Menganalisis risiko peminjam gagal membayar

Sebagai kredit analyst proyek kami ialah menyiapkan laporan untuk bank bagian kredit. Kami mencari tahu pengaruh status perkawinan seorang nasabah dan jumlah anak terhadap probabilitas ketepatan waktu dalam melunasi pinjaman. Bank sudah memiliki beberapa data mengenai kelayakan kredit nasabah.

Laporan Anda akan dipertimbangkan pada saat membuat **penilaian kredit** untuk calon nasabah. **Penilaian kredit** digunakan untuk mengevaluasi kemampuan calon peminjam untuk melunasi pinjaman mereka.

Tujuan utama dari poject ini adalah untuk mengetahui kelayakan seorang klien untuk mendapatkan kredit berdasarkan status dan keadaan mereka yang tersimpan dalam data kita. Kita juga menguji kapasitas nasabah berdasarkan karakteristik mereka yang kita rangkum berdasarkan kategori-kategori sehingga diperoleh *pattern* untuk memberikan lampu kuning kepada nasabah yang masuk ke dalam kategori tertentu.

Hipotesis project :
1. Apakah terdapat korelasi antara jumlah anak dengan kemampuan melunasi pinjaman tepat waktu?
2. Apakah terdapat korelasi antara status keluarga dengan kemampuan melunasi pinjaman tepat waktu?
3. Apakah terdapat korelasi antara kelas ekonomi dengan kemampuan melunasi pinjaman tepat waktu?
4. Apakah terdapat korelasi antara tujuan kredit dengan kemampuan melunasi pinjaman tepat waktu?

Membuka *file* data dan menampilkan informasi umumnya.

Kita akan mulai dengan mengimport library dan memuat data.

```
# Memuat semua perpustakaan
import pandas as pd


# muat data
df = pd.read_csv('/datasets/credit_scoring_eng.csv')
```

Soal 1. Eksplorasi Data

Deskripsi Data

- `children` - jumlah anak dalam keluarga

- *days_employed* - pengalaman kerja dalam hari
- *dob_years* - usia klien dalam tahun
- *education* - pendidikan klien
- *education_id* - tanda pengenal pendidikan
- *family_status* - status perkawinan
- *family_status_id* - tanda pengenal status perkawinan
- *gender* - jenis kelamin klien
- *income_type* - jenis pekerjaan
- *debt* - apakah klien memiliki hutang pembayaran pinjaman
- *total_income* - pendapatan bulanan
- *purpose* - tujuan mendapatkan pinjaman

```
# Memeriksa jumlah baris dan kolom dalam dataset
df.shape
```

```
(21525, 12)
```

```
# Menampilkan 10 baris pertama dalam dataset
df.head(10)
```

	children	days_employed	dob_years	education	education_id	family_sta
0	1	-8437.673028	42	bachelor's degree	0	mai
1	1	-4024.803754	36	secondary education	1	mai
2	0	-5623.422610	33	Secondary Education	1	mai
3	3	-4124.747207	32	secondary education	1	mai
4	0	340266.072047	53	secondary education	1	civil partner
5	0	-926.185831	27	bachelor's degree	0	civil partner
6	0	-2879.202052	43	bachelor's degree	0	mai
7	0	-152.779569	50	SECONDARY EDUCATION	1	mai
8	2	-6929.865299	35	BACHELOR'S DEGREE	0	civil partner
9	0	-2188.756445	41	secondary education	1	mai

Dari sampel data yang ditampilkan, terdapat beberapa masalah yang bisa dideteksi yaitu tedapat value negatif dan value yang saya rasa sangat tinggi dari kolom `days_employed` yang saya rasa tidak masuk akal karena pada kolom menampilkan pengalaman kerja dalam hari , serta penulisan huruf kapital yang tidak tidak teratur pada kolom `education`.

```
# Mendapatkan informasi seluruh kolom dalam data
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21525 entries, 0 to 21524
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   children          21525 non-null  int64  
 1   days_employed     19351 non-null  float64
 2   dob_years         21525 non-null  int64  
 3   education         21525 non-null  object 
 4   education_id      21525 non-null  int64  
 5   family_status     21525 non-null  object 
 6   family_status_id  21525 non-null  int64  
 7   gender            21525 non-null  object 
 8   income_type       21525 non-null  object 
 9   debt              21525 non-null  int64  
 10  total_income      19351 non-null  float64
 11  purpose           21525 non-null  object 
dtypes: float64(2), int64(5), object(5)
memory usage: 2.0+ MB
```

Terdapat nilai yang hilang pada kolom `days_employed` dan `total_income`.

```
# Menampilkan nilai yang hilang dalam dataset
df[df['days_employed'].isna()].head(10)
```

	children	days_employed	dob_years	education	education_id	family_stat
12	0	NaN	65	secondary education	1	civil partners
26	0	NaN	41	secondary education	1	marr
29	0	NaN	63	secondary education	1	unmarr
41	0	NaN	50	secondary education	1	marr
55	0	NaN	54	secondary education	1	civil partners

Sejauh ini, dari yang saya lihat saya asumsikan bahwa value yang hilang memang tampak simetris karena berada pada baris yang sama, tapi untuk dapat menyimpulkan apakah nilai yang hilang memang bebar-benar berada pada baris yang sama perlu dilakukan investigasi lebih lanjut.

```
# Melakukan beberapa pemfilteran untuk mengetahui jumlah baris dari value yang hila
df_filtered_nan = df[df['days_employed'].isna()]
df_filtered_nan = df_filtered_nan[df_filtered_nan['total_income'].isna()]
df_filtered_nan.shape[0]
```

```
2174
```

Data yang haliang dalam dataset kita berjumlah `2174` baris.

```
# Sekarang kta akan menghitung rasio value yang hilang dari seluruh dataframe
df_distribution_nan = df_filtered_nan.shape[0] / df.shape[0]
print(f'Distribusi nilai yang hilang sebesar: {df_distribution_nan:0%}')
```

```
Distribusi nilai yang hilang sebesar: 10.099884%
```

Kesimpulan menengah

Kita bisa simpulkan jumlah nilai hilang sama dengan jumlah tabel yang difilter. Artinya nilai yang hilang dari tabel yang difilter simetris.

Persentase data yang hilang dari dataframe sebesar `10%`, cukup berpengaruh terhadap data kita bukan?

Selanjutnya saya akan menghitung jumlah persentase dari value yang hilang apakah memiliki dampak yang signifikan terhadap data dalam dataset, sehingga kita akan mengetahui langkah yang tepat untuk memproses data yang hilang tersebut.

```
# Menerapkan filter untuk menampilkan baris yang hilang dari data
df_filtered_nan.head(10)
```

	children	days_employed	dob_years	education	education_id	family_stat
12	0	NaN	65	secondary education	1	civil partners
26	0	NaN	41	secondary education	1	marr
29	0	NaN	63	secondary education	1	unmarr
41	0	NaN	50	secondary education	1	marr
55	0	NaN	54	secondary education	1	civil partners
65	0	NaN	21	secondary education	1	unmarr
67	0	NaN	52	bachelor's degree	0	marr
72	1	NaN	32	bachelor's degree	0	marr
82	2	NaN	50	bachelor's degree	0	marr
83	0	NaN	52	secondary education	1	marr

Value yang hilang tampak simetris dari table yang ditampilkan.

```
# Memeriksa distribusi
print(df_filtered_nan['income_type'].value_counts())
print()
df_filtered_nan['income_type'].value_counts() / df['income_type'].value_counts() *
```

```
employee         1105
business          508
retiree           413
civil servant     147
entrepreneur        1
Name: income_type, dtype: int64

business                        9.990167
```

```
civil servant                 10.075394
employee                       9.937944
entrepreneur                  50.000000
paternity / maternity leave         NaN
retiree                       10.710581
student                             NaN
unemployed                          NaN
Name: income_type, dtype: float64
```

Cukup terpola disini tetapi memang ada beberapa kategori yang valuenya tidak bisa dikatakan valid untuk dilakukan perhitungan mengingat jumlahnya hanya sedikit sekitar `1 - 2` data saja jumlahnya seperti kategori `unemployed, paternity / maternity leave, student,` dan `entrepreneur`. Secara overall data yang hilang terdistribusi sebesar `10%` pada setiap kategori.

Kemungkinan penyebab hilangnya nilai dalam data

Belum dapat ditentukan penyebab nilai yang hilang, kita harus mempertimbangkan kemungkinan-kemungkinan lain penyebab nilai yang hilai apakah nilai yang memiliki karakteristik tertentu seperti dari klien yang sudah menikah, jumlah anak, ataupun klien yang memiliki tunggakan pembayaran kredit.

```
# Memeriksa distribusi di seluruh dataset
df.isna().sum() / df.shape[0] * 100
```

```
children             0.000000
days_employed       10.099884
dob_years            0.000000
education            0.000000
education_id         0.000000
family_status        0.000000
family_status_id     0.000000
gender               0.000000
income_type          0.000000
debt                 0.000000
total_income        10.099884
purpose              0.000000
dtype: float64
```

Kesimpulan menengah

Jumlah nilai yang hilang dalam dataset mirip dengan tabel yang difilter.
Dan jumlah data yang hilang pada kolom `days_employed` sama dengan `total_income` yang artinya error hanya terjadi pada dua kolom tersebut.

Hal ini menimbulkan pertanyaan apakah nilai yang hilang membentuk suatu pola atau terjadi secara acak?
Untuk mejawab pertannyaan ini mari kita lakukan analisa lebih lanjut.

```
# Memeriksa apakah ada pola lain yang menyebabkan hilangnya data dari kolom 'famil
print(df_filtered_nan['family_status'].value_counts())
```

```
print()
df_filtered_nan['family_status'].value_counts() / df['family_status'].value_counts
```

```
married              1237
civil partnership     442
unmarried             288
divorced              112
widow / widower        95
Name: family_status, dtype: int64

married               9.991922
civil partnership    10.581757
unmarried            10.238180
divorced              9.372385
widow / widower       9.895833
Name: family_status, dtype: float64
```

Kesimpulan menengah

Cukup menarik bahwa nilai yang hilang terdistribusi secara merata dari kategori di kolom `family_status` yaitu sekitar `10%`. Tetapi juga dikatahui tidak ada ketegori khusus yang mengakibatkan data hilang disebabkan oleh salah satu kategori saja.

```
# Check for relation both 'gender' and missing value
print(df_filtered_nan['gender'].value_counts())
print()
df_filtered_nan['gender'].value_counts() / df['gender'].value_counts() * 100
```

```
F    1484
M     690
Name: gender, dtype: int64

F      10.424276
M       9.467618
XNA          NaN
Name: gender, dtype: float64
```

Pada kategori gender data yang hilang masing-masing terdistribusi sebesar `9%` dan `10%` artinya data yang hilang tidak hanya terdapat pada satu kategori saja.

```
# Memeriksa pendapatan dari beberapa pekerjaan yang kemungkinan tidak memiliki 'in
df[df['income_type'].isin(['unemployed', 'paternity / maternity leave', 'student',
```

	children	days_employed	dob_years	education	education_id	family_
3133	1	337524.466835	31	secondary education	1	
5936	0	NaN	58	bachelor's degree	0	
9410	0	-578.751554	22	bachelor's degree	0	u
14798	0	395302.838654	45	Bachelor's Degree	0	civil pa

Dari tabel diatas kita bisa mendapatkan informasi bahwa nilai yang hilang tidak selalu diakibatkan oleh pekerjaan yang biasanya tidak memiliki penghasilan, mungkin ada beberapa alasan bagaimana cara mereka mendapatkan penghasilan meskipun tidak sedang memiliki pekerjaan yang regular. Seperti pada baris `3133` klien yang `unemployed` tetapi mengajukan pinjaman untuk membeli properti untuk desewakan, meskipun belum diketahui darimana penghasilannya, Klien `9410` seorang `student` apakah dia seorang pekerja part time atau memiliki *rich parents* tetapi cukup aneh mengingat kegunaannya tidak untuk melanjutkan pendidikan melainkan membangun properti. Selain itu di beberapa negara juga memilki peraturan bahwa `paternity / maternity leave` *still getting paid*.

Kesimpulan

Konklusi yang bisa kita ambil mengenai penyebab data yang hilang adalah *human error* karena data yang hilang tidak terjadi pada kategori tertentu saja melainkan pada beberapa kategori.

Dari beberapa pengujian yang sudah saya lakukan saya menemukan bahwa dari kolom `family_status` dan `income_type` saya menemukan bahwa setiap kategori dalam masing-masing kolom tersebut memiliki nilai yang hilang hampir sama di setiap kategori yang sekitar `10%` artinya nilai yang hilang terdistribusi secara merata di setiap kategori dan nilainya simetris dengan pengujian yang kita lakukan sebelumnya dengan mencari distribusi nilai yang hilang di seluruh dataset yang menghasilkan angka juga sebesar `10%`.

Untuk beberapa masalah seperti:
1. Untuk nilai yang hilang saya akan membuat beberapa kateghori berdasarkan usia untuk mencari rata-rata `days_employed` dan `total_income` yang akan digunakan untuk mengisi value yang hilang.
2. Untuk mengatasi register yang berbeda pada kolom `education` saya akan mengubah semua huruf menjadi `lower`.
3. Kita bisa melakukan `drop` untuk nilai duplikat.

Transformasi data

[ ] ↳ *52 sel tersembunyi*

Bekerja dengan nilai yang hilang

Saya memasukkan dictionary `numpy` untuk mempercepat pekerjaan saya yang akan digunakan untuk me-`replace` nilai `0` pada kolom `days_employed` setelah membuat beberapa kategori usia.

```
# Import dictionary
import numpy as np
```

Memperbaiki nilai yang hilang di `total_income`

[ ] ↳ *17 sel tersembunyi*

Memperbaiki nilai di `days_employed`

[ ] ↳ *10 sel tersembunyi*

Pengkategorian Data

Sepertinya saya menemukan hal menarik di kolom `purpose` yaitu banyak sekali peng-kategorian data yang saya rasa bisa kita sederhanakan menjadi lebih general, sehingga kita akan lebih mudah dalam melakukan investigasi dalam mengambil keputusan.

[ ] ↳ *13 sel tersembunyi*

Memeriksa Hipotesis

[ ] ↳ *15 sel tersembunyi*

Kesimpulan Umum

Kita telah melakukan proses *cleansing data* untuk memperbaiki data-data yang bermasalah dalam dataset kita. Pembersihan yang kita lakukan meliputi mengisi value yang hilang, menghapus nilai duplikat, memperbaiki register yang tak beraturan, nilai yang terlalu besar, hingga mengganti nilai yang tidak wajar, sehingga kita mendapati dataset yang dapat kita olah untuk proses analisa kredit.

Temuan yang kita dapatkan setelah melakukan beberapa eksplorasi kita mendapati bahwa terdapat korelasi antara jumlah anak dan status perkawinan dalam risiko pemayaran kredit, klien yang tidak memiliki anak akan lebih mudah dalam melunasi hutangnya dibandingkan dengan klien yang memiliki anak. Klien yang menikah atau pernah memiliki pasangan memiliki risiko lebih rendah gagal bayar daripada klien dengan status *single* maupun tinggal bersama. Klien yang memiliki penghasilan lebih rendah akan lebih tinggi untuk memiliki hutang pinjaman, dan klien yang menggunakan uangnya untuk keperluan rumah akan lebih besar persentase mereka untuk dapat melunasi hutangnya.

Tetapi apakah semua manipulasi data yang kita lakukan dapat kita gunakan dalam proses *decision making* sehingga akan meminimalisir risiko yang akan terjadi di kemudian hari?