

Analisis Perbandingan Prediksi Kampanye Pemasaran Deposit Berjangka Menggunakan Metode Logistic Regression dan Random Forest

Syenira Sheila Febrianti¹

¹Sistem dan Teknologi Informasi, Fakultas Teknik, Universitas Negeri Jakarta, Jakarta, Indonesia

Article Info	RINGKASAN
Article history: 01 Desember 2022	Salah satu investasi yang banyak diminati oleh masyarakat, terutama masyarakat yang merupakan nasabah suatu bank yaitu investasi deposito berjangka. Deposito berjangka merupakan salah satu produk yang ditawarkan oleh bank dan merupakan sumber pendapatan utama bagi suatu bank. Dalam melakukan kampanye pemasaran terkait deposit berjangka kepada nasabahnya, bank melakukan beberapa cara yaitu melalui pemasaran melalui email, iklan, pemasaran melalui telepon, dan pemasaran digital. Pentingnya prediksi awal keberhasilan kampanye pemasaran ini dikarenakan efektifitas partisipasi nasabah berperan besar dalam keberhasilan kampanye pemasaran. Penelitian ini bertujuan untuk menganalisis perbandingan prediksi kampanye pemasaran deposito berjangka menggunakan metode Logistic Regression dan Random Forest sehingga nasabah yang berpotensi untuk membuat deposito berjangka dapat diidentifikasi dan kemungkinan tingkat keberhasilan kampanye pemasaran lebih besar. Kedua metode tersebut juga dipadukan dengan metode <i>oversampling</i> dan <i>undersampling</i> dikarenakan dataset yang digunakan dalam penelitian tidak seimbang. Hasil yang diperoleh melalui penelitian ini yaitu metode klasifikasi Logistic Regression pada dataset yang tidak seimbang merupakan model klasifikasi terbaik dengan nilai <i>precision</i> yaitu 65.49% dan <i>recall</i> 30.96% yang mana nilai <i>precision</i> lebih diutamakan dibandingkan <i>recall</i> sebab pengukuran ini dipertimbangkan untuk memprediksi kelas minoritas. Namun, model dengan peningkatan yang cukup besar pada nilai <i>recall</i> yang dihasilkan dan memiliki <i>precision</i> yang cukup baik yaitu Random Forest dengan dataset seimbang yang didapatkan melalui <i>oversampling</i> dengan nilai <i>precision</i> 59.08% dan nilai <i>recall</i> 50.76%.
Keywords: Deposito berjangka Kampanye pemasaran Logistic regression Random Forest	

1. PENDAHULUAN

Investasi, dijelaskan dalam Kamus Besar Bahasa Indonesia (KBBI) dapat diartikan sebagai uang atau modal yang ditanamkan pada suatu perusahaan atau proyek demi keuntungan atau manfaat yang akan datang[1]. Investasi juga dapat diartikan sebagai kegiatan penanaman dana atau aset berharga lainnya pada instrumen tertentu dalam jangka waktu tertentu[2]. Berbagai jenis investasi yang dilakukan masyarakat saat ini telah meningkat cukup pesat. Hal ini didukung oleh pernyataan Kemenkeu yang menyatakan bahwa investasi di Indonesia tumbuh moderat pada triwulan II-2022 dari berbagai bidang[3]. Salah satu investasi yang banyak diminati oleh masyarakat, terutama masyarakat yang merupakan nasabah suatu bank yaitu investasi deposito berjangka. Deposito berjangka merupakan salah satu produk yang ditawarkan oleh bank dan merupakan sumber pendapatan utama bagi suatu bank. Tertuang dalam Undang-Undang Perbankan No.10 Tahun 1988 Pasal 1 : “Deposito merupakan simpanan yang penarikannya hanya dapat dilakukan pada waktu tertentu berdasarkan perjanjian nasabah penyimpan dengan Bank”[4]. Deposito berjangka adalah deposito yang diterbitkan menurut jangka waktu tertentu dengan tingkat bunga yang telah disepakati, sehingga dapat dikatakan bahwa deposito berjangka ini hanya dapat diambil sesuai waktu yang telah ditentukan antara nasabah dengan pihak bank dan jika salah satu pihak melanggar kesepakatan maka akan ada sanksi yang dikenakan.

Namun demikian, alasan banyaknya nasabah yang berinvestasi dalam deposito berjangka diantaranya yaitu bunga yang ditawarkan cukup besar, keamanan uang terjamin, resiko yang kecil dan akses

bunga yang mudah. Dalam melakukan kampanye pemasaran terkait deposit berjangka kepada nasabahnya, bank melakukan beberapa cara yaitu melalui pemasaran melalui email, iklan, pemasaran melalui telepon, dan pemasaran digital. Diantara beberapa cara tersebut, kampanye pemasaran melalui telepon masih tetap menjadi salah satu cara paling efektif untuk menjangkau orang. Pentingnya prediksi awal keberhasilan kampanye pemasaran ini dikarenakan efektifitas partisipasi nasabah berperan besar dalam keberhasilan kampanye pemasaran. Oleh sebab itu, sangat penting untuk mengidentifikasi nasabah yang berpotensi untuk membuat deposito berjangka dengan melakukan pengolahan data yang telah dimiliki bank sehingga kemungkinan tingkat keberhasilan kampanye pemasaran lebih besar. Hal ini tentu berpengaruh besar bagi keefisienan pihak bank dalam memprioritaskan nasabah yang harus dihubungi dan mengetahui apakah nasabah yang akan dihubungi berpotensi untuk membuat deposito berjangka atau tidak.

Dalam penelitian terdahulu telah banyak dilakukan penelitian terkait metode klasifikasi Logistic Regression dan Random Forest. Diantaranya penelitian yang dilakukan oleh Ichsan Firmansyah et al.[5] membandingkan metode klasifikasi Logistic Regression dan Random Forest dalam mengklasifikasi pasien positif dan negatif COVID-19 berdasarkan gejala yang dimiliki pasien. Hasil yang diperoleh melalui penelitian tersebut yaitu algoritma Random Forest memiliki akurasi, ketepatan dan sensitifitas yang lebih tinggi dibandingkan algoritma Logistic Regression. Pada penelitian [6] dilakukan klasifikasi diabetes menggunakan Logistic Regression dan didapatkan hasil yang cukup baik dalam akurasi dan ketepatan dengan nilai akurasi yang dicapai yaitu 75,97% dan presisi sebesar 76,92%. Dalam evaluasi Random Forest yang dilakukan dalam penelitian [7] menggunakan K-Fold Cross Validation, dengan nilai k sebesar 10 dicapai nilai akurasi sebesar 76,2% dan sebesar 77,3%. Penelitian [8] menunjukkan klasifikasi Random Forest memiliki nilai akurasi yang lebih tinggi dibandingkan nilai akurasi dari Logistic Regression dengan 71% benar dalam mengklasifikasi mode musik atau lagu. Nilai akurasi metode klasifikasi Random Forest tersebut berbanding lurus dengan banyaknya *tree* yang digunakan dalam klasifikasi.

Berdasarkan beberapa penelitian diatas, metode klasifikasi Logistic Regression dan Random Forest memiliki hasil yang baik. Penelitian ini bertujuan untuk menganalisis perbandingan dua metode dalam memprediksi keberhasilan kampanye pemasaran deposit berjangka yaitu metode klasifikasi Logistic Regression dan Random Forest untuk mengetahui metode klasifikasi mana yang memiliki performa terbaik dan hasil klasifikasi dilihat menggunakan menggunakan *confusion matrix* dan kurva ROC.

2. METODE

Metode yang digunakan dalam penelitian ini yaitu metode klasifikasi Logistic Regression dan Random Forest. Kedua metode tersebut dibandingkan oleh penulis menggunakan bahasa *python* di *Google Collabs*. Dua pendekatan utama yaitu pendekatan kualitatif dan pendekatan kuantitatif, digunakan dalam penelitian ini agar diperoleh pemahaman yang lebih mendalam dan hasil pengujian yang maksimal. Pendekatan kualitatif digunakan dalam “*grounded theory research*” yang mana pengertian dan konsep yang ada dikembangkan menjadi sebuah teori. Data yang digunakan bersifat deskriptif yang berupa gejala-gejala yang ditafsir menurut budaya yang bersangkutan dengan mencari makna semantis universal dari gejala yang sedang diteliti. Sedangkan, pendekatan kuantitatif digunakan dalam pengujian teori, memberikan deskripsi statistik, serta menunjukkan fakta dan hubungan antar variabel. Penekanan dalam pendekatan kuantitatif ini merupakan hal-hal yang bersifat kongkrit, uji empiris dan fakta yang nyata dalam menaksir hasil pengujian[9].

Data Mining

Data mining digunakan untuk memperoleh pengetahuan tersembunyi dalam suatu database. Data mining juga merupakan sebuah proses mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar dengan memanfaatkan teknik statistik, matematika, kecedasan buatan dan pembelajaran mesin[10]. Istilah lain dari data mining yaitu *Knowledge Discovery in Database* (KDD), yang mana merupakan proses pengumpulan, pemakaian data historis yang digunakan dalam menemukan keteraturan, pola atau hubungan yang cenderung tidak disadari keberadaannya dalam suatu dataset berukuran besar. Hasil dari data mining dapat digunakan untuk memperbaiki pengambilan keputusan di masa yang akan datang[11].

Metode penelitian ini menggunakan model proses *Cross Industry Standard Process for Data Mining* (CRISP-DM) yang mana terdiri dari beberapa tahapan sebagai berikut.

1. *Bussiness Understanding*

Tahapan ini merupakan tahap penentuan tujuan penelitian yaitu menganalisis perbandingan prediksi kampanye pemasaran deposit berjangka menggunakan metode Logistic Regression dan Random Forest untuk mengetahui performa metode klasifikasi mana yang memiliki performa terbaik dalam memprediksi potensi nasabah yang berlangganan deposito berjangka.

2. Data Understanding

Dataset yang digunakan dalam penelitian ini merupakan data kampanye pemasaran deposito berjangka melalui telepon yang dilakukan oleh suatu bank. Penelitian ini menggunakan dua dataset yaitu *train.csv* dan *test.csv*. Dataset *train.csv* terdiri dari 45.211 baris dengan 18 kolom dan dataset *test.csv* terdiri dari 4.521 baris dengan 18 kolom. Dataset terdiri dari 16 atribut fitur dan 1 atribut label.

Tabel 1. Atribut dan Nilai Kategori
Dalam Indikator Kampanye Pemasaran Deposito Berjangka

Atribut	Keterangan
y	Variabel <i>output</i> yang mana merupakan atribut yang mengindikasikan apakah nasabah mengambil deposit berjangka atau tidak (kategorikal : 'yes', 'no')

Atribut	Keterangan
Age	Atribut yang mengindikasikan usia nasabah. Usia subjek disesuaikan dengan nasabah (numerikal)
Job	Atribut yang mengindikasikan pekerjaan nasabah (kategorikal : 'admin.', 'unknown', 'unemployed', 'management', 'housemaid', 'entrepreneur', 'student', 'bluecollar', 'selfemployed', 'retired', 'technician', 'services')
Marital	Atribut yang mengindikasikan status perkawinan nasabah (kategorikal : 'divorced', 'married', 'single')
Education	Atribut yang mengindikasikan pendidikan terakhir nasabah (kategorikal : 'unknown', 'secondary', 'primary', 'tertiary')
Default	Atribut yang menyatakan apakah nasabah memiliki kredit dalam default atau tidak (kategorikal : 'yes', 'no')
Balance	Atribut yang menyatakan rata-rata saldo tahunan nasabah (numerikal)
Housing	Atribut yang mengindikasikan apakah nasabah memiliki pinjaman perumahan atau tidak (kategorikal : 'yes', 'no')
Loan	Atribut yang mengindikasikan apakah nasabah memiliki pinjaman pribadi atau tidak (kategorikal : 'yes', 'no')
Contact	Atribut yang menyatakan jenis komunikasi kontak (kategorikal : 'unknown', 'telephone', 'cellular')
Day	Atribut yang menyatakan hari kontak terakhir dalam sebulan (kategorikal : 'mon', 'tue', ..., 'fri')
Month	Atribut yang menyatakan bulan kontak terakhir tahun ini (kategorikal : 'jan', 'feb', 'mar', ..., 'nov', 'dec')
Duration	Atribut yang menyatakan durasi kontak terakhir (numerik : detik)
Campaign	Atribut yang menyatakan jumlah kontak yang dilakukan selama kampanye pemasaran ini dan untuk nasabah ini (numerik : termasuk kontak terakhir)
Pdays	Atribut yang menyatakan jumlah hari yang berlalu setelah nasabah terakhir dihubungi dari kampanye pemasaran sebelumnya (numerik : - 1 berarti klien sebelumnya tidak dihubungi)
Previous	Atribut yang menyatakan jumlah kontak yang dilakukan sebelum kampanye pemasaran ini dan untuk nasabah ini (numerik)
Poutcome	Atribut yang menyatakan hasil dari kampanye pemasaran sebelumnya (kategorikal : 'unknown', 'other', 'failure', 'success')

3. Data Preparation

Pada tahap ini, ada beberapa tahapan yang dilakukan dalam melakukan pengolahan dataset kampanye pemasaran deposito berjangka yaitu, *data preparation* merupakan proses mempersiapkan data termasuk

didalamnya mengecek data yang memiliki nilai kosong dan mengecek data yang memiliki nilai duplikat; *data cleaning* yang mana merupakan proses pembersihan data yang memiliki nilai yang kosong dan menghilangkan *noise* pada data; dan *data reduction* merupakan proses pengurangan jumlah atribut yang kurang sesuai dengan kelas yang akan diprediksi atau pemilihan atribut yang tepat sesuai dengan syarat atribut penentu (kelas). Dalam penelitian ini, data *training* yang digunakan yaitu sebanyak 60% dari total jumlah data dan data *testing* sebanyak 40% dari total jumlah data.

4. *Modelling*

Setelah dilakukan *data preparation*, tahap selanjutnya yaitu melakukan proses pemodelan menggunakan *software Google Colab* dan bahasa pemrograman *python* menggunakan metode klasifikasi Logistic Regression dan Random Forest.

5. *Evaluation*

Pola pemodelan yang telah terbentuk dari setiap metode klasifikasi yang digunakan, dilakukan proses evaluasi pada tahap ini menggunakan *confusion matrix* sehingga diperoleh keakuratan informasi model. Tahap evaluasi ini juga dinyatakan sebagai tahap klasifikasi sebab pada setiap model algoritma dilakukan pengujian nilai akurasi dan proses evaluasi juga ditunjukkan melalui kurva ROC (AUC). Pada tahap ini, dikarenakan adanya ketidakseimbangan data didalam dataset yang digunakan maka untuk memaksimalkan nilai akurasi model dilakukan proses *oversampling* dan *undersampling*.

6. *Deployment*

Setelah proses pemodelan dan evaluasi dilakukan, maka tahap selanjutnya yaitu menentukan model klasifikasi yang memiliki nilai akurasi tertinggi diantara keenam jenis model yaitu Logistic Regression dengan dataset yang tidak seimbang, Logistic Regression dengan dataset seimbang melalui *oversampling*, Logistic Regression dengan dataset yang seimbang melalui *undersampling*, Random Forest dengan dataset yang tidak seimbang, Random Forest dengan dataset seimbang melalui *oversampling* dan Random Forest dengan dataset yang seimbang melalui *undersampling*. Berdasarkan hal tersebut, maka algoritma dengan nilai akurasi yang tinggi merupakan algoritma klasifikasi terbaik.

Logistic Regression

Logistic Regression merupakan algoritma yang menggunakan distribusi Bernoulli untuk pengamatan variabel respon, berupa kategori biner (0 dan 1) atau sifat dikotomik, yang dipengaruhi oleh variabel prediktor dalam sebuah model regresi pada proses klasifikasinya[5]. Dapat dikatakan algoritma ini digunakan untuk menganalisis data untuk menunjukkan hubungan antara variabel respon dengan variabel penjelasnya. Akhir-akhir ini algoritma Logistic Regression sering digunakan dalam melakukan klasifikasi[12].

Random Forest

Random Forest merupakan sebagai salah satu algoritma klasifikasi yang mudah dalam pengaplikasiannya, memiliki beban komputasi yang rendah namun tingkat akurasi yang tinggi. Algoritma ini mengaplikasikan *bagging* dan pemilihan fitur secara acak pada setiap iterasinya untuk membentuk pohon-pohon yang menghasilkan klasifikasi data dari pohon dengan nilai voting tertinggi[5]. Random Forest hanya dapat menunjukkan tingkat kepentingan variabel dan tidak dapat menentukan signifikansi variabel dikarenakan menggunakan ansambel Decision-Tree[12].

3. Hasil dan Pembahasan

3.1. Dataset

Penelitian ini menggunakan dataset kampanye pemasaran deposito berjangka melalui telepon yang dilakukan oleh suatu bank. Pengolahan data dilakukan menggunakan *software Google Colab* dan bahasa pemrograman *python*.

3.2. Data Pre-processing

A. Mengecek Data Kosong

Data kosong pada dataset kampanye pemasaran deposito berjangka ditangani dalam tahap pre-processing dengan memasukkan perintah isna yang ditunjukkan pada Gambar 1. Dalam dataset tersebut, seluruh atribut bernilai kosong sehingga dapat diambil kesimpulan bahwa tidak terdapat data kosong dalam dataset.

```
#Mengecek data yang kosong
print(train.isna().values.any())
print(train.isna().sum())

False
age          0
job          0
marital      0
education    0
default      0
balance      0
housing      0
loan         0
contact      0
day          0
month        0
duration     0
campaign     0
pdays       0
previous     0
poutcome     0
y            0
dtype: int64
```

Gambar 1. Potongan Source Code Cek Data yang Kosong

B. Mengecek Duplikasi Data

Selanjutnya, pada tahap pre-processing dilakukan pengecekan duplikasi data dengan memasukkan perintah menghitung jumlah record duplikat yang ditunjukkan pada Gambar 2. Hasil yang diperoleh yaitu tidak terdapat duplikasi data pada dataset tersebut.

```
#Mengecek duplikasi data
print(train.duplicated().value_counts())

False    45211
dtype: int64
```

Gambar 2. Potongan Source Code Cek Duplikasi Data

C. Mengecek Kesalahan Pengejaan Nama Kolom

Tahap berikutnya yaitu dilakukan pengecekan pengejaan nama kolom yang ditunjukkan pada Gambar 3. Hasil yang diperoleh yaitu tidak terdapat kesalahan pengejaan nama kolom pada dataset tersebut.

```
#Mengecek kesalahan pengejaan nama kolom
train.columns

Index(['age', 'job', 'marital', 'education', 'default', 'balance', 'housing',
       'loan', 'contact', 'day', 'month', 'duration', 'campaign', 'pdays',
       'previous', 'poutcome', 'y'],
      dtype='object')
```

Gambar 3. Potongan Source Code Cek Pengejaan Nama Kolom

D. Menghilangkan White space

Selanjutnya, yaitu menghilangkan *whitespace* dalam dataset. *White space* dihilangkan sebab dianggap tidak memiliki pengaruh apapun.

```
#Menghilangkan white space
train.columns = train.columns.str.replace(' ', '')
```

Gambar 3. Potongan Source Code Cek Pengejaan Nama Kolom

E. Mengecek Nilai Konsistensi

Pengecekan nilai konsistensi setiap fitur dalam dataset ditunjukkan pada Gambar 3. Hasil yang didapatkan yaitu nilai *unknown* memiliki proporsi yang insignifikan.

```
#Konsistensi value
job_status = train["job"]
print(job_status.value_counts())

blue-collar    9732
management    9458
technician     7597
admin.         5171
services       4154
retired        2264
self-employed  1579
entrepreneur   1487
unemployed     1303
housemaid      1240
student        938
unknown        288
Name: job, dtype: int64

marriage_status = train["marital"]
print(marriage_status.value_counts())

married    27214
single     12790
divorced    5207
```

Gambar 5. Potongan Source Code Cek Nilai Konsistensi

F. Koefisien Korelasi Pearson

Penggunaan Korelasi Pearson adalah untuk mengetahui tingkat hubungan antara dua variabel numerik berskala interval untuk mencegah terjadinya *overfitting* yang ditunjukkan pada Gambar 6.

```
#Mengecek korelasi variabel untuk mencegah overfitting
print(train.corr())
```

	age	balance	day	duration	campaign	pdays	previous
age	1.000000	0.097783	-0.009120	-0.004648	0.004760	-0.023758	0.001288
balance	0.097783	1.000000	0.004503	0.021560	-0.014578	0.003435	0.016674
day	-0.009120	0.004503	1.000000	-0.030206	0.162490	-0.093044	-0.051710
duration	-0.004648	0.021560	-0.030206	1.000000	-0.084570	-0.001565	0.001203
campaign	0.004760	-0.014578	0.162490	-0.084570	1.000000	-0.088628	-0.032855
pdays	-0.023758	0.003435	-0.093044	-0.001565	-0.088628	1.000000	0.454820
previous	0.001288	0.016674	-0.051710	0.001203	-0.032855	0.454820	1.000000

Gambar 6. Potongan Source Code Koefisien Korelasi Pearson

G. Menghilangkan Noise

Pada tahap ini, dilakukan penghapusan beberapa fitur dikarenakan tidak memiliki pengaruh terhadap dataset. Penghapusan dilakukan melalui dua cara, yaitu metode *replace* untuk variabel *unknown* pada fitur *job*, *education*, dan *contact*; dan *drop features* pada fitur *month*, *previous*, *day* dan *pdays*. Penghapusan fitur diputuskan berdasarkan hasil eksplorasi data melalui grafik. Perintah untuk melakukan penghapusan tersebut ditunjukkan pada Gambar 7 dan 8.

```
#Menghilangkan noise
#Metode replace
train_test_concat["job"].replace(["unknown"],train_test_concat["job"].mode(),inplace = True)
train_test_concat["education"].replace(["unknown"],train_test_concat["education"].mode(),inplace = True)
train_test_concat["contact"].replace(["unknown"],train_test_concat["contact"].mode(),inplace = True)
```

Gambar 7. Potongan Source Code Menghilangkan Noise

```
#Drop fitur
train_test_concat.drop(columns = ["month", "previous", "day", "pdays"], inplace = True)
print(train_test_concat)
```

Gambar 8. Potongan Source Code Drop Fitur

H. Konversi Tipe Atribut Kategorikal menjadi Numerik

Pada dataset yang digunakan dalam penelitian ini, terdapat atribut dengan tipe kategorikal yang mana perlu dilakukan pengkonversian menjadi tipe numerik agar data dapat diproses menggunakan *python*. Pengkonversian dilakukan dengan *encoding* data kategorikal kedalam

numerik. Jika fitur memiliki dua variabel 'yes' atau 'no' maka akan dikonversi menjadi 1 dan 0, jika fitur memiliki lebih dari dua variabel seperti pada fitur *marital* yang memiliki variabel 'married', 'single', 'divorced' maka akan dikonversi menjadi [1,0,0], [0,1,0], [0,0,1] yang ditunjukkan pada Gambar 9.

```
#Encoding Fitur kategorikal
train_test_concat['default'] = train_test_concat['default'].map({'yes': 1, 'no': 0})
train_test_concat['housing'] = train_test_concat['housing'].map({'yes': 1, 'no': 0})
train_test_concat['loan'] = train_test_concat['loan'].map({'yes': 1, 'no': 0})
train_test_concat['contact'] = train_test_concat['contact'].map({'telephone': 1, 'cellular': 0})
train_test_concat['y'] = train_test_concat['y'].map({'yes': 1, 'no': 0})
train_test_concat

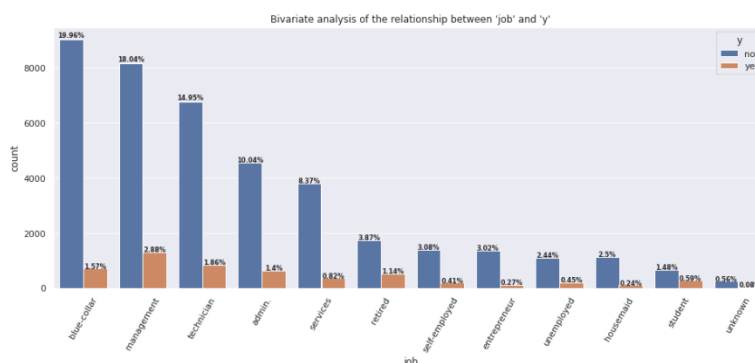
train_test_concat = pd.get_dummies(train_test_concat, columns=['job', 'marital', 'education', 'poutcome'])
train_test_concat
```

Gambar 9. Potongan Source Code Encoding Fitur Kategorikal

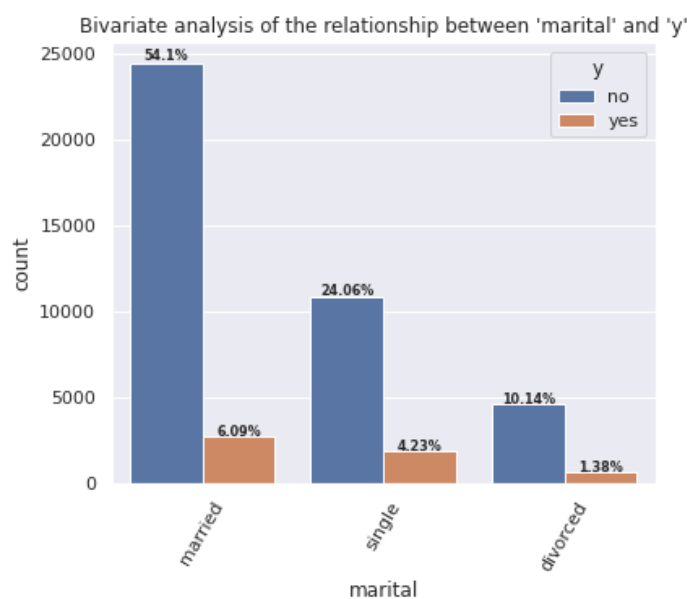
3.3. Eksplorasi Data

Eksplorasi data dilakukan dengan tujuan agar dapat memahami dataset yang digunakan dalam penelitian ini. Informasi mengenai dataset didapatkan melalui penyajian informasi atribut dalam bentuk grafik. Fitur kategorikal divisualisasi menggunakan *bar charts* dengan tujuan untuk menganalisis frekuensi berdasarkan kategori dan fitur numerikal divisualisasi menggunakan *box plot* untuk mendeteksi *outliers*.

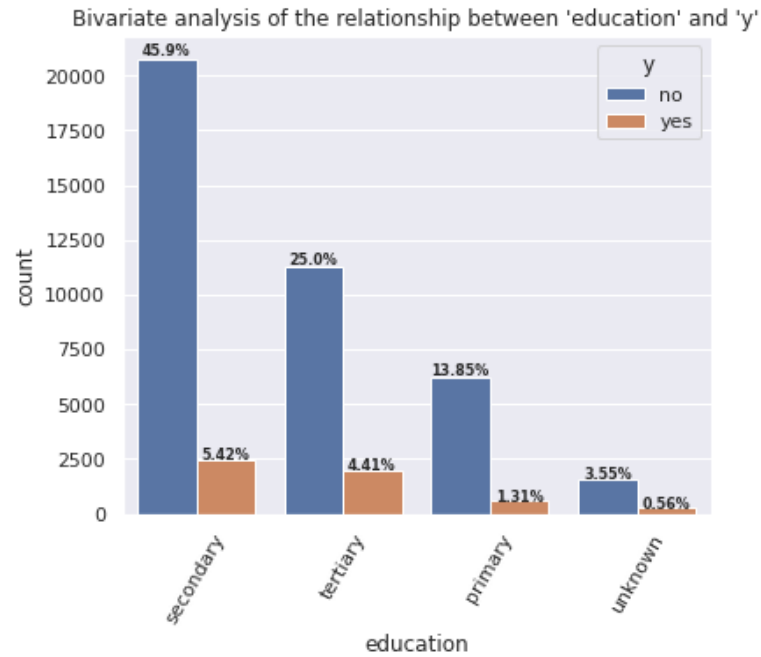
A. Fitur Kategorikal (*job*, *marital*, *education*, *default*, *housing*, *loan*, *poutcome*, *y*)



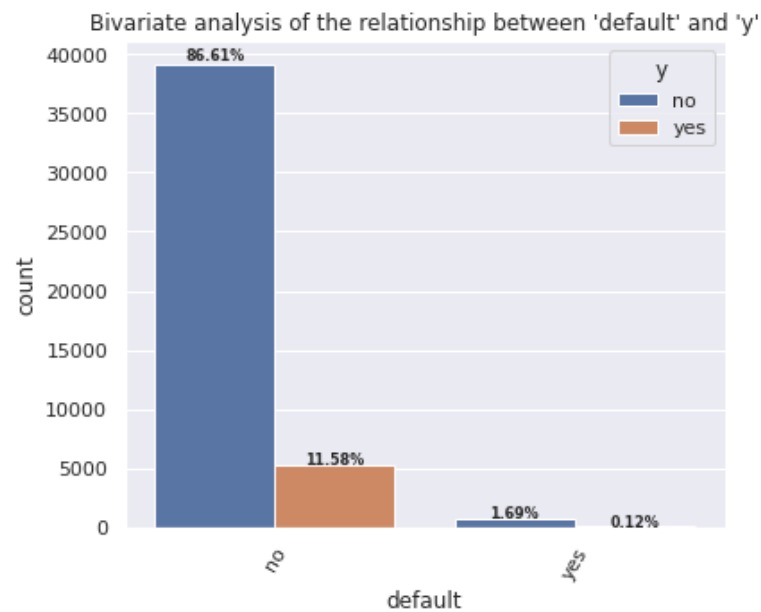
Gambar 10. Grafik Hubungan Pekerjaan dengan Nasabah yang Berlangganan Deposit Berjangka



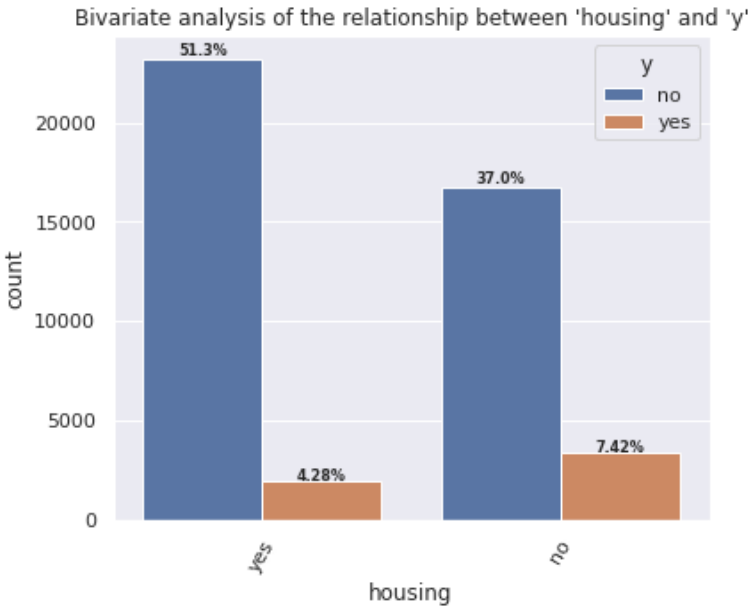
Gambar 11. Grafik Hubungan Status Pekawinan dengan Nasabah yang Berlangganan Deposit Berjangka



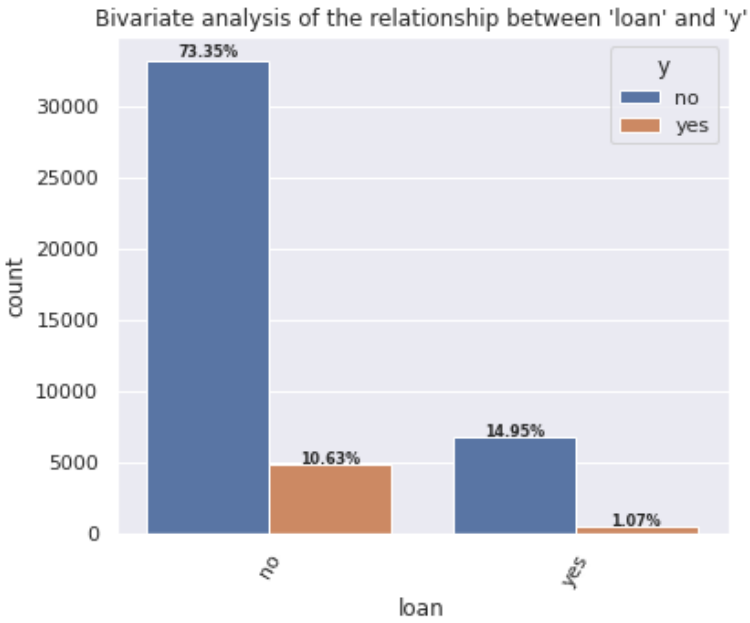
Gambar 10. Grafik Hubungan Pendidikan dengan Nasabah yang Berlangganan Deposit Berjangka



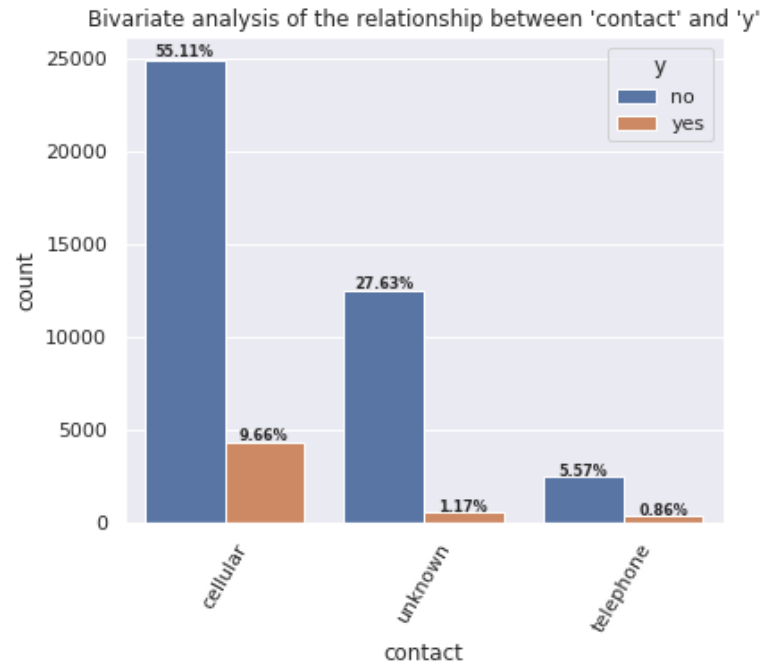
Gambar 11. Grafik Hubungan Kredit Default dengan Nasabah yang Berlangganan Deposit Berjangka



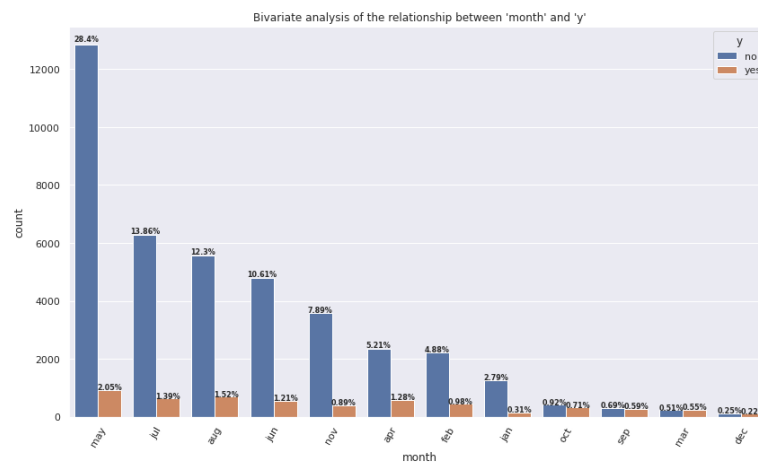
Gambar 12. Grafik Hubungan Pinjaman Perumahan dengan Nasabah yang Berlangganan Deposit Berjangka



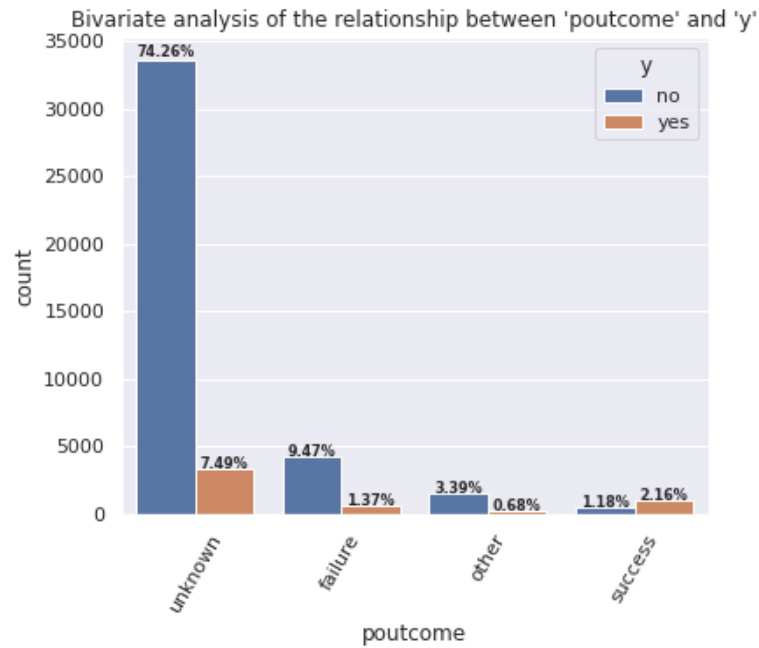
Gambar 12. Grafik Hubungan Pinjaman dengan Nasabah yang Berlangganan Deposit Berjangka



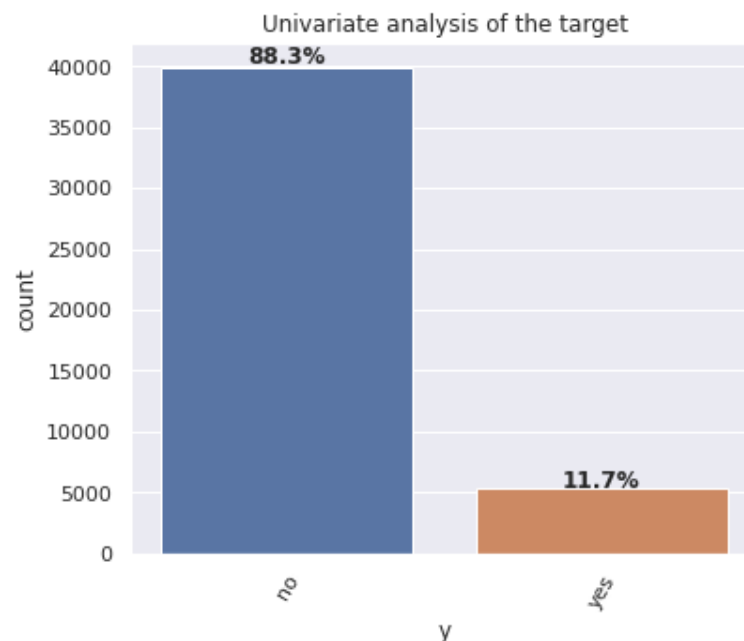
Gambar 13. Grafik Hubungan Jenis Kontak dengan Nasabah yang Berlangganan Deposit Berjangka



Gambar 14. Grafik Hubungan Bulan Kontak Terakhir dengan Nasabah yang Berlangganan Deposit Berjangka



Gambar 15. Grafik Hubungan Pinjaman Perumahan dengan Nasabah yang Berlangganan Deposit Berjangka



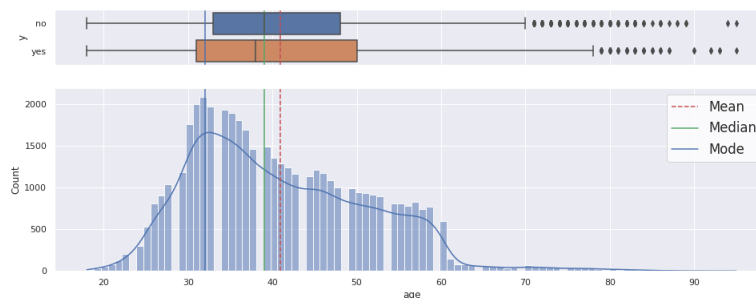
Gambar 16. Grafik Perbandingan Nasabah yang Berlangganan Deposit Berjangka dengan yang Tidak Berlangganan

Berdasarkan grafik fitur kategorikal diatas, dapat diambil kesimpulan bahwa :

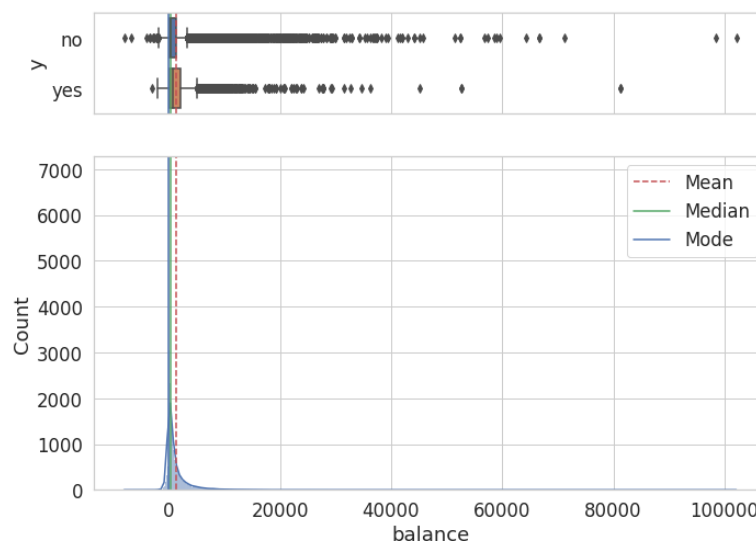
- Panggilan yang dilakukan bank sebagian besar melayani nasabah yang memiliki pekerjaan kasar (*blue-collar worker*), tetapi nasabah yang berlangganan deposit berjangka banyak yang berasal dari pekerjaan manajerial.
- Sebagian besar panggilan yang dilakukan bank melayani nasabah yang telah menikah dengan jumlah keberhasilan terbanyak juga. Namun rasio keberhasilan terbaik relatif terhadap jumlah total nasabah yang belum menikah.
- Sebagian besar nasabah dengan pendidikan yang tinggi berlangganan deposito berjangka.
- Sebagian besar nasabah memiliki kredit *default*. Hal ini lumrah sebab nasabah yang memiliki kredit tentu tidak ingin berlangganan penawaran bank yang baru.

- Panggilan yang bank lakukan lebih banyak masuk kepada nasabah dengan pinjaman perumahan, namun lebih banyak keberhasilan dalam hal jumlah dan rasio relatif terhadap jumlah nasabah yang tidak memiliki pinjaman perumahan.
- Lebih banyak panggilan masuk kepada nasabah tanpa pinjaman pribadi dengan jumlah keberhasilan terbanyak juga dibandingkan nasabah yang tidak memiliki pinjaman pribadi.
- Sebagian besar panggilan dilakukan bank melalui kontak seluler. Terdapat juga banyak panggilan yang dilakukan dari tipe kontak lain yang tidak diketahui.
- Sebagian besar panggilan dilakukan selama musim panas hingga musim gugur (Mei hingga Agustus). Namun, rasio terbaik dari kampanye pemasaran deposito berjangka yang berhasil dalam hal jumlah klien dalam bulan tertentu adalah pada bulan Oktober, September, Desember, dan Maret. Kemungkinan pada bulan tersebut sesuai dengan waktu tertentu yang mana nasabah lebih mudah diakses atau ada situasi tertentu pada bulan tersebut yang memungkinkan nasabah membayar lebih banyak deposito berjangka. Fitur ini akan dihilangkan dikarenakan tidak mempengaruhi hasil.
- Sebagian besar hasil dari kampanye pemasaran sebelumnya tidak mengetahui mengenai kampanye pemasaran deposito berjangka. Namun, keberhasilan dari kampanye sebelumnya memiliki rasio relatif yang baik.
- Sebagian besar nasabah yang dihubungi oleh bank tidak ingin berlangganan deposito berjangka. Hanya 11.7% nasabah yang ingin berlangganan deposito berjangka. Berdasarkan grafik tersebut, terlihat adanya ketidakseimbangan data.

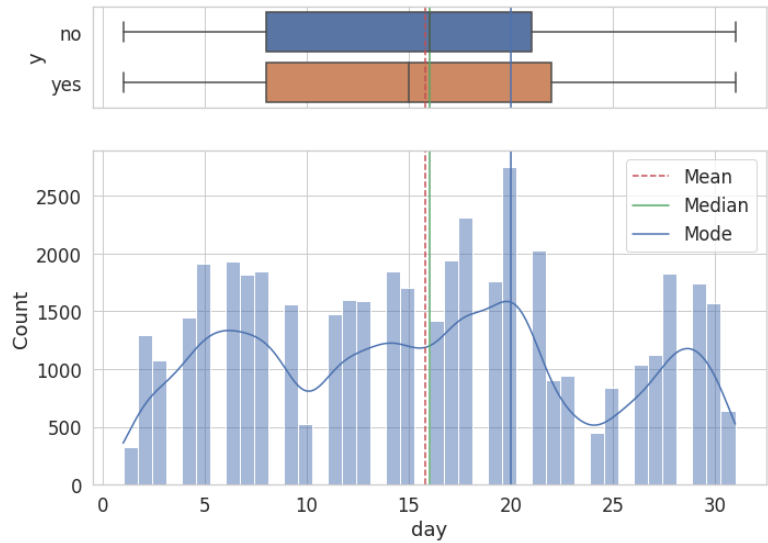
B. Fitur Numerikal (*age*, *balance*, *duration*, *campaign*, *pdays*, *previous*)



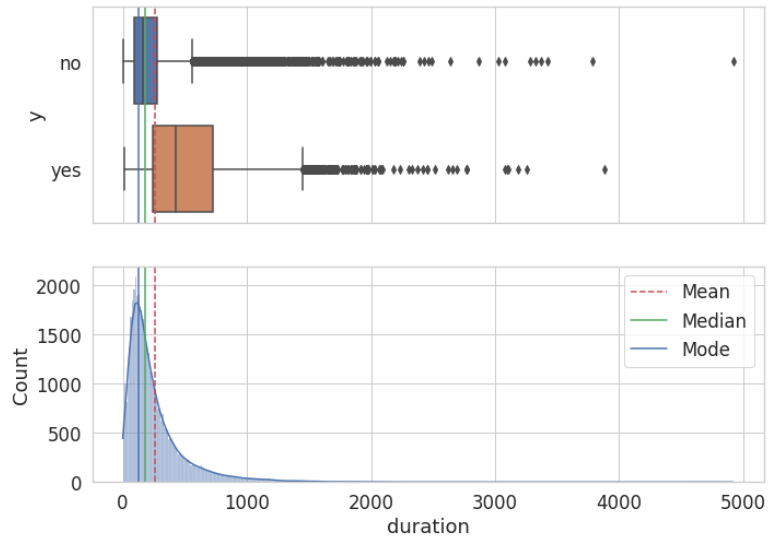
Gambar 17. Grafik Hubungan Umur Nasabah dengan Nasabah yang Berlangganan Deposit Berjangka



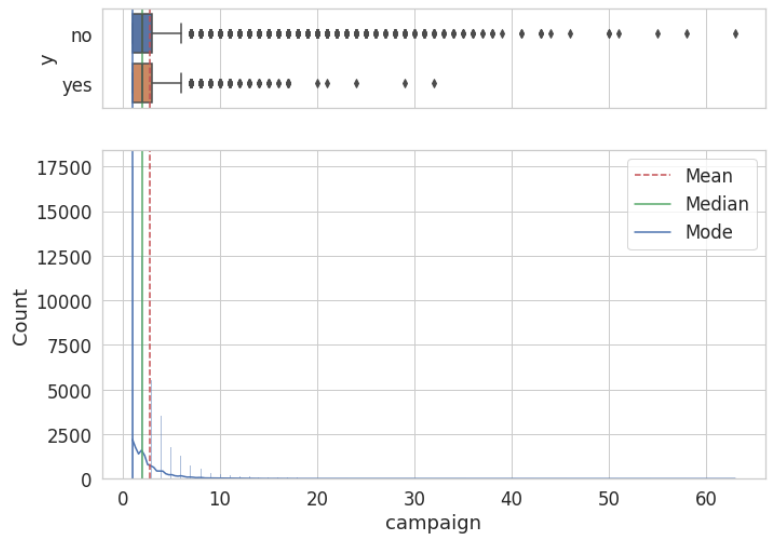
Gambar 18. Grafik Hubungan Saldo Nasabah dengan Nasabah yang Berlangganan Deposit Berjangka



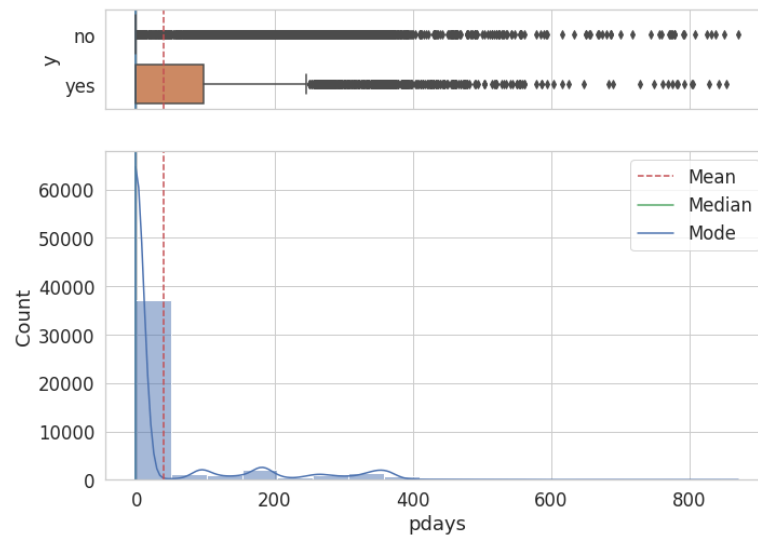
Gambar 19. Grafik Hubungan Hari Kontak Terakhir dengan Nasabah yang Berlangganan Deposit Berjangka



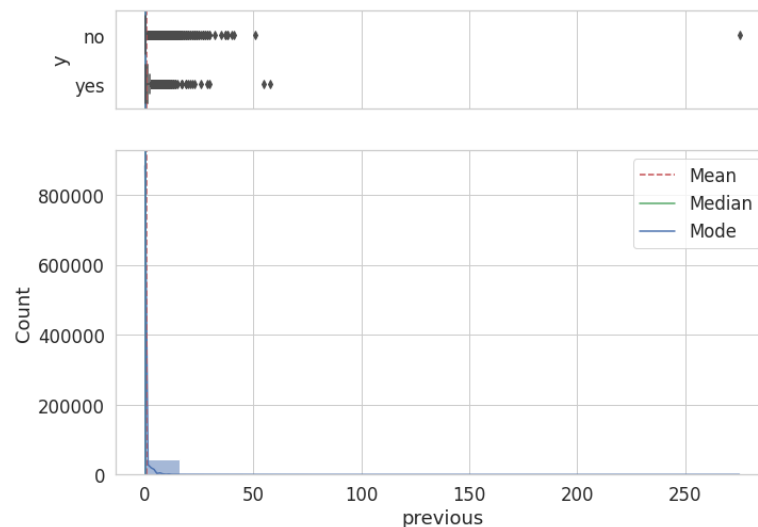
Gambar 20. Grafik Hubungan Durasi Kontak Terakhir dengan Nasabah yang Berlangganan Deposit Berjangka



Gambar 21. Grafik Hubungan Jumlah Kontak Dalam Kampanye Sekarang dengan Nasabah yang Berlangganan Deposit Berjangka



Gambar 22. Grafik Hubungan Hari Kontak Terakhir dengan Nasabah yang Berlangganan Deposit Berjangka



Gambar 23. Grafik Hubungan Jumlah Kontak Dalam Kampanye Sebelumnya dengan Nasabah yang Berlangganan Deposit Berjangka

Berdasarkan grafik fitur numerikal diatas, dapat diambil kesimpulan bahwa :

- Pada histogram ditampilkan kelompok populasi umur nasabah yaitu antara 20 tahun hingga 60 tahun. *Box plot* menampilkan target umur nasabah yaitu antara 30 tahun hingga 50 tahun. Hal ini dikarenakan kemungkinan pada umur tersebut merupakan umur produktif dan stabil. Trend ini dapat diperkuat dengan adanya fitur pekerjaan.
- Mayoritas nasabah yang dihubungi memiliki saldo tahunan rata-rata hampir nol.
- Histogram menunjukkan simetri tertentu pada seluruh kumpulan data hari terakhir nasabah dihubungi dengan puncaknya yaitu pada hari ke-20. *Box plot* menunjukkan bahwa nasabah berlangganan pada hari apapun selama seminggu.
- Durasi kontak terakhir memiliki pengaruh terhadap variabel *output y*. Pada garfik yang ditampilkan, dapat dilihat bahwa pada 0 menit hingga 2 menit mayoritas nasabah menolak panggilan, dalam rentang 2 hingga 12 menit nasabah mengambil keputusan dan hanya beberapa nasabah yang membutuhkan durasi yang cukup lama untuk menolak atau menerima penawaran deposito berjangka.

- Melalui *box plot* jumlah kontak yang dihubungi pada kampanye sebelumnya dapat diketahui bahwa nasabah yang berlangganan deposito berjangka dengan yang tidak berlangganan memiliki perbandingan yang sama yaitu 50 : 50. Nasabah yang dihubungi beberapa kali merupakan nasabah yang memutuskan berlangganan deposito berjangka, sehingga mencoba menghubungi nasabah terlalu sering merupakan hal yang percuma.
- Pada grafik hubungan jumlah hari yang berlalu setelah nasabah terakhir dihubungi dari kampanye pemasaran sebelumnya dengan nasabah yang berlangganan deposito berjangka, dapat dilihat bahwa median bernilai -1 dan *box plot* menampilkan bahwa hampir sebagian besar nasabah dihubungi untuk pertama kali. Fitur ini akan dihilangkan dikarenakan tidak berpengaruh terhadap hasil.
- Jumlah kontak yang dilakukan bank pada kampanye sebelumnya ditunjukkan pada grafik yang mana 36.954 sebesar 0 yang mengartikan bahwa sebanyak 36.954 nasabah dihubungi untuk pertama kalinya dalam kampanye pemasaran sekarang. *Box plot* juga mengindikasikan tidak adanya distribusi yang mengartikan fitur ini tidak memiliki hubungan apapun terhadap hasil.

3.4. Klasifikasi Logistic Regression

Klasifikasi menggunakan metode Logistic Regression dilakukan dengan membagi dataset menjadi 60% untuk data *training* dan 40% untuk data *testing* untuk mengetahui keberhasilan kampanye pemasaran deposito berjangka yang ditunjukkan pada Gambar 23.

```
y = train_test_concat["y"]
X = train_test_concat.drop("y",axis = 1)

X_train , X_test , y_train , y_test = train_test_split(X, y, test_size = 0.4, random_state = 42)

#Klasifikasi Logistic Regression 1
logreg = LogisticRegression(solver='liblinear')
logreg.fit(X_train,y_train)
y_pred = logreg.predict(X_test)
```

Gambar 24. Potongan *Source Code* Klasifikasi Logistic Regression

Hasil akurasi yang diperoleh menggunakan algoritma Logistic Regression yaitu sebesar 90.16% dengan jumlah *True Positive* sebanyak 17.226 data, *False Positive* sebanyak 1.583 data, *True Negative* sebanyak 710 data, dan *False Negative* sebanyak 374 data yang ditunjukkan pada Gambar 25.

```
[[17226 1583]
 [ 374  710]]
      precision    recall  f1-score   support

      0       0.92      0.98      0.95      17600
      1       0.65      0.31      0.42       2293

 accuracy          0.90      19893
 macro avg       0.79      0.64      0.68      19893
weighted avg       0.89      0.90      0.89      19893

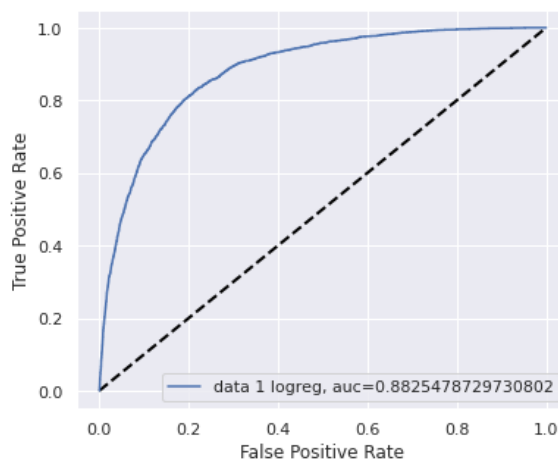
Accuracy: 0.9016236867239733
Precision: 0.6549815498154982
Recall: 0.30963802878325336
F1 score: 0.4204915605567071
Cross Validation mean: 0.898957795578991
```

Gambar 25. *Confusion Matrix* Logistic Regression

Dari hasil yang diperoleh pada Gambar 25 menunjukkan nilai *recall* yang sangat kecil jika dibandingkan dengan nilai *precision*. Nilai *precision* sebesar 92% adalah benar, namun representasi *recall* hanya menunjukkan nilai sebesar 30% kasus positif yang diidentifikasi benar dan 42% prediksi positif. Hal ini dikarenakan adanya ketidakseimbangan data dalam dataset yang digunakan.

Agar hal ini dapat diatasi maka dilakukan *oversampling* atau *undersampling* pada dataset kampanye pemasaran deposito berjangka.

Hasil dari prediksi kampanye pemasaran deposito berjangka menggunakan metode klasifikasi Logistic Regression dapat dilihat dalam kurva ROC (*Receive Operating Characteristic*) pada Gambar 25. Pada kurva tersebut diperoleh bahwa nilai AUC yaitu sebesar 0.882.



Gambar 26. Hasil Visualisasi Kurva ROC (AUC) LR

Dalam tesisnya, Yasilnacar melakukan penyajian terhadap penilaian AUC untuk mengkategorikan model prediksi yang dihasilkan sebagai berikut dalam Tabel 2[13].

Tabel 2. Klasifikasi Nilai AUC

Nilai AUC	Keterangan
>0.9 – 1	Luar biasa
>0.8 – 0.9	Sangat baik
>0.7 – 0.8	Baik
>0.6 – 0.7	Cukup baik
0.5 – 0.6	Tidak baik

Berdasarkan nilai akurasi AUC tersebut, dapat disimpulkan bahwa algoritma Logistic Regression yang digunakan untuk menganalisis dan memprediksi dataset kampanye pemasaran deposito berjangka diatas termasuk dalam kategori klasifikasi yang sangat baik.

A. *Oversampling*

Oversampling dilakukan dengan meningkatkan sampel kelas minoritas dengan cara menduplikasi secara acak sampel kelas minoritas hingga sampel kelas minoritas dan sampel kelas mayoritas seimbang, klasifikasi Logistic Regression dengan *oversampling* ditunjukkan pada Gambar 27.


```
#Oversampling
ROS = RandomOverSampler(sampling_strategy='minority', random_state=1)

X_train_ROS, y_train_ROS = ROS.fit_resample(X_train, y_train)
np.bincount(y_train_ROS)

array([26322, 26322])

#Klasifikasi Logistic Regression 2 dengan oversampling
logreg_oversampling = LogisticRegression(solver='liblinear')
logreg_oversampling.fit(X_train_ROS, y_train_ROS)
y_pred_oversampling = logreg_oversampling.predict(X_test)
```

Gambar 27. Potongan Source Code LR + oversampling

Hasil akurasi yang diperoleh menggunakan algoritma Logistic Regression dengan *oversampling* yaitu sebesar 82.57% dengan jumlah *True Positive* sebanyak 14.648 data, *False Positive* sebanyak 514 data, *True Negative* sebanyak 1.779 data, dan *False Negative* sebanyak 2.954 data yang ditunjukkan pada Gambar 28. Dapat dilihat bahwa nilai *recall* meningkat dan lebih tinggi dibandingkan nilai *precision*.

```
[[14648  514]
 [ 2952 1779]]
      precision    recall  f1-score   support

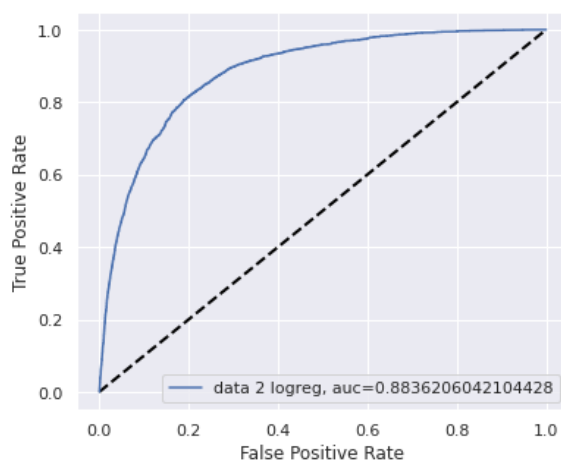
      0       0.97      0.83      0.89      17600
      1       0.38      0.78      0.51       2293

 accuracy      0.83      19893
 macro avg      0.67      0.80      0.70      19893
weighted avg      0.90      0.83      0.85      19893

Accuracy: 0.8257678580405168
Precision: 0.3760304375396322
Recall: 0.7758395115569123
F1 score: 0.5065489749430524
Cross Validation mean: 0.898957795578991
```

Gambar 28. Confusion Matrix LR + oversampling

Hasil dari prediksi kampanye pemasaran deposito berjangka dengan dataset yang seimbang dengan metode klasifikasi Logistic Regression setelah dilakukan *oversampling* dapat dilihat dalam kurva ROC (*Receive Operating Characteristic*) pada Gambar 27. Pada kurva tersebut diperoleh bahwa nilai AUC yaitu sebesar 0.8836.



Gambar 29. Hasil Visualisasi Kurva ROC (AUC) LR + oversampling

B. Undersampling

Undersampling dilakukan dengan mengurangi sampel kelas mayoritas dengan cara melakukan penghapusan secara acak sampel kelas mayoritas hingga sampel kelas mayoritas dan sampel kelas minoritas seimbang, klasifikasi Logistic Regression dengan *undersampling* ditunjukkan pada Gambar 30.

```
#Undersampling
RUS = RandomUnderSampler(sampling_strategy = 'majority', random_state = 1)

X_train_RUS, y_train_RUS = RUS.fit_resample(X_train, y_train)
np.bincount(y_train_RUS)

array([3517, 3517])

#Klasifikasi Logistic Regression 3 dengan undersampling
logreg_undersampling = LogisticRegression(solver='liblinear')
logreg_undersampling.fit(X_train_RUS, y_train_RUS)
y_pred_undersampling = logreg_undersampling.predict(X_test)
```

Gambar 30. Potongan Source Code LR + *undersampling*

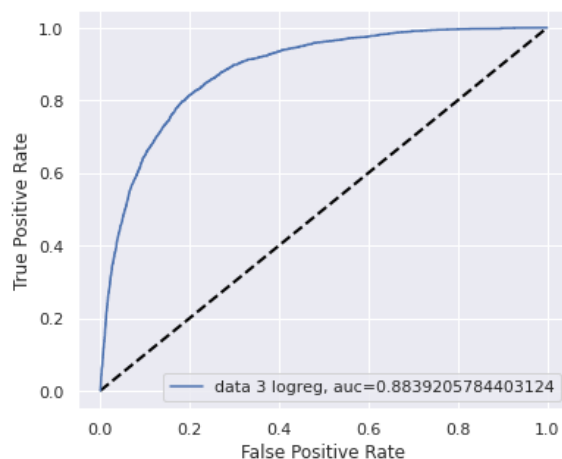
Hasil akurasi yang diperoleh menggunakan algoritma Logistic Regression dengan *undersampling* yaitu sebesar 82.88% dengan jumlah *True Positive* sebanyak 14.735 data, *False Positive* sebanyak 539 data, *True Negative* sebanyak 1.754 data, dan *False Negative* sebanyak 2.865 data yang ditunjukkan pada Gambar 31. Dapat dilihat bahwa nilai *recall* meningkat dan lebih tinggi dibandingkan nilai *precision*.

	precision	recall	f1-score	support
0	0.96	0.84	0.90	17600
1	0.38	0.76	0.51	2293
accuracy			0.83	19893
macro avg	0.67	0.80	0.70	19893
weighted avg	0.90	0.83	0.85	19893

Accuracy: 0.8288845322475242
Precision: 0.37973587356570687
Recall: 0.7649367640645443
F1 score: 0.5075231481481481
Cross Validation mean: 0.898957795578991

Gambar 31. Confusion Matrix LR + *undersampling*

Hasil dari prediksi kampanye pemasaran deposito berjangka dengan dataset yang seimbang dengan metode klasifikasi Logistic Regression setelah dilakukan *undersampling* dapat dilihat dalam kurva ROC (*Receive Operating Characteristic*) pada Gambar 32. Pada kurva tersebut diperoleh bahwa nilai AUC yaitu sebesar 0.8836.

Gambar 32. Hasil Visualisasi Kurva ROC (AUC) LR + *undersampling*

3.5. Klasifikasi Random Forest

Klasifikasi menggunakan metode Random Forest untuk memprediksi keberhasilan kampanye pemasaran deposito berjangka ditunjukkan pada Gambar 33.

```
#Klasifikasi Random Forest 1
rfc = RandomForestClassifier(random_state=0)
rfc.fit(X_train,y_train)
y_pred_rfc = rfc.predict(X_test)
```

Gambar 33. Potongan *Source Code* Random Forest

Hasil akurasi yang diperoleh menggunakan algoritma Random Forest yaitu sebesar 90.48% dengan jumlah *True Positive* sebanyak 17.118 data, *False Positive* sebanyak 1.411 data, *True Negative* sebanyak 882 data, dan *False Negative* sebanyak 482 data yang ditunjukkan pada Gambar 34.

```
[[17118 1411]
 [ 482  882]]
      precision    recall  f1-score   support

     0       0.92      0.97      0.95      17600
     1       0.65      0.38      0.48       2293

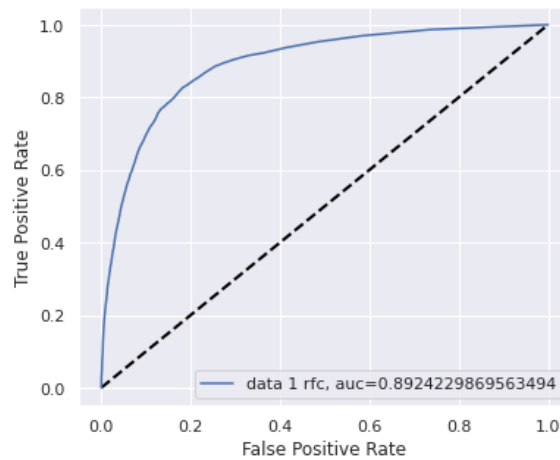
 accuracy          0.90      19893
 macro avg       0.79      0.68      0.71      19893
 weighted avg    0.89      0.90      0.89      19893

Accuracy: 0.9048408988086262
Precision: 0.6466275659824047
Recall: 0.38464893153074575
F1 score: 0.4823625922887613
Cross Validation mean: 0.90113611907493
```

Gambar 34. *Confusion Matrix* Random Forest

Dari hasil yang diperoleh pada Gambar 34 menunjukkan nilai *recall* yang sangat kecil jika dibandingkan dengan nilai *precision* yang mana lebih dari 60%. Hal ini dikarenakan adanya ketidakseimbangan data dalam dataset yang digunakan. Oleh sebab itu, dilakukan *oversampling* atau *undersampling* pada dataset kampanye pemasaran deposito berjangka untuk menyeimbangkan data.

Hasil dari prediksi kampanye pemasaran deposito berjangka menggunakan metode Random Forest dapat dilihat dalam kurva ROC (*Receive Operating Characteristic*) pada Gambar 34. Pada kurva tersebut diperoleh bahwa nilai AUC yaitu sebesar 0.8924, sehingga dapat dikatakan bahwa metode Random Forest merupakan metode klasifikasi yang sangat baik.



Gambar 35. Hasil Visualisasi Kurva ROC (AUC) Random Forest

A. *Oversampling*

Oversampling dilakukan dengan meningkatkan sampel kelas minoritas dengan cara menduplikasi secara acak sampel kelas minoritas hingga sampel kelas minoritas dan sampel kelas mayoritas seimbang, klasifikasi Random Forest dengan *oversampling* ditunjukkan pada Gambar 36.

```
#Klasifikasi Random Forest 2 dengan oversampling
rfc_oversampling = RandomForestClassifier(random_state=0)
rfc_oversampling.fit(X_train_ROS, y_train_ROS)
y_pred_rfc_oversampling = rfc_oversampling.predict(X_test)
```

Gambar 36. Potongan Source Code Random Forest + *oversampling*

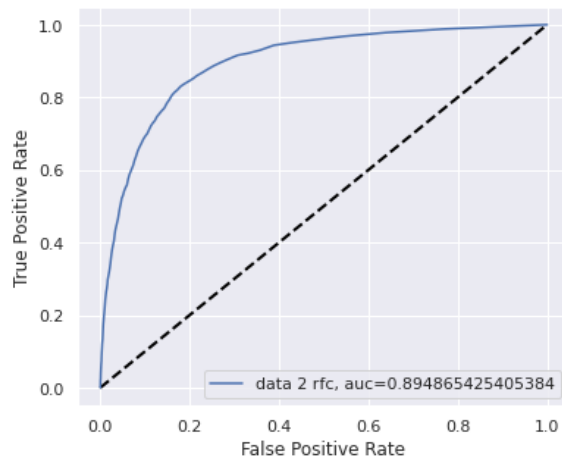
Hasil akurasi yang diperoleh menggunakan algoritma Random Forest dengan *oversampling* yaitu sebesar 90.27% dengan jumlah *True Positive* sebanyak 16.794 data, *False Positive* sebanyak 1.129 data, *True Negative* sebanyak 1164 data, dan *False Negative* sebanyak 896 data yang ditunjukkan pada Gambar 37.

[[16794 1129]					
[806 1164]]					
		precision	recall	f1-score	support
0	0.94	0.95	0.95	17600	
1	0.59	0.51	0.55	2293	
accuracy				0.90	19893
macro avg	0.76	0.73	0.75	19893	
weighted avg	0.90	0.90	0.90	19893	
Accuracy: 0.9027296033780727					
Precision: 0.5908629441624366					
Recall: 0.5076319232446577					
F1 score: 0.5460942997888811					
Cross Validation mean: 0.90113611907493					

Gambar 37. *Confusion Matrix* Random Forest + *oversampling*

Hasil dari prediksi kampanye pemasaran deposito berjangka dengan dataset yang seimbang dengan metode klasifikasi Random Forest setelah dilakukan *oversampling* dapat dilihat dalam

kurva ROC (*Receive Operating Characteristic*) pada Gambar 38. Pada kurva tersebut diperoleh bahwa nilai AUC yaitu sebesar 0.8836.



Gambar 38. Hasil Visualisasi Kurva ROC (AUC) Random Forest + *oversampling*

B. Undersampling

Undersampling dilakukan dengan meningkatkan sampel kelas minoritas dengan cara menduplikasi secara acak sampel kelas minoritas hingga sampel kelas minoritas dan sampel kelas mayoritas seimbang, klasifikasi Random Forest dengan *undersampling* ditunjukkan pada Gambar 39.

```
#Klasifikasi Random Forest 3 dengan undersampling
rfc_undersampling = RandomForestClassifier(random_state = 0)
rfc_undersampling.fit(X_train_RUS, y_train_RUS)
y_pred_rfc_undersampling = rfc_undersampling.predict(X_test)
```

Gambar 39. Potongan Source Code Random Forest + *undersampling*

Hasil akurasi yang diperoleh menggunakan algoritma Random Forest dengan *oversampling* yaitu sebesar 81.46% dengan jumlah *True Positive* sebanyak 14.301 data, *False Positive* sebanyak 388 data, *True Negative* sebanyak 1.905 data, dan *False Negative* sebanyak 3.299 data yang ditunjukkan pada Gambar 37.

```
[[14301  388]
 [ 3299 1905]]
      precision    recall  f1-score   support

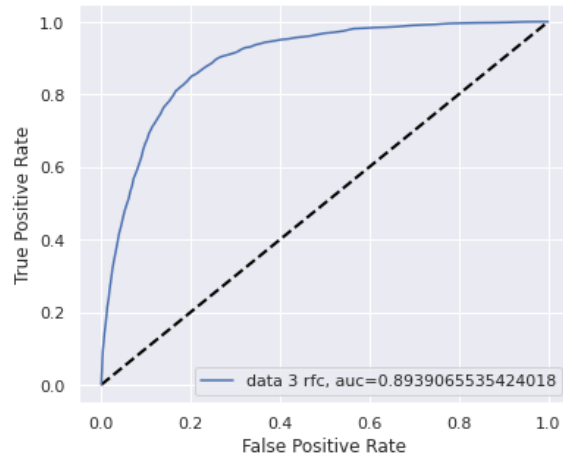
      0       0.97      0.81      0.89      17600
      1       0.37      0.83      0.51       2293

   accuracy      0.81      19893
  macro avg       0.67      0.82      0.70      19893
 weighted avg       0.90      0.81      0.84      19893

Accuracy: 0.8146584225606998
Precision: 0.36606456571867796
Recall: 0.8307893589184474
F1 score: 0.508203281312525
Cross Validation mean: 0.90113611907493
```

Gambar 40. Confusion Matrix Random Forest + *undersampling*

Hasil dari prediksi kampanye pemasaran deposito berjangka dengan dataset yang seimbang dengan metode klasifikasi Random Forest setelah dilakukan *undersampling* dapat dilihat dalam kurva ROC (*Receive Operating Characteristic*) pada Gambar 41. Pada kurva tersebut diperoleh bahwa nilai AUC yaitu sebesar 0.8939.



Gambar 41. Hasil Visualisasi Kurva ROC (AUC) Random Forest + *undersampling*

Berdasarkan hasil evaluasi yang dilakukan menggunakan *confusion matrix* dan kurva ROC (AUC), maka dapat diambil kesimpulan seperti yang ditunjukkan pada Tabel 3.

Tabel 3. Hasil Evaluasi Model Dengan *Confusion Matrix* dan Kurva ROC

	Accuracy	Precision	Recall	F1 Score	AUC Score
Logistic Regression	90.16%	65.49%	30.96%	42.04%	88.25%
Logistic Regression + <i>oversampling</i>	82.57%	37.60%	77.58%	50.65%	88.36%
Logistic Regression + <i>undersampling</i>	82.88%	37.97%	76.49%	50.75%	88.39%
Random Forest	90.48%	64.66%	38.46%	48.23%	89.24%
Random Forest + <i>oversampling</i>	90.27%	59.08%	50.76%	54.60%	89.48%
Random Forest + <i>undersampling</i>	81.24%	36.60%	83.07%	50.82%	89.93%

Model yang dibangun memiliki performa yang sangat baik dalam hal hasil nilai *accuracy* dan AUC *score*, tetapi memiliki nilai yang sangat rendah dalam performa pengukuran lainnya terutama dalam hal memprediksi kelas minoritas yaitu nasabah yang akan berlangganan deposito berjangka dari kampanye pemasaran yang dilakukan bank melalui telepon. Oleh sebab itu, untuk meningkatkan performa pengukuran lainnya, pada penelitian ini dilakukan *oversampling* dan *undersampling* untuk menyeimbangkan dataset yang digunakan. Berdasarkan hal tersebut, maka dari hasil yang ditampilkan Tabel 3 maka Logistic Regression pada dataset yang tidak seimbang merupakan model klasifikasi terbaik dengan nilai *precision* yaitu 65.49% dan *recall* 30.96% yang mana nilai *precision* lebih diutamakan dibandingkan *recall* sebab pengukuran ini dipertimbangkan untuk memprediksi kelas minoritas. Namun, model dengan peningkatan yang cukup besar pada nilai *recall* yang

dihasilkan dan memiliki *precision* yang cukup baik yaitu Random Forest dengan dataset seimbang yang didapatkan melalui *oversampling* dengan nilai *precision* 59.08% dan nilai *recall* 50.76%.

4. Kesimpulan

Pada penelitian analisis perbandingan prediksi kampanye pemasaran deposito berjangka menggunakan metode klasifikasi Logistik Regression dan Random Forest didapatkan bahwa Logistic Regression pada dataset yang tidak seimbang merupakan model klasifikasi terbaik dengan nilai *precision* yaitu 65.49% dan *recall* 30.96% yang mana nilai *precision* lebih diutamakan dibandingkan *recall* sebab pengukuran ini dipertimbangkan untuk memprediksi kelas minoritas. Namun, perlu dilakukan analisis lebih lanjut di masa depan mengenai model Random Forest dengan dataset seimbang yang diperoleh melalui *oversampling* dikarenakan peningkatan yang cukup baik pada nilai *recall* dengan nilai *precision* yang dikategorikan cukup baik.

Peneliti mengharapkan model yang dibangun dapat disempurnakan dengan mengkombinasikan dengan model lainnya dan memasukkan lebih banyak data kedalam dataset. Peneliti juga menyarankan untuk penelitian selanjutnya agar dilakukan analisis perbandingan model lainnya sehingga prediksi kampanye pemasaran deposito berjangka yang dilakukan dapat lebih tepat dan akurat di masa yang akan datang.

5. Ucapan terima kasih

Penulis mengucapkan terima kasih kepada Ibu Irma Permatasari, M.Eng selaku Dosen Mata Kuliah Pengolahan Citra yang telah membimbing penulis dalam penulisan jurnal ini.

REFERENCES

- [1] E. Setiawan, "Kamus Besar Bahasa Indonesia -Arti Kata Investasi," *KBBI Online*. 2019, [Online]. Available: <http://kbbi.web.id/pasar>.
- [2] M. A. Wibisono and R. Januarita, "Kegiatan Investasi yang Dilakukan Secara Online Dihubungkan dengan Undang-Undang Nomor 11 Tahun 2008 tentang Informasi dan Elektronik," no. 1, pp. 989–997, 2008.
- [3] Rahayu Puspasari, "Pertumbuhan Ekonomi Triwulan II 2022 Menguat Signifikan di Tengah Ketidakpastian dan Tren Perlambatan Global." pp. 1–3, 2022, [Online]. Available: https://fiskal.kemenkeu.go.id/files/siaran-pers/file/1659935841_sp115-pertumbuhan-indonesia-kuartal-ii-tahun-2022.pdf.
- [4] "10Tahun_1998Uu."
- [5] I. Firmansyah, J. T. Samudra, D. Pardede, Z. Situmorang, and U. P. Utama, "KOMPARASI RANDOM FOREST DAN LOGISTIC REGRESSION DALAM KLASIFIKASI PENDERITA COVID-19 BERDASARKAN," vol. 4307, no. 3, pp. 595–601, 2022.
- [6] F. I. Kurniadi and V. K. Putri, "Perbandingan Regresi Linear dengan Heaviside Activation Function dengan Logistic Regression untuk Klasifikasi Diabetes," *Ultimatics*, vol. X, no. 1, pp. 7–10, 2018, [Online]. Available: <https://www.webmd.com/diabetes/default.htm>.
- [7] D. Yusuf, B., Qalbi, M., Basrul and M. I., Malahayati & Ellyadi, "Implementasi Algoritma Naive Bayes Dan Random Forest Dalam Memprediksi Prestasi Akademik Mahasiswa Universitas Islam Negeri Ar-Raniry Banda Aceh," *Cybersp. J. Pendidik. Teknol. Inf.*, vol. 4, no. 1, pp. 50–58, 2020.
- [8] N. Khasawneh, E. Faouri, and M. Fraiwan, "Automatic Detection of Tomato Diseases Using Deep Transfer Learning," *Appl. Sci.*, vol. 12, no. 17, 2022, doi: 10.3390/app12178467.
- [9] Jonathan Sarwono, "Memadu Pendekatan Kuantitatif dan Kualitatif," *J. Ilm. Manaj. Bisnis*, vol. 9, no. 2, pp. 119–132, 2010, [Online]. Available: www.jonathansarwono.info.
- [10] E. et al Turban, "Decision Support Systems and Intelligent Systems (Sistem Pendukung Keputusan dan Sistem Cerdas," *Andi Offset*, 2005.
- [11] B. Santosa, *Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis*. Yogyakarta: Graha Ilmu, 2007.
- [12] L. B. C. Tanujaya, B. Susanto, and A. Saragih, "Perbandingan Metode Regresi Logistik dan Random Forest untuk Klasifikasi Fitur Mode Audio Spotify," *Indones. J. Data Sci.*, vol. 1, no. 3, pp. 68–78, 2020.
- [13] E. K. Yasilnacar, "The Application of Computational Intelligence to Landslide Susceptibility Mapping in Turkey," University of Melbourne, 2005.

