method='pearson')

columns = df_encoded.columns

for i in range(0, len(columns)):
  for j in range(i + 1, len(columns)):
    if corr[columns[i]][columns[j]] >= 0.5 or corr[columns[i]][columns[j]] <= -0.5:
      print("{:30s} {:30s} {:f}".format(columns[i], columns[j], corr[columns[i]][columns[j]]))
```

RESULT

Kami melakukan encoding untuk atribut-atribut yang memiliki data kategorikal. Setelah itu, kami mencoba menampilkan keseluruhan korelasi yang didapatkan melalui heatmap.

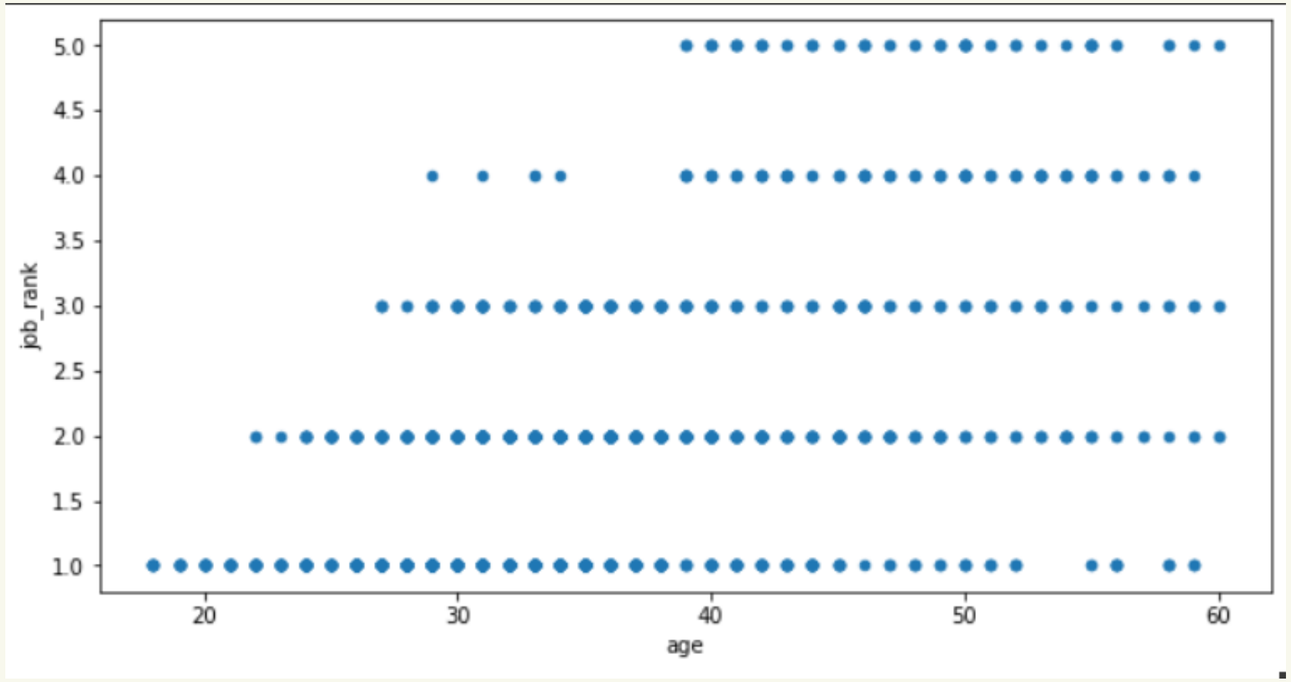
Kami memasukkan atribut-atribut yang memiliki korelasi dengan atribut lainnya yang memenuhi persyaratan >= 0,5 atau <= -0,5.

Hasil perhitungan tersebut dapat dilihat pada gambar di samping.

age	job_rank	0.509604
age	time_total_working	0.680381
job_rank	monthly_income	0.950300
job_rank	time_total_working	0.782208
job_rank	time_current_company	0.534739
job_rank	role_Manager	0.552744
monthly_income	time_total_working	0.772893
monthly_income	time_current_company	0.514285
monthly_income	role_Manager	0.619573
salary_increment_percentage	rate_performance	0.773550
time_total_working	time_current_company	0.628133
time_current_company	time_current_role	0.758754
time_current_company	time_last_promotion	0.618409
time_current_company	time_current_manager	0.769212
time_current_role	time_last_promotion	0.548056
time_current_role	time_current_manager	0.714365
time_last_promotion	time_current_manager	0.510224
division_Health and Technology	division_Marketing	-0.906818
division_Health and Technology	role_Sales Executive	-0.733497
division_Human Resource	major_Human Resources	0.646436
division_Human Resource	role_Human Resources	0.904983
division_Marketing	major_Marketing	0.527691
division_Marketing	role_Sales Executive	0.808869
major_Computer Science	major_Life Sciences	-0.568774
major_Human Resources	role_Human Resources	0.549751
marriage_status_Married	marriage_status_Single	-0.629981

VISUALISASI

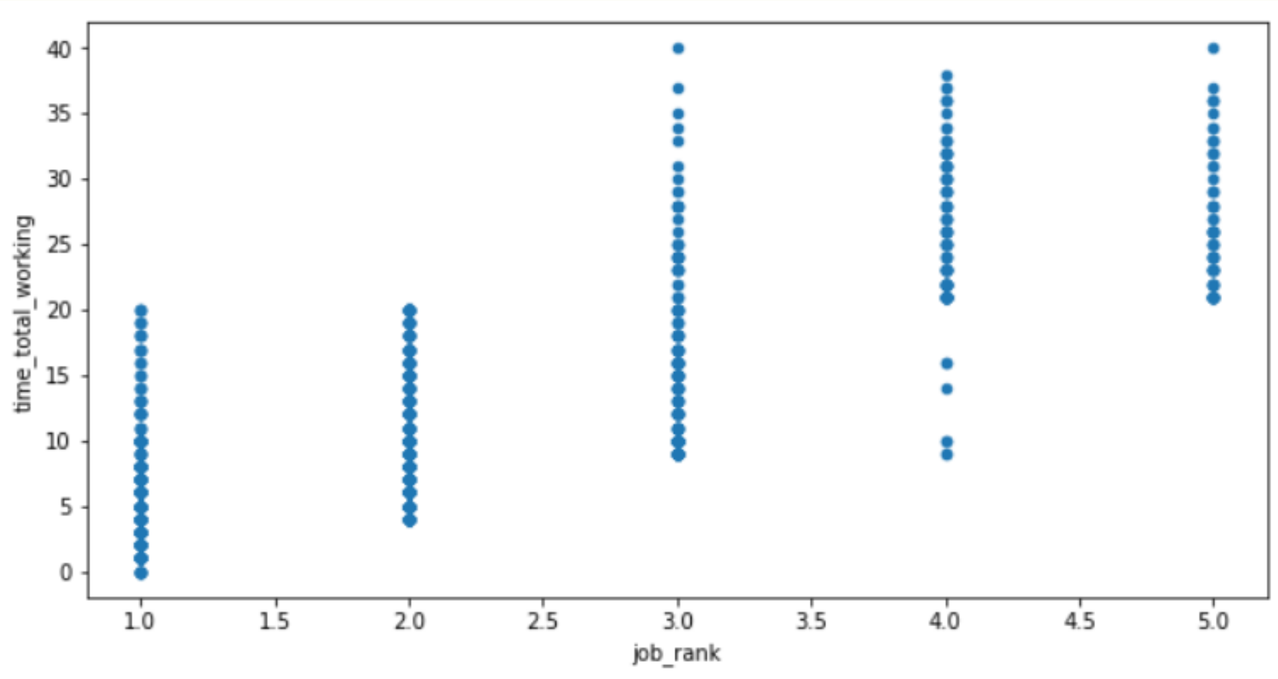
Age - Job Rank



Atribut Age dan Job Rank adalah atribut yang saling terkait dimana seringkali umur karyawan dapat diperkirakan melalui jabatan mereka, begitu juga sebaliknya

VISUALISASI

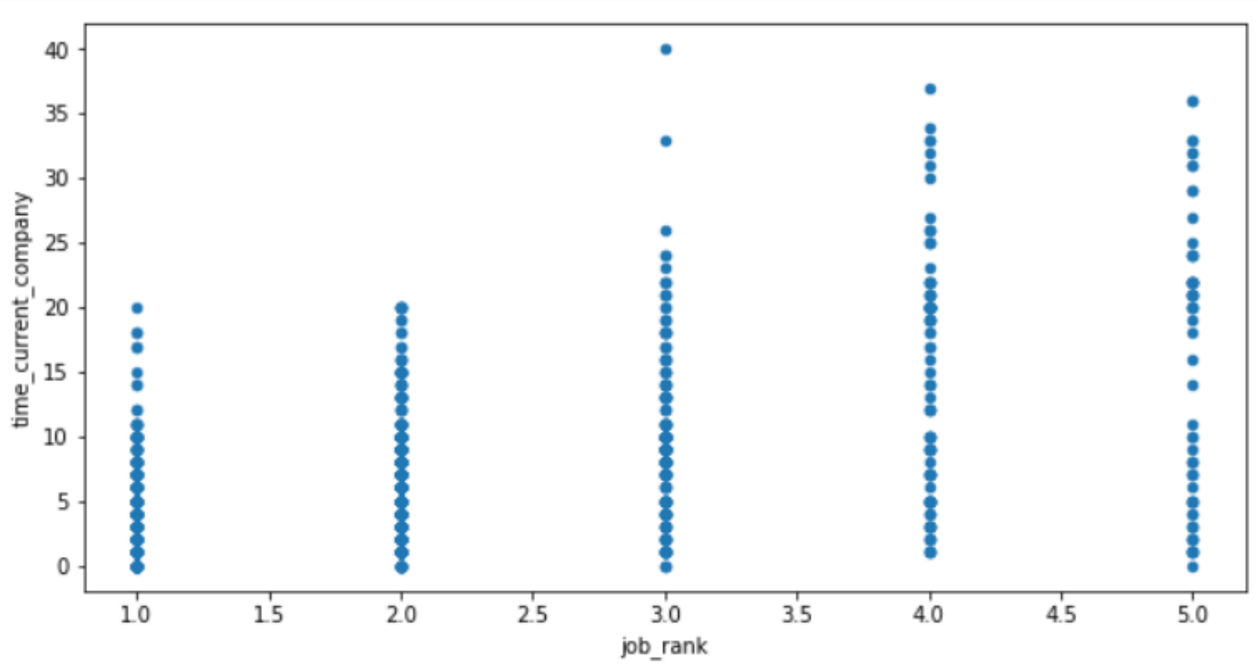
Job Rank - Time Total Working



Atribut Job Rank dan Time Total Working adalah atribut yang saling terkait dimana jumlah waktu pengalaman kerja dapat dilihat dari jabatan mereka, begitu juga sebaliknya

VISUALISASI

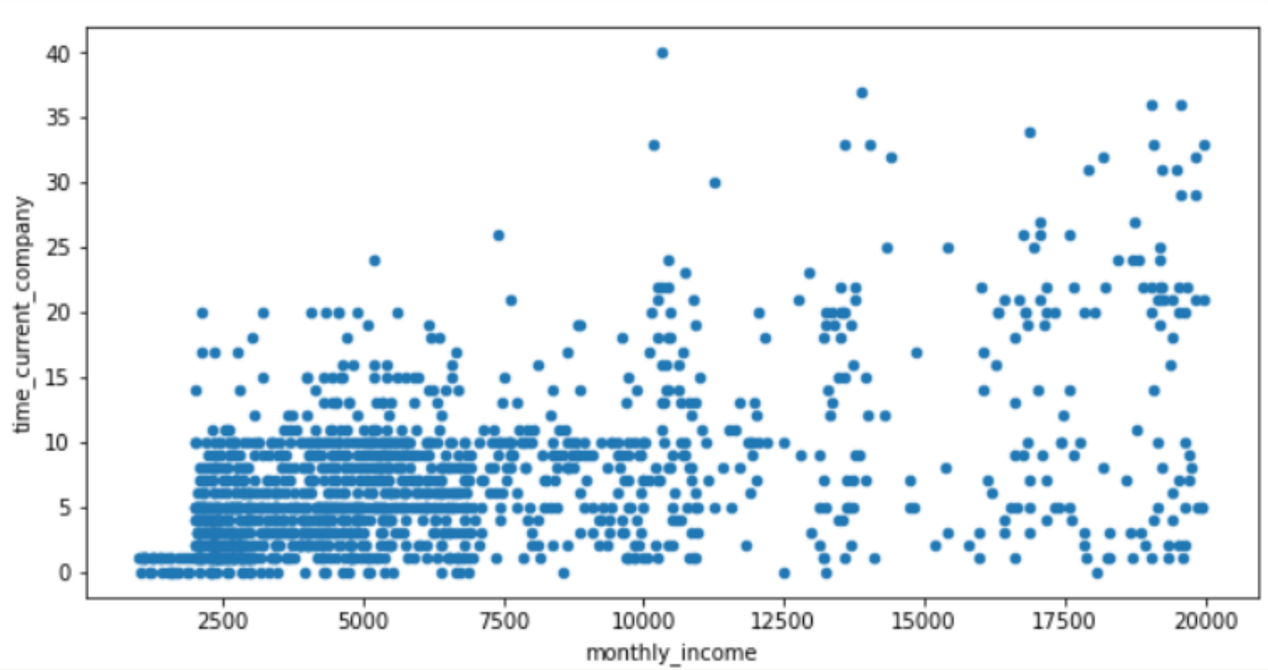
Job Rank - Time Current Company



Atribut Job Rank dan Time Current Company adalah atribut yang saling terkait dimana kita dapat memperkirakan jabatan karyawan melalui lama mereka bekerja di perusahaan tersebut, begitu juga sebaliknya

VISUALISASI

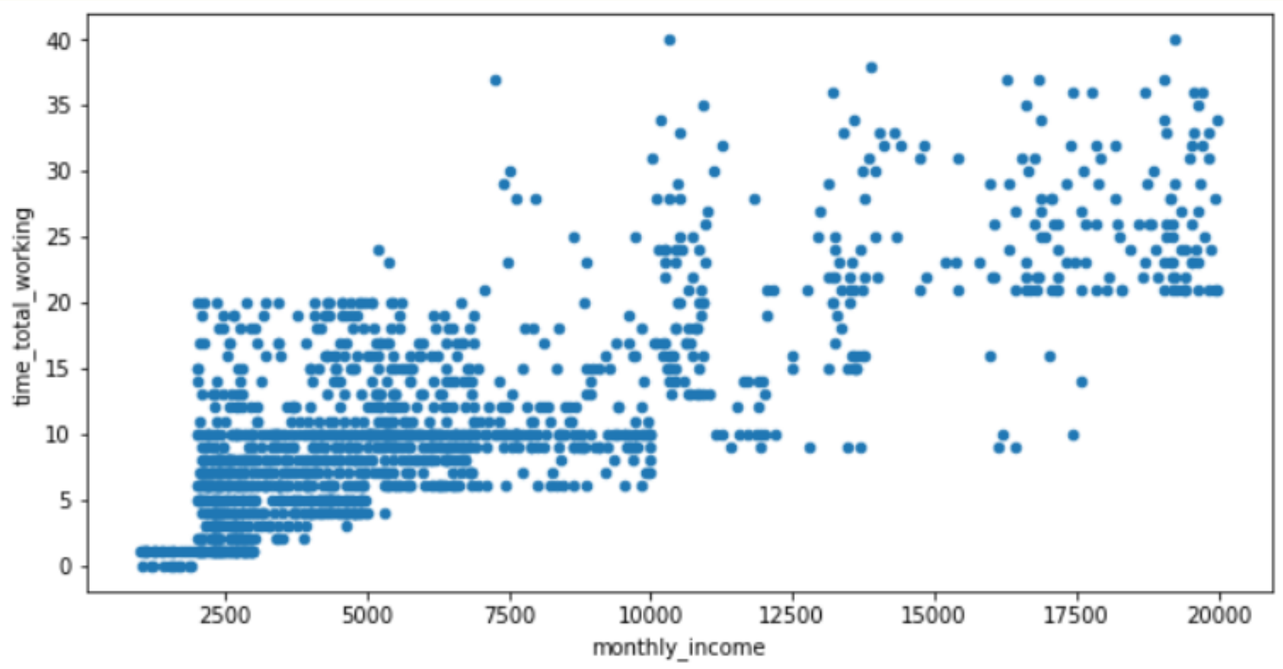
Monthly Income - Time Current Company



Atribut Monthly Income dan Time Current Company adalah atribut yang saling terkait dimana pendapatan karyawan dapat diperkirakan melalui lama mereka bekerja di perusahaan tersebut, begitu juga sebaliknya

VISUALISASI

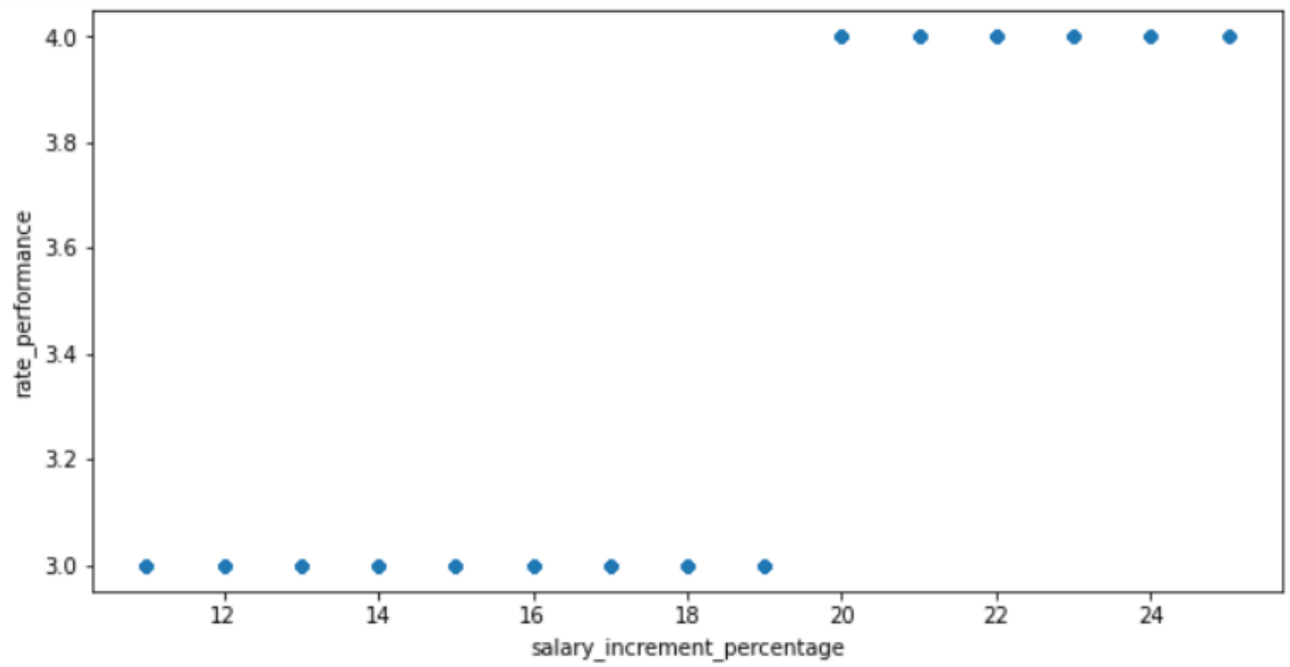
Monthly Income - Time Total Working



Atribut Monthly Income dan Time Total Working adalah atribut yang saling terkait dimana pendapatan karyawan dapat diperkirakan melalui waktu pengalaman kerja mereka secara keseluruhan, begitu juga sebaliknya

VISUALISASI

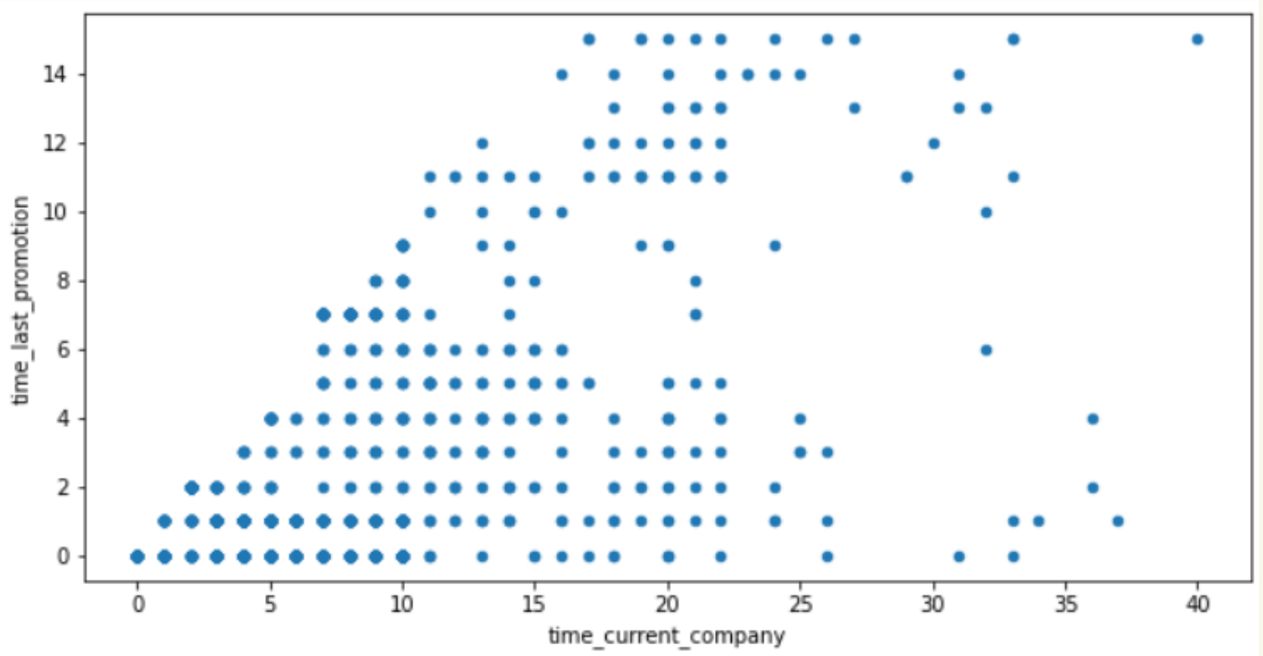
Salary Increment - Rate Performance



Atribut Salary Increment dan Rate Performance adalah atribut yang saling terkait dimana persentase kenaikan gaji karyawan dapat diperkirakan melalui kinerja mereka, begitu juga sebaliknya

VISUALISASI

Time Current Company - Time Last Promotion



Atribut Time Current Company dan Time Last Promotion adalah atribut yang saling terkait dimana lama seseorang bekerja di perusahaan tersebut dapat diperkirakan melalui promosi terakhir mereka, begitu juga sebaliknya

PROGRES KELOMPOK

EKSPLORASI TAMBAHAN

01

Apakah penilaian karyawan terhadap dirinya sendiri sebanding dengan penilaian oleh perusahaan?

02

Role apa yang mempunyai gaji Entry Level paling tinggi?

01

Apakah penilaian karyawan terhadap dirinya sendiri sebanding dengan penilaian oleh perusahaan?

01

1 APAKAH PENILAIAN KARYAWAN TERHADAP DIRINYA SENDIRI SEBANDING DENGAN PENILAIAN OLEH PERUSAHAAN?

```
df2_marketing = df2[(df2.division == "Marketing")]
df2_hr = df2[(df2.division == "Human Resource")]
df2_hat = df2[(df2.division == "Health and Technology")]

sc_marketing = df2_marketing.score_contribution.mean()
sc_hr = df2_hr.score_contribution.mean()
sc_hat = df2_hat.score_contribution.mean()
rp_marketing = df2_marketing.rate_performance.mean()
rp_hr = df2_hr.rate_performance.mean()
rp_hat = df2_hat.rate_performance.mean()

print("======= Marketing =======")
print("SC mean    :", round(sc_marketing, 2))
print("RP mean    :", round(rp_marketing, 2))
print("Difference :", round(abs(sc_marketing - rp_marketing), 2))
print("==== Human Resources ====")
print("SC mean    :", round(sc_hr, 2))
print("RP mean    :", round(rp_hr, 2))
print("Difference :", round(abs(sc_hr - rp_hr), 2))
print("== Health & Technology ==")
print("SC mean    :", round(sc_hat, 2))
print("RP mean    :", round(rp_hat, 2))
print("Difference :", round(abs(sc_hat - rp_hat), 2))
```

Filter data dengan memisahkan data menjadi karyawan per divisi.

Menghitung rata-rata score_contribution dan rate_performance setiap karyawan per divisinya.

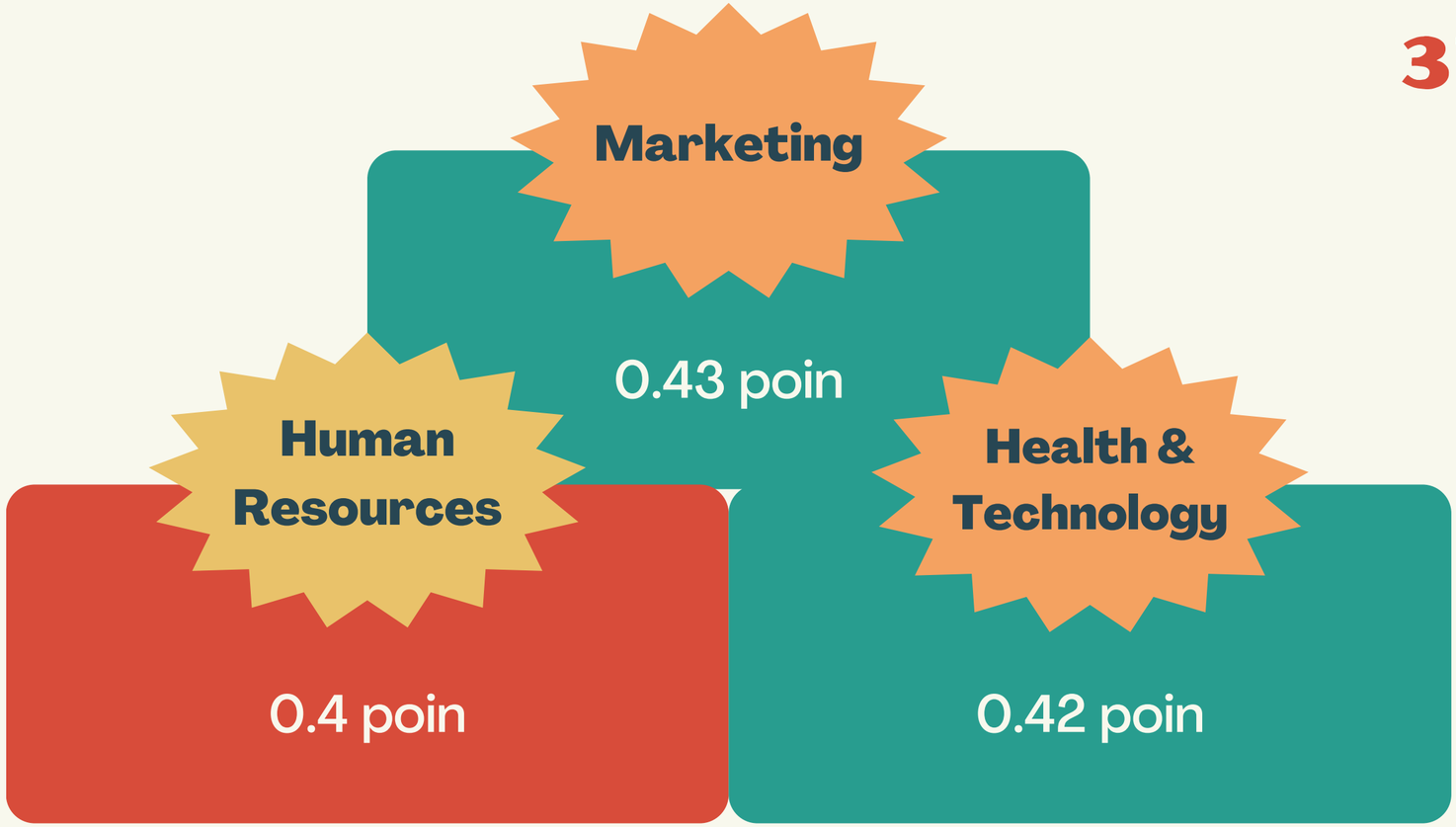
Asumsi :

1. Score_contribution adalah peniliaian karyawan terhadap kontribusi mereka ke perusahaan
2. Rate_performance adalah nilai performa karyawan yang ditentukan oleh perusahaan.

Menghitung selisih penilaian performa karyawan terhadap diri sendiri dengan penilaian dari perusahaan.

RESULT

```
======= Marketing =======
SC mean     : 2.7
RP mean     : 3.14
Difference : 0.43
==== Human Resources ====
SC mean     : 2.75
RP mean     : 3.14
Difference : 0.4
== Health & Technology ==
SC mean     : 2.74
RP mean     : 3.16
Difference : 0.42
```



Berdasarkan hasil yang didapatkan, pada seluruh divisi, terlihat bahwa penilaian performa dari perusahaan lebih tinggi dibandingkan penilaian karyawan terhadap diri sendiri.

02

Dari karyawan yang bekerja di tingkat entry level, role apa yang mempunyai gaji paling tinggi?

02

2 DARI KARYAWAN YANG BEKERJA DI TINGKAT ENTRY LEVEL, ROLE APA YANG MEMPUNYAI GAJI PALING TINGGI?

```
df2_rank_1 = df2[(df2.job_rank == 1)]
print(df2_rank_1["role"].unique())
```

```
['Laboratory Technician' 'Research Scientist' 'Sales Representative'
 'Human Resources']
```

```
lt_mean = df2_rank_1[(df2_rank_1.role == "Laboratory Technician")]["monthly_income"].mean()
rs_mean = df2_rank_1[(df2_rank_1.role == "Research Scientist")]["monthly_income"].mean()
sr_mean = df2_rank_1[(df2_rank_1.role == "Sales Representative")]["monthly_income"].mean()
hr_mean = df2_rank_1[(df2_rank_1.role == "Human Resources")]["monthly_income"].mean()

print("Laboratory Technician :", round(lt_mean, 2))
print("Research Scientist    :", round(rs_mean, 2))
print("Sales Representative  :", round(sr_mean, 2))
print("Human Resources       :", round(hr_mean, 2))
```

```
Laboratory Technician : 2854.59
Research Scientist    : 2827.65
Sales Representative  : 2506.72
Human Resources       : 2733.21
```

Melihat role apa saja yang ada pada jabatan entry level.

Menghitung rata-rata monthly_income pada setiap role entry level.

RESULT

Laboratory Technician

2854.59 USD

Research Scientist

2827.65 USD

Sales Representative

2506.72 USD

Human Resources

2733.21 USD

Dapat dilihat bahwa pada jabatan entry level, role dengan gaji tertinggi dimiliki oleh role Laboratory Technician

PROGRES KELOMPOK

SOAL NOMOR 2

A

Lakukan prediksi untuk mengetahui apakah karyawan akan resign atau tidak di perusahaan tersebut. Bagaimana hasil prediksi anda dapat membantu perusahaan dalam mengambil keputusan?

B

Lakukan prediksi untuk mengetahui berapa lama seorang karyawan akan bertahan di perusahaan tersebut. Bagaimana hasil prediksi anda dapat membantu perusahaan dalam mengambil keputusan?

C

Lakukan analisis cluster yang dapat terbentuk pada data karyawan. Deskripsikan karakteristik masing-masing cluster yang didapatkan!

PREPROCESSING

1

```
# Membuang kolom yang tidak diperlukan

df_processed = df_processed.drop('underage', axis=1)
df_processed = df_processed.drop('working_hours', axis=1)
df_processed = df_processed.drop('employee_id', axis=1)
```

Membuang kolom-kolom yang tidak diperlukan karena nilai pada feature tersebut sama atau tidak berpengaruh

PREPROCESSING

2

```
Mengecek nilai null

def cek_null(df):
    col_na = df.isnull().sum().sort_values(ascending=True)
    percent = col_na / len(df)

    missing_data = pd.concat([col_na, percent], axis=1, keys=['Total', 'Percent'])

    if (missing_data[missing_data['Total'] > 0].shape[0] == 0):
        print("Tidak ditemukan missing value pada dataset")

    else:
        print(missing_data[missing_data['Total'] > 0])

cek_null(df_processed)
```

```
Tidak ditemukan missing value pada dataset
```

cek nilai null yang ada pada datset.

PREPROCESSING

3

```
# Mengecek keberadaan nilai duplikat

print("Jumlah duplikasi data : " + str(df_processed.duplicated().sum()))
```

```
Jumlah duplikasi data : 0
```

cek jumlah duplikasi pada nilai-nilai di dataset

PREPROCESSING

4

	Column	Outlier	persentase
0	last_year_training_time	238	0.161905
1	rate_performance	226	0.153741
2	monthly_income	114	0.077551
3	time_last_promotion	107	0.072789
4	time_current_company	104	0.070748
5	time_total_working	63	0.042857
6	companies_count	52	0.035374
7	time_current_role	21	0.014286
8	time_current_manager	14	0.009524
9	score_work_life_balance	0	0.000000
10	score_work_relationahip	0	0.000000
11	age	0	0.000000
12	home_distance	0	0.000000
13	score_job_satisfication	0	0.000000
14	job_rank	0	0.000000
15	score_contribution	0	0.000000
16	hourly_rate	0	0.000000
17	score_environment	0	0.000000
18	education	0	0.000000
19	salary_increment_percentage	0	0.000000

cek jumlah dan presentasi jumlah outlier pada setiap feature di dataset

PREPROCESSING

5

```
ocessed.skew(axis = 0, skipna = True)
```

```
ython-input-419-ec2dc7fdb642>:1: FutureWa
  df_processed.skew(axis = 0, skipna = True)
age                              0.413286
home_distance                    0.958118
education                       -0.289681
score_environment               -0.321654
hourly_rate                     -0.032311
score_contribution              -0.498419
job_rank                         1.025401
score_job_satisfication         -0.329672
monthly_income                   1.369817
companies_count                  1.026471
salary_increment_percentage      0.821128
rate_performance                 1.921883
score_work_relationahip         -0.302828
time_total_working               1.117172
last_year_training_time          0.553124
score_work_life_balance         -0.552480
time_current_company             1.764529
time_current_role                0.917363
time_last_promotion              1.984290
time_current_manager             0.833451
dtype: float64
```

Cek apakah dataset terdistribusi normal dengan nilai skewness di setiap feature

PREPROCESSING

6

```
Lakukan encoding untuk dataset

# Label encoding untuk data kategorikal boolean
le = LabelEncoder()
df_processed["resign"] = le.fit_transform(df_processed["resign"])
df_processed["gender"] = le.fit_transform(df_processed["gender"])
df_processed["over_time"] = le.fit_transform(df_processed["over_time"])

# One hot encoding untuk data kategorikal non boolean namun tidak ordinal
df_processed = pd.get_dummies(df_processed)
```

Melakukan label encoding pada data kategorikal dan one hot encoding pada data kategorikal yang tidak ordinal

A

Lakukan prediksi untuk mengetahui apakah karyawan akan resign atau tidak di perusahaan tersebut. Bagaimana hasil prediksi anda dapat membantu perusahaan dalam mengambil keputusan?

A

A

LAKUKAN PREDIKSI UNTUK MENGETAHUI APAKAH KARYAWAN AKAN RESIGN ATAU TIDAK DI PERUSAHAAN TERSEBUT.

```
X1 = df_processed.drop('resign', axis=1)
y1 = df_processed['resign']

X1_train, X1_test, y1_train, y1_test = train_test_split(X1, y1, test_size = 0.2, random_state = 123)

scaler = MinMaxScaler()
X1_train_scaled = scaler.fit_transform(X1_train)
X1_test_scaled = scaler.transform(X1_test)

logisticRegressionModel1 = LogisticRegression()
logisticRegressionModel1.fit(X1_train_scaled, y1_train)
prediction1 = logisticRegressionModel1.predict(X1_test_scaled)
```

- Pertama-tama, kami menghilangkan feature 'resign' karena feature ini untuk dijadikan target.
- Kemudian kami membaga dataset menjadi data training dan data testing. Setelah kedua data dinormalisasi, kami menggunakan logistic regression untuk memprediksi data training terhadap data testing.
- Kami menggunakan logistic regression karena cocok untuk memprediksi data yang memiliki nilai korelasi yang rendah . Output dari logistic regression ini berupa 0/1 label.

RESULT

CONFUSION MATRIX

Confusion Matrix

Prediction	0	1
Actual		
0	248	8
1	22	16

CLASSIFICATION REPORT

Hasil Evaluasi berdasarkan classification report

	precision	recall	f1-score	support
0	0.92	0.97	0.94	256
1	0.67	0.42	0.52	38
accuracy			0.90	294
macro avg	0.79	0.69	0.73	294
weighted avg	0.89	0.90	0.89	294

Informasi lebih lengkap
F1 Macro Average: 0.7295474058628726
F1 Micro Average: 0.8979591836734694
Precision Macro Average: 0.7925925925925925
Precision Micro Average: 0.8979591836734694
Recall Macro Average: 0.6949013157894737
Recall Micro Average: 0.8979591836734694

Accuracy dari model yang kami bentuk untuk mengetahui karyawan akan resign atau tidak di perusahaan tersebut sudah cukup tinggi, dibuktikan dengan nilai akurasinya yaitu 0.898.

A BAGAIMANA HASIL PREDIKSI ANDA DAPAT MEMBANTU PERUSAHAAN DALAM MENGAMBIL KEPUTUSAN?

Retention efforts

Perusahaan dapat memberikan effort ,seperti training, yang menargetkan karyawan yang akan resign (dengan mempertimbangkan alasan karyawan yang akan resign)

Succession planning

Perusahaan dapat mempersiapkan lebih dulu untuk pengganti karyawan-karyawan yang akan resign

Budgeting and forecasting

Data yang memprediksi karyawan yang akan resign dapat dijadikan dasar pengambilan keputusan-keputusan perusahaan seperti budgeting

Resource allocation

Pengalokasian task dan karyawan juga bisa terpengaruh dengan data prediksi ini, dimana bidang yang memiliki high risk karyawan yang akan resign dibutuhkan alokasi yang baik

B

Lakukan prediksi untuk mengetahui berapa lama seorang karyawan akan bertahan di perusahaan tersebut. Bagaimana hasil prediksi anda dapat membantu perusahaan dalam mengambil keputusan?

B

B

LAKUKAN PREDIKSI UNTUK MENGETAHUI BERAPA LAMA SEORANG KARYAWAN AKAN BERTAHAN DI PERUSAHAAN TERSEBUT.

```
# Mengambil dataset yang berisi hanya orang yang resign

df_yes = df_processed.copy()
df_yes = df_yes[(df_yes.resign == 1)]

# Menjadikan time_current_company sebagai target variable

X2 = df_yes.drop('time_current_company', axis=1)
y2 = df_yes['time_current_company']

# Split dataset menjadi train dan test

X2_train, X2_test, y2_train, y2_test = train_test_split(X2, y2 , test_size=0.2, random_state=43)

non_categorical = ['age', 'home_distance', 'hourly_rate', 'monthly_income', 'companies_count','salary_increment_percentage','time_total_working',
       'last_year_training_time', 'time_current_role', 'time_last_promotion','time_current_manager']

scaler = MinMaxScaler()

X2_train_scaled = X2_train.copy()
X2_test_scaled = X2_test.copy()

X2_train_scaled[non_categorical] = scaler.fit_transform(X2_train_scaled[non_categorical])
X2_test_scaled[non_categorical] = scaler.transform(X2_test_scaled[non_categorical])
```

Melakukan hal yang sama sampai optimisasi seperti di nomor 2.A. , hanya saja kali ini feature targetnya adalah 'time_current_company'

RESULT

Kami menggunakan linear regression karena cocok untuk memprediksi data yang memiliki nilai korelasi yang rendah. Linear Regression digunakan untuk model yang memiliki hubungan antara dependent variable dengan independent variable yang banyak (bukan untuk klasifikasi).

Linear Regression

```
[ ]  linearModel = LinearRegression()
     linearModel.fit(X2_train_scaled, y2_train)
     linear_prediction = linearModel.predict(X2_test_scaled)
```

```
[ ]  metrics(y2_test, linear_prediction)
```

```
MAE: 2.022176180353666
MSE: 7.191215380449957
RMSE: 2.681644156194098
R_squared: 0.8214517841657324
```

RESULT

Selain linear regression, untuk memprediksi model ini kami juga menggunakan lasso regression yang hasilnya lebih optimal dibanding linear regression.

Lasso Regression

```
lassoModel = Lasso(alpha=best_alpha)
lassoModel.fit(X2_train_scaled, y2_train)
lasso_prediction = lassoModel.predict(X2_test_scaled)
metrics(y2_test, lasso_prediction)
```

```
MAE: 1.6672751143055498
MSE: 6.541224772216793
RMSE: 2.557518212164383
R_squared: 0.837590177645723
```

RESULT

Pada lasso regression kami juga menentukan kolom (selected_columns) yang akan digunakan untuk Random Forest Regression

```
selected_columns = []

for i in range(len(X2_test_scaled.columns)):
    if(lassoModel.coef_[i] != 0):
        selected_columns.append(X2_test_scaled.columns[i])

selected_columns
```

```
['education',
 'score_environment',
 'gender',
 'score_contribution',
 'job_rank',
 'score_job_satisfication',
 'companies_count',
 'over_time',
 'score_work_relationahip',
 'time_total_working',
 'time_current_role',
 'time_last_promotion',
 'time_current_manager',
 'role_Sales Executive']
```

RESULT

selected_columns digunakan untuk memprediksi data dengan random forest regression. Hasil dari random forest regression lebih optimal dibandingkan lasso dan linear regression.

```
randomForestModel = RandomForestRegressor(random_state=43)
randomForestModel.fit(X2_train, y2_train)

prediction2 = randomForestModel.predict(X2_test)
metrics(y2_test, prediction2)
```

```
MAE: 1.0291666666666668
MSE: 3.628200000000001
RMSE: 1.9047834522590752
R_squared: 0.909916668821932
```

Random Forest Regression

RESULT

Kemudian kami melakukin tuning dan menggunakan best parameter untuk random forest regression namun masih lebih optimal random forest regression dibandingkan dengan tuning.

w/ tuning

Random Forest Regression

```
clf.best_params_
```

```
{'criterion': 'friedman_mse',
 'max_depth': None,
 'min_samples_leaf': 1,
 'min_samples_split': 7}
```

```
randomForestModel = RandomForestRegressor(criterion="friedman_mse", max_depth=None, min_sampl
randomForestModel.fit(X2_train, y2_train)

prediction2 = randomForestModel.predict(X2_test)
metrics(y2_test, prediction2)
```

```
MAE: 1.0085934875309877
MSE: 3.7053699857506346
RMSE: 1.9249337613930082
R_squared: 0.9080006417607498
```

B

BAGAIMANA HASIL PREDIKSI ANDA DAPAT MEMBANTU PERUSAHAAN DALAM MENGAMBIL KEPUTUSAN?

Perusahaan dapat merencanakan langkah preventif untuk mengurangi kemungkinan kehilangan karyawan yang berkualitas, seperti meningkatkan kompensasi dan benefit, meningkatkan kesempatan untuk karir dan pengembangan profesional, atau meningkatkan budaya perusahaan yang lebih positif.

Memiliki informasi tentang kemungkinan masa tinggal karyawan dapat membantu perusahaan untuk merencanakan kebutuhan rekrutmen di masa depan dan mengambil tindakan yang diperlukan untuk mengisi kekosongan yang mungkin terjadi.

Lakukan analisis cluster yang dapat terbentuk pada data karyawan. Deskripsikan karakteristik masing-masing cluster yang didapatkan!

C LAKUKAN ANALISIS CLUSTER YANG DAPAT TERBENTUK PADA DATA KARYAWAN. DESKRIPSIKAN KARAKTERISTIK MASING-MASING CLUSTER YANG DIDAPATKAN!

- Pertama-tama, kami memilih atribut untuk dilakukan clustering.

```
[ ]  from sklearn.cluster import KMeans
     from sklearn.metrics import silhouette_samples, silhouette_score
     from yellowbrick.cluster import SilhouetteVisualizer

[ ]  df_cluster = df_encoded.copy()

     X = df_cluster[['score_contribution', 'rate_performance']]
```

Pemilihan kedua atribut di atas berdasarkan keinginan kelompok kami yang ingin melakukan clustering dengan mengelompokkan karyawan berdasarkan kinerja karyawan tersebut. Kinerja karyawan tersebut dilihat dari 2 pandangan, yaitu berdasarkan kesadaran akan kontribusi dari karyawan itu sendiri (score_contribution) dan nilai performa yang sebenarnya terjadi (rate_performance).

C LAKUKAN ANALISIS CLUSTER YANG DAPAT TERBENTUK PADA DATA KARYAWAN. DESKRIPSIKAN KARAKTERISTIK MASING-MASING CLUSTER YANG DIDAPATKAN!

- Lalu, kami mencoba mencari nilai k yang paling optimal sebelum melakukan clustering.

```
[ ]  fig, ax = plt.subplots(3, 2, figsize=(20,10))
     for k in [2, 3, 4, 5, 6]:
         # Create KMeans instance for different number of clusters
         clusterer = KMeans(n_clusters = k)

         # Draw silhouette diagram
         q, mod = divmod(k, 2)
         visualizer = SilhouetteVisualizer(clusterer, colors = 'yellowbrick', ax = ax[q-1][mod])
         visualizer.fit(X)

         # Compute silhoutte score
         # This gives a perspective into the density and separation of the formed clusters
         cluster_labels = clusterer.fit_predict(X)
         silhouette_avg = silhouette_score(X, cluster_labels)
         print(
             "For n_clusters =",
             k,
             "The average silhouette_coefficient is :",
             silhouette_avg,
         )
```

Berdasarkan hasil perhitungan di atas, kami menemukan bahwa nilai k=2 lah yang menghasilkan hasil terbaik. Hal itu dikarenakan dari keseluruhan aspek seperti silhouette coefficient, fluktuasi ukuran yang mirip, dan ketebalan plotnya nilai k=2 menunjukkan hasil yang paling optimal.

C

LAKUKAN ANALISIS CLUSTER YANG DAPAT TERBENTUK PADA DATA KARYAWAN. DESKRIPSIKAN KARAKTERISTIK MASING-MASING CLUSTER YANG DIDAPATKAN!

- Selanjutnya, kami menggunakan K-Means Clustering untuk melakukan clustering dengan menggunakan nilai k=2. Lalu, mencoba mengambarkan pembagian cluster melalui penggambaran di scatter plot.

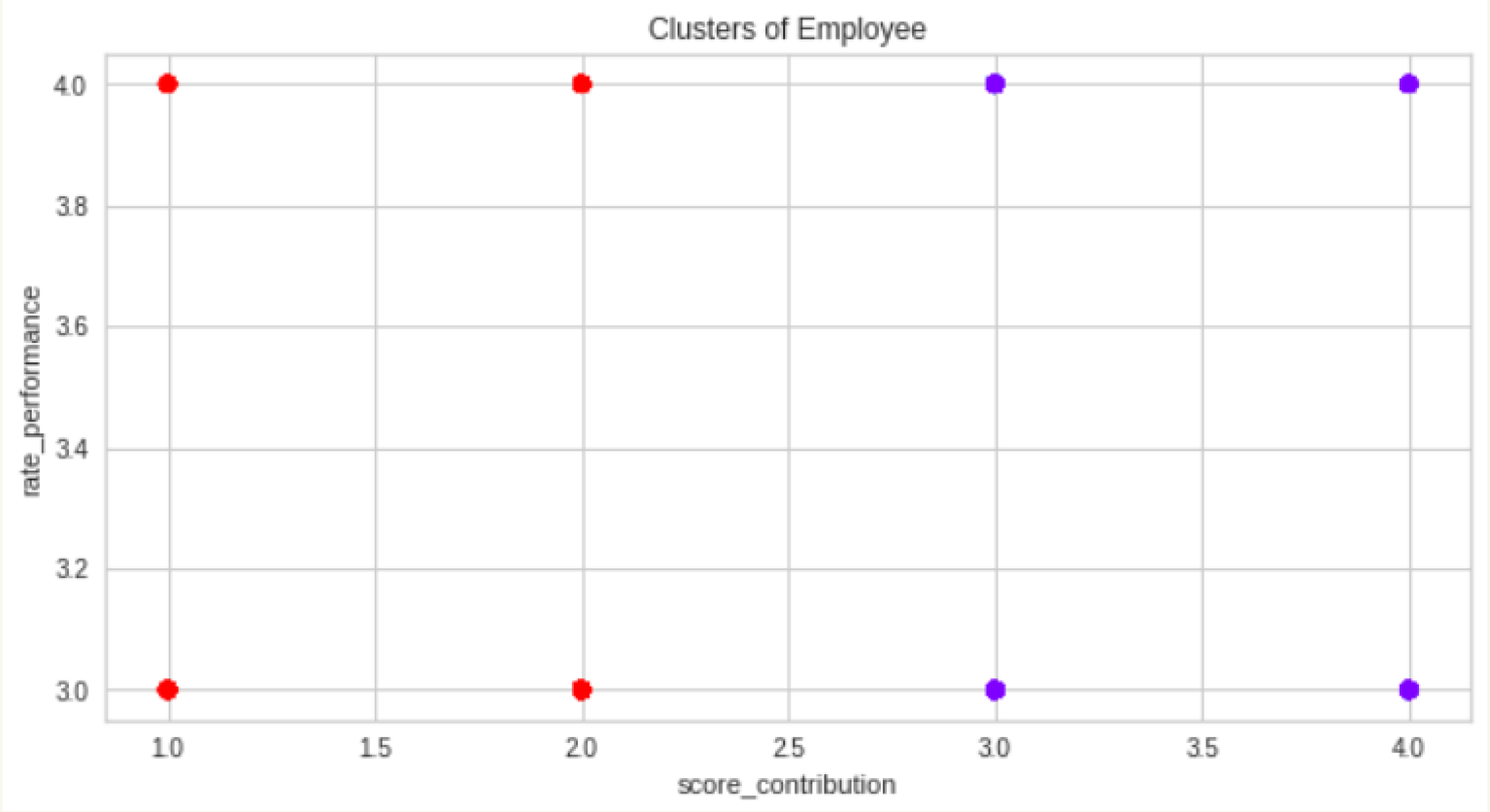
```
[ ]  kmeans = KMeans(n_clusters=2)
     cluster_assignment = kmeans.fit_predict(X)
     data_with_clusters = pd.DataFrame(X.copy(), columns=('score_contribution', 'rate_performance'))
     data_with_clusters['Clusters'] = cluster_assignment
```

```
[ ]  # Create figure
     fig = plt.figure(figsize = (10, 5))
     ax = plt.axes()

     # Prepare data
     x = data_with_clusters['score_contribution']
     y = data_with_clusters['rate_performance']
     cluster = data_with_clusters['Clusters']

     # Create plot
     ax.scatter(x, y, c = cluster, cmap = "rainbow")
     plt.title("Clusters of Employee")
     ax.set_xlabel('score_contribution')
     ax.set_ylabel('rate_performance')

     # Show plot
     plt.show()
```

RESULT

Hasil Clustering:

- Cluster 1 adalah cluster karyawan yang memiliki nilai kepuasan akan kontribusi yang diberikan kepada perusahaannya tergolong sangat rendah atau rendah, akan tetapi memiliki nilai performa yang tergolong tinggi dan sangat tinggi.
- Cluster 2 adalah cluster karyawan yang memiliki nilai kepuasan akan kontribusi yang diberikan kepada perusahaannya tergolong tinggi dan sangat tinggi yang juga sesuai dengan nilai performa yang tergolong tinggi dan sangat tinggi.



BE AI

THANK YOU!

Please let us know if you have any questions.