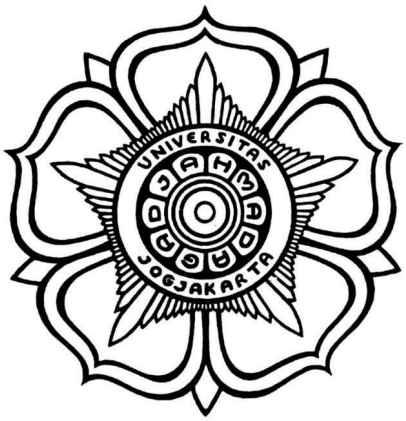
BIG DATA ANALYSIS

Ujian Akhir Semester



BY :

NADHIFA SOFIA

(19/448721/PPA/05804)

MASTER OF COMPUTER SCIENCE

DEPARTMENT OF COMPUTER SCIENCE AND ELECTRONICS

FACULTY OF MATHEMATICS AND NATURAL SCIENCES

UNIVERSITAS GADJAH MADA

2020

Instruksi

1. Ujian akhir semester dilaksanakan dalam bentuk submission report dalam bentuk white paper atau paper ilmiah yang memuat poin-poin pertanyaan yang diberikan
2. Submission dilaksanakan pada waktu berlangsungnya Ujian Akhir Semester
3. Gunakan Dataset dan dashboard analytic pada covid19.gamabox.id sebagai baseline analisis
4. File di submit melalui tautan : http://ugm.id/S2ABD

CO3. Be able to understand and to use big data technologies to process data

1. Jelaskan dan berikan analisis terhadap dashboard analytic yang telah dihasilkan pada covid19.gamabox.id yang dapat mengindentifikasi
    a. Jenis analisis yang telah dilakukan (descriptive, prescriptive, predictive dan cognitive)
    b. Berikan feedback terkait analisis yang dapat ditingkatkan untuk memunculkan insight dari data yang telah tersedia

CO4. Be able to implement and analysis algorithms of large scale data analysis

2. Jelaskan dan berikan analisis apa yang bisa dilakukan berbasis Exploratory Data Analysis, Algoritmik termasuk machine learning dan deep learning yang bisa dilakukan dengan data set yang ada, dengan melingkupi sebagai berikut
    a. Tujuan dari analisis yang akan dilakukan
    b. Bagaimana pengolahan/penyiapan data yang dilakukan
    c. Bagaimana implementasi algoritma yang dipilih dan dijalankan

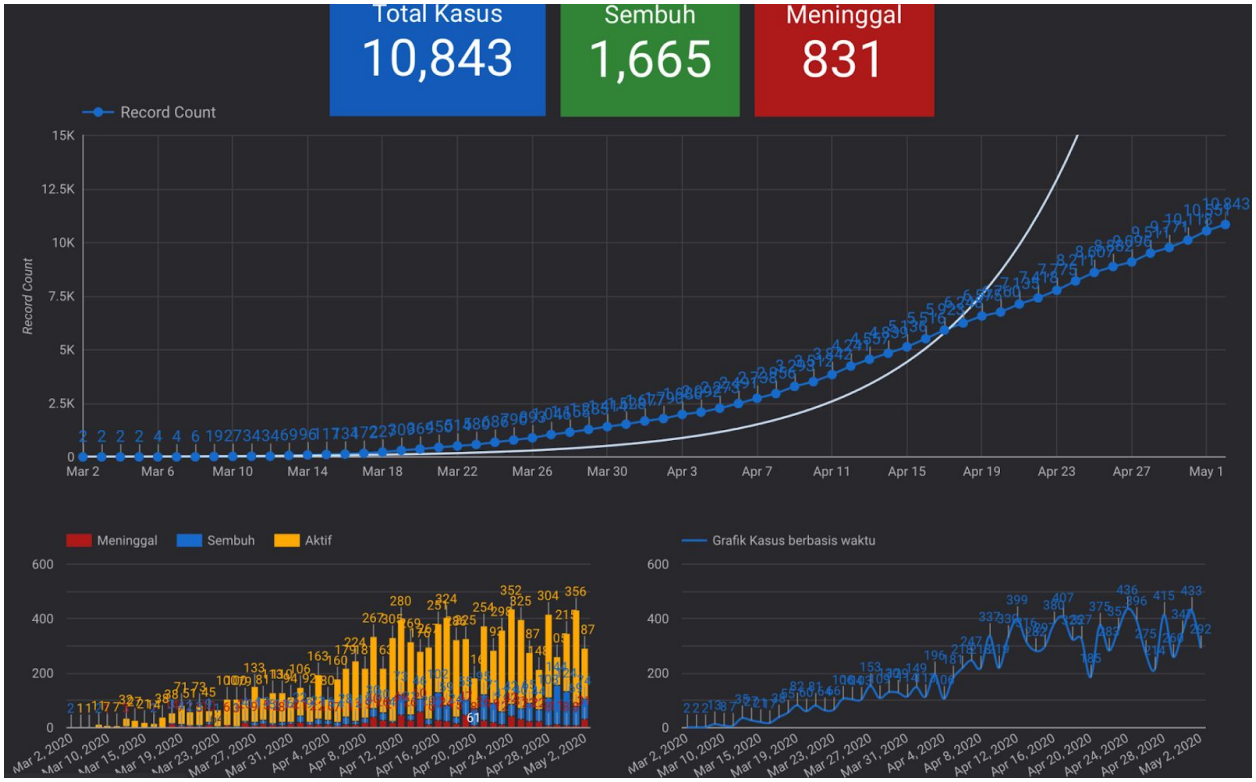
CO5. Be able to implement and analysis algorithms of data visualization

3. Jelaskan hasil dari implementasi analisis yang dilakukan dengan menjelaskan
    a. Jelaskan insight hasil data analisis yang dihasilkan
    b. Berikan penjelasan yang dapat dinarasikan untuk dipahami masyarakat awal bisa dalam bentuk naratif atau visualiasi.

Jawab:

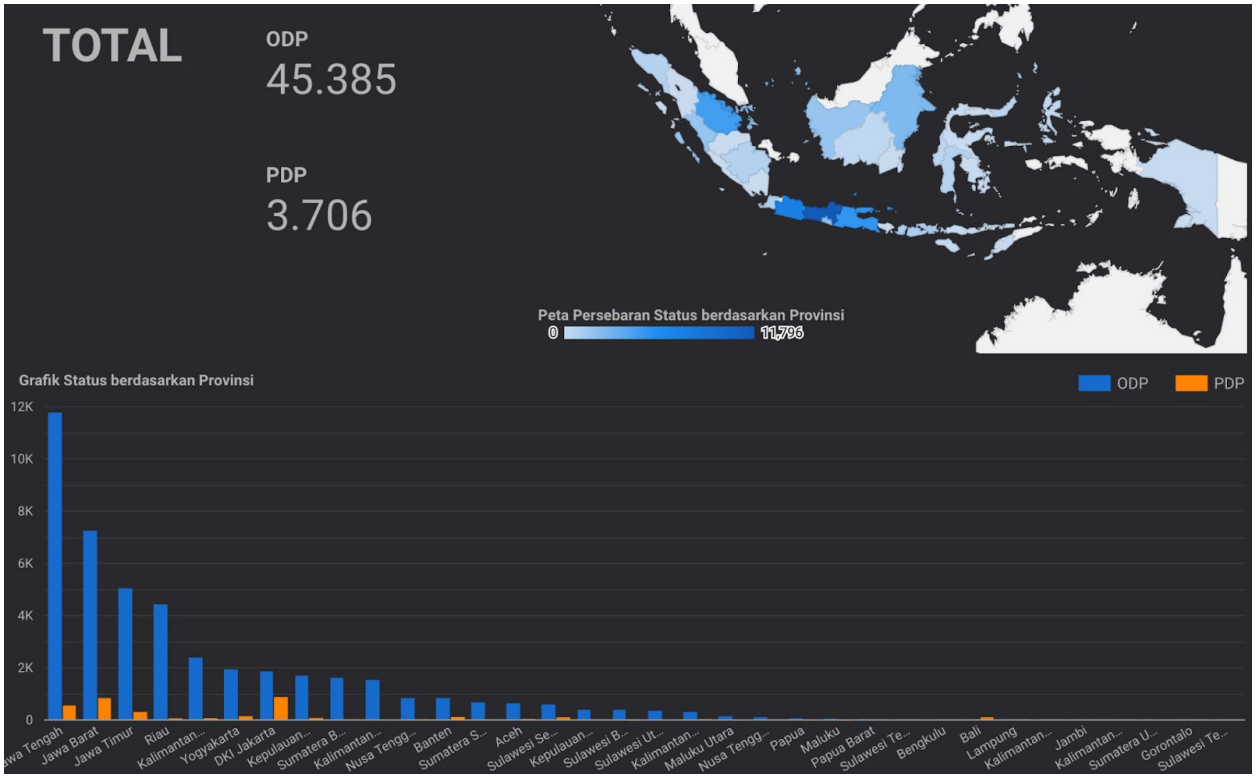
1. A. Jenis analisis yang dilakukan pada http://covid19.gamabox.id/analysis

Gambar 1. Segitiga analisis data [



Gambar 2. Analisis kasus dengan angka [1]

Gambar 3. Peta persebaran COVID-19 [1]



Gambar 4. Data ODP-PDP berdasarkan provinsi [1]

- Deskriptif
  - Hasil analisis deskriptif adalah pemahaman karakteristik bidang yang dikaji: apa yang wajar/tidak wajar, hal-hal yang kerap terjadi, hal-hal yang menonjol, dan hubungan antar variabel dalam data.
  - Contoh kegiatan analisis deskriptif adalah membuat ringkasan/agregasi data, sebagian teknik data mining, statistik deskriptif, dan visualisasi data.
  - Contoh pertanyaan yang dapat dipecahkan analisis deskriptif:
    - Kapan aktif kasus terbanyak yang dilaporkan oleh http://covid19.gamabox.id/analysis?
    - Sudah berapa banyak total kasus tertanggal 1 Mei 2020?
    - Berapa rata-rata kasus aktif tiap hari?

- Prediktif
  - Hasil analisis prediktif adalah ramalan terkait suatu variabel berdasarkan variabel lainnya di dalam data.
  - Contoh kegiatan analisis prediktif adalah sebagian teknik data mining, machine learning, regresi linier, dan simulasi.
  - Contoh pertanyaan yang dapat dipecahkan analisis prediktif:
    - Akan ada berapa banyak total kasus pada bulan Mei 2020?
    - Provinsi mana yang akan mendapatkan kasus COVID-19 terbanyak pada bulan Mei 2020?
    - Provinsi mana yang akan mendapatkan kasus COVID-19 paling sedikit pada bulan Mei 2020?

- Preskriptif
  - Hasil analisis preskriptif adalah rekomendasi aksi yang diperkirakan akan memaksimalkan pencapaian tujuan yang kita inginkan.
  - Contoh kegiatan analisis preskriptif adalah sebagian teknik simulasi, machine learning, teknik optimalisasi, dan analisa keputusan.
  - Contoh pertanyaan yang dapat dipecahkan:
    - Berapa persentase antara ODP kasus COVID-19 dan total penduduk Indonesia pada bulan Mei 2020?
    - Dimana sebaiknya dilakukan PSBB supaya provinsi tersebut tidak mendapatkan persebaran kasus lagi?

- Kognitif
  - Analitik kognitif menerapkan teknologi cerdas untuk membawa semua sumber data ini dalam jangkauan proses analisis untuk pengambilan keputusan dan intelijen bisnis.
  - Contoh kegiatan analisis kognitif memahami tidak hanya kata-kata dalam teks, tetapi konteks penuh dari apa yang sedang ditulis atau diucapkan, atau mengenali objek dalam gambar dalam sejumlah besar informasi.

- Contoh pertanyaan yang dapat dipecahkan:
  - Berapa persen peningkatan total kasus COVID-19 setiap hari?
  - Apakah kasus COVID-19 ini bisa diklasifikasikan sebagai peningkatan eksponensial?

B. Feedback analisis untuk http://covid19.gamabox.id/analysis

- Untuk pemaparan data sudah bagus, namun sebaiknya untuk angka pada peta persebaran tiap provinsi serta data ODP-PDP tiap provinsi tidak mengalami perubahan signifikan
- Sebaiknya terdapat perbedaan legenda warna pada peta persebaran COVID-19 seperti yang sudah dideskripsikan oleh kota kiri atas.

2. A. Tujuan analisis yang dilakukan
   - Untuk memahami kumpulan data covid-19 per tanggal 29 April 2020

B. Pengolahan/penyiapan data yang dilakukan [2-6]

- Pengecekan tipe data

Check the data types

Di sini kita memeriksa datatypes, terkadang perlu untuk mengkonversi string itu ke data integer hanya maka kita dapat memplot data melalui grafik . Di sini, dalam hal ini, data sudah bagus sehingga tidak perlu khawatir.

```
In [5]: # Checking the data type
        df.dtypes

Out[5]: t5                                  float64
        Provinsi                             object
        ODP Proses/Pemantauan                 int64
        ODP selesai                           int64
        ODP Total                             int64
        PDP Proses/Rawat Inap/Rawat Jalan     int64
        PDP Negatif/sembuh/selesai            int64
        PDP Meninggal                       float64
        PDP Total                             int64
        Positif Meninggal                     int64
        Positif Sembuh                        int64
        Positif dirawat                       int64
        Positif Total                         int64
        dtype: object
```

- Pengecekan distribusi (eg: menghapus kolom yang tidak sesuai/dibutuhkan)

Drop irrelevant columns

b. Bagaimana pengolahan/penyiapan data yang dilakukan

Langkah ini tentu diperlukan di setiap EDA karena kadang-kadang akan ada banyak kolom yang tidak pernah kita gunakan dalam studi kasus. Dalam hal ini, kolom seperti 't5' tidak terlalu digunakan, jadi saya hapus dulu.

In [6]:
```
# Menghapus kolom yang tidak digunakan
df = df.drop(['t5'], axis=1)
df.head()
```

Out[6]:

	Provinsi	ODP Proses/Pemantauan	ODP selesai	ODP Total	PDP Proses/Rawat Inap/Rawat Jalan	PDP Negatif/sembuh/selesai	PDP Meninggal	PDP Total	Positif Meninggal	Positif Sembuh	Positif dirawat	Positif Total
0	Aceh	316	1203	1519	13	69	1.0	83	1	4	4	9
1	Sumatera Utara	2970	0	2970	88	0	0.0	88	0	0	30	30
2	Sumatera Barat	1570	2496	4066	15	80	0.0	95	1	4	21	26
3	Riau	4434	0	4434	72	0	0.0	72	0	0	1	1
4	Kepulauan Riau	597	1752	2394	67	153	4.0	222	7	5	36	48

- Melakukan feature engineering (eg: menghapus data yang duplikat/redundansi)

Check duplicates

c. Bagaimana implementasi algoritma yang dipilih dan dijalankan

In [7]:
```
df.shape
```

Out[7]:
```
(35, 12)
```

In [8]:
```
# Mengecheck apakah terdapat duplikasi antar baris
duplicate_rows_df = df[df.duplicated()]
print('number of duplicate rows: ', duplicate_rows_df.shape)
```

```
number of duplicate rows:  (0, 12)
```

In [9]:
```
# Menghitung total baris tiap kolom
df.count()
```

Out[9]:
```
Provinsi                            35
ODP Proses/Pemantauan               35
ODP selesai                         35
ODP Total                           35
PDP Proses/Rawat Inap/Rawat Jalan   35
PDP Negatif/sembuh/selesai          35
PDP Meninggal                       34
PDP Total                           35
Positif Meninggal                   35
Positif Sembuh                      35
Positif dirawat                     35
Positif Total                       35
dtype: int64
```

- Mengganti missing values menjadi 0 atau menghapus missing values

Check missing values

Dalam kasus ini, missing values terdapat pada 'PDP Total' untuk Prov. Papua Barat, oleh karena itu saya replace menjadi 0.

In [10]:
```
# Mencari missing values
print(df.isnull().sum())
```

```
Provinsi                            0
ODP Proses/Pemantauan               0
ODP selesai                         0
ODP Total                           0
PDP Proses/Rawat Inap/Rawat Jalan   0
PDP Negatif/sembuh/selesai          0
PDP Meninggal                       1
PDP Total                           0
Positif Meninggal                   0
Positif Sembuh                      0
Positif dirawat                     0
Positif Total                       0
dtype: int64
```

In [11]:
```
# Menganti missing values
df['PDP Meninggal'].fillna( method ='ffill', inplace = True)
```

In [12]:
```
# Mengecek kembali apakah masih ada missing values
print(df.isnull().sum())
```

```
Provinsi                            0
ODP Proses/Pemantauan               0
ODP selesai                         0
ODP Total                           0
PDP Proses/Rawat Inap/Rawat Jalan   0
PDP Negatif/sembuh/selesai          0
PDP Meninggal                       0
PDP Total                           0
Positif Meninggal                   0
Positif Sembuh                      0
Positif dirawat                     0
Positif Total                       0
dtype: int64
```

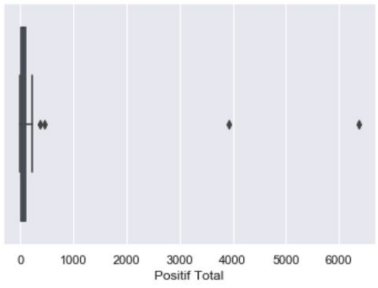
- Mengecek outliers

Detecting outliers

Outliers adalah titik atau kumpulan poin yang berbeda dari poin lainnya. Terkadang mereka bisa sangat tinggi atau sangat rendah. Sering kali ide yang baik untuk mendeteksi dan menghapus pencilan. Karena outlier adalah salah satu faktor untuk menghasilkan model yang kurang akurat, maka ide yang baik adalah untuk menghapusnya. Deteksi outlier dan menghilangkan yang akan saya lakukan disebut teknik skor IQR. Seringkali outlier dapat dilihat dengan visualisasi menggunakan plot kotak.

In [13]:
```
sns.boxplot(x=df['Positif Total'])
```

Out[13]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a1bab81d0>
```



In [14]:
```
# Mencari nilai IQR pada kolom tertentu
q1 = df['Positif Total'].quantile(.25)
q2 = df['Positif Total'].quantile(.5)
q3 = df['Positif Total'].quantile(.75)
mask = df['Positif Total'].between(q1, q2, inclusive=True)
iqr = df.loc[mask, 'Positif Total']
```

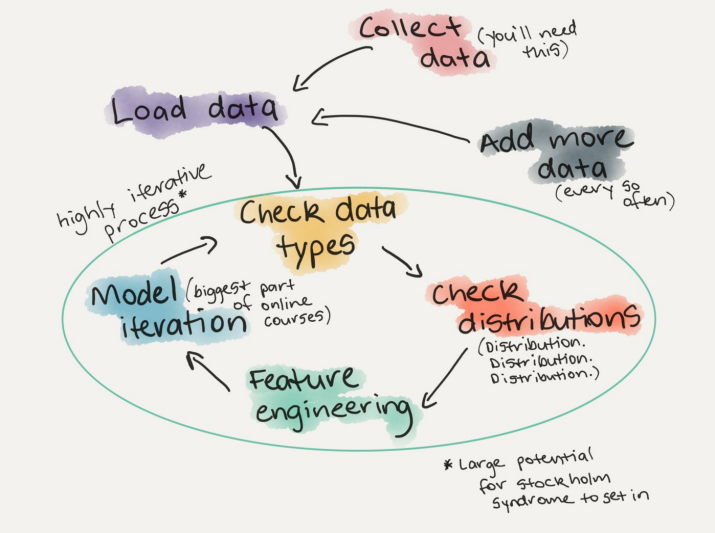
```
In [15]: print(iqr)

1     30
2     26
4     48
7     13
8     10
9     14
16    10
24    16
33    16
Name: Positif Total, dtype: int64
```

```
In [16]: q1 = df.quantile(.25)
         q2 = df.quantile(.5)
         q3 = df.quantile(.75)
         iqr = q3 - q1
         print(iqr)

ODP Proses/Pemantauan                1699.0
ODP selesai                          2462.5
ODP Total                            3637.5
PDP Proses/Rawat Inap/Rawat Jalan     154.0
PDP Negatif/sembuh/selesai            101.0
PDP Meninggal                           0.5
PDP Total                             391.0
Positif Meninggal                       5.0
Positif Sembuh                         13.0
Positif dirawat                        83.5
Positif Total                          95.5
dtype: float64
```

```
In [17]: df.shape
Out[17]: (35, 12)
```



Gambar 5. Siklus EDA [5]

C. Implementasi algoritma yang dipilih dan dijalankan

- Lebih lengkapnya ada di https://github.com/dhifaaans/uas_abd/blob/master/uas_nadhifasofia_KKPMDD.ipynb
- Mengimplementasi penggunaan salah satu dari algoritma di Gambar 6

	Model	Score
7	CatBoost	81.78
6	Gradient Boosting Trees	81.10
5	Decision Tree	79.42
1	Logistic Regression	78.52
2	Naive Bayes	76.38
0	KNN	73.68
4	Linear SVC	72.33
3	Stochastic Gradient Decent	61.19

Gambar 6. Peringkat performa model machine learning yang bisa digunakan untuk mengolah EDA versi Kaggle [5]

- Contohnya penggunaan CatBoost (Categorical Booster) ⇔ Algoritma pembelajaran mesin karena library ini didasarkan pada gradient boosting library.
- Gradient boosting adalah algoritma pembelajaran mesin yang kuat yang secara luas diterapkan untuk berbagai jenis tantangan bisnis seperti fraud detection, pemilihan rekomendasi, prediksi, dll. Ini juga dapat mengembalikan hasil yang sangat baik dengan data yang relatif lebih sedikit, tidak seperti model Deep Learning yang perlu belajar dari sejumlah besar data [7].
- Contoh pada studi kasus ini adalah prediksi kasus COVID-19 untuk setiap provinsi di Indonesia.
- Cara kerja algoritmanya ada pada Gambar 7.

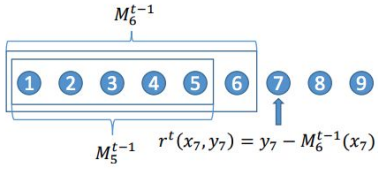


Figure 1: Ordered boosting principle.

Algorithm 1: Ordered boosting

input : $\{(\mathbf{x}_k, y_k)\}_{k=1}^n$, I;

$\sigma \leftarrow$ random permutation of $[1, n]$;
$M_i \leftarrow 0$ for $i = 1..n$;
for $t \leftarrow 1$ **to** I **do**
  for $i \leftarrow 1$ **to** n **do**
    $r_i \leftarrow y_i - M_{\sigma(i)-1}(i)$;
  for $i \leftarrow 1$ **to** n **do**
    $\Delta M \leftarrow$
     $LearnModel((\mathbf{x}_j, r_j) :$
     $\sigma(j) \leq i)$;
    $M_i \leftarrow M_i + \Delta M$;
return M_n

Algorithm 2: Building a tree in CatBoost

input : M,$\{y_i\}_{i=1}^n$, α, L, $\{\sigma_i\}_{i=1}^s$, $Mode$

$grad \leftarrow CaclGradient(L, M, y)$;
$r \leftarrow random(1, s)$;
$G \leftarrow (grad_r(1), \ldots, grad_r(n))$ for $Plain$;
$G \leftarrow (grad_{r,\sigma_r(1)-1}(i)$ for $i = 1$ to $n)$ for $Ordered$;
$T \leftarrow$ empty tree;
foreach *step of top-down procedure* **do**
  foreach *candidate split* c **do**
    $T_c \leftarrow$ add split c to T;
    if $Mode == Plain$ **then**
      $\Delta(i) \leftarrow$ avg$(grad_r(p)$ for
       $p : leaf(p) = leaf(i))$ for all i;
    if $Mode == Ordered$ **then**
      $\Delta(i) \leftarrow$ avg$(grad_{r,\sigma_r(i)-1}(p)$ for
       $p : leaf(p) = leaf(i), \sigma_r(p) < \sigma_r(i))$ $\forall i$;
    $loss(T_c) \leftarrow ||\Delta - G||_2$
  $T \leftarrow$ argmin$_{T_c}(loss(T_c))$
if $Mode == Plain$ **then**
  $M_{r'}(i) \leftarrow M_{r'}(i) - \alpha$ avg$(grad_{r'}(p)$ for
   $p : leaf(p) = leaf(i))$ for all r', i;
if $Mode == Ordered$ **then**
  $M_{r',j}(i) \leftarrow M_{r',j}(i) - \alpha$ avg$(grad_{r',j}(p)$ for
   $p : leaf(p) = leaf(i), \sigma_{r'}(p) \leq j$ for all r', j, i;
return T, M

Gambar 7. Cara kerja algoritma CatBoost [8]

3. A. Jelaskan insight hasil data analisis yang dihasilkan
   - Heat Maps adalah jenis plot yang diperlukan ketika kita perlu menemukan variabel dependen. Salah satu cara terbaik untuk menemukan hubungan antara fitur dapat dilakukan dengan menggunakan heat maps. Pada peta panas di bawah ini kita tahu bahwa fitur harga terutama tergantung pada ODP Proses, ODP Selesai, dsb.
     - Insight: Dengan legend warna yang semakin **hijau**, berarti antar fitur sangat **tinggi** tingkat korelasinya.
     - Insight: Dengan legend warna yang semakin **coklat**, berarti antar fitur sangat **rendah** tingkat korelasinya.

```
In [18]: # Menemukan hubungan antar variabel
         plt.figure(figsize=(20,10))
         c= df.corr()
         sns.heatmap(c,cmap='BrBG',annot=True)
         c
```

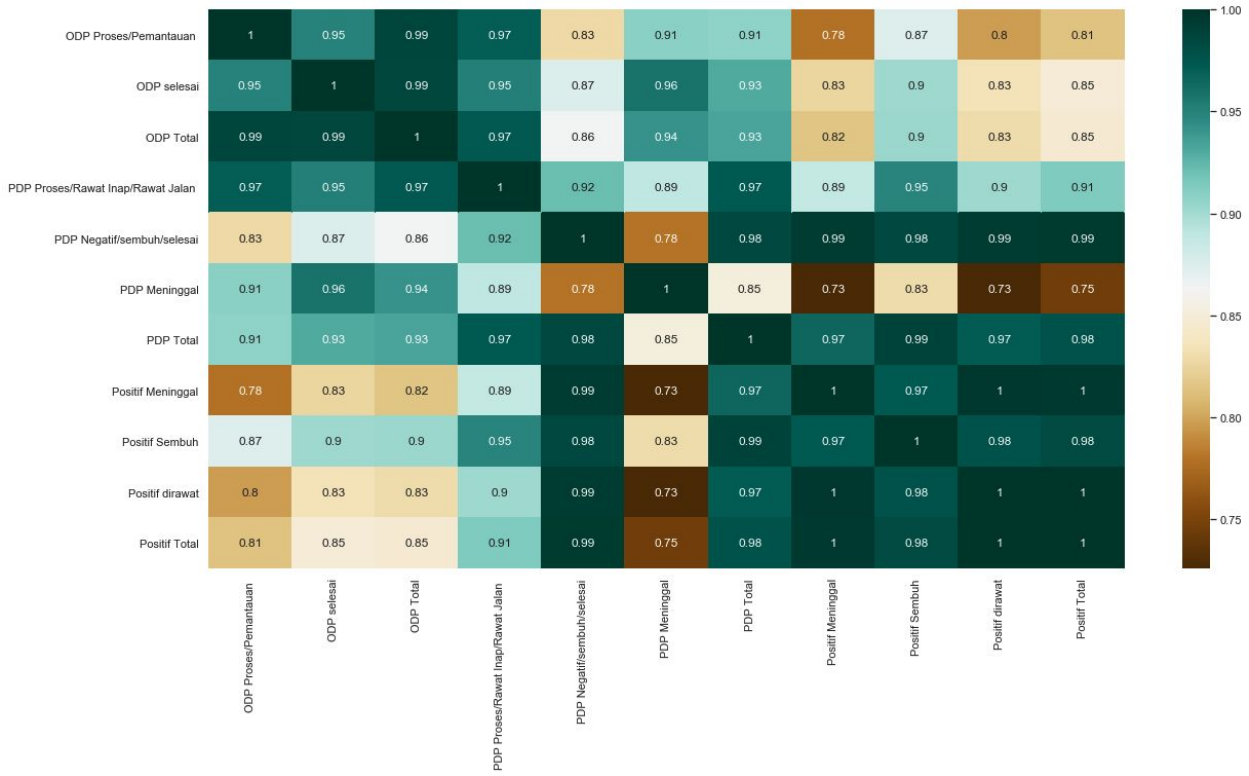
Out[18]:

	ODP Proses/Pemantauan	ODP selesai	ODP Total	PDP Proses/Rawat Inap/Rawat Jalan	PDP Negatif/sembuh/selesai	PDP Meninggal	PDP Total	Positif Meninggal	Positif Sembuh	Positif dirawat	Positif Total
ODP roses/Pemantauan	1.000000	0.950029	0.986181	0.970613	0.827979	0.909946	0.907743	0.778710	0.874489	0.797446	0.813834
ODP selesai	0.950029	1.000000	0.985730	0.952457	0.874916	0.959318	0.930961	0.826443	0.902118	0.834255	0.848688
ODP Total	0.986181	0.985730	1.000000	0.973998	0.863977	0.941306	0.933564	0.815595	0.903582	0.829927	0.846559
PDP Proses/Rawat Inap/Rawat Jalan	0.970613	0.952457	0.973998	1.000000	0.920913	0.890786	0.973033	0.888101	0.949523	0.902513	0.913845
PDP ıtif/sembuh/selesai	0.827979	0.874916	0.863977	0.920913	1.000000	0.780419	0.984560	0.994183	0.984182	0.993642	0.994966
PDP Meninggal	0.909946	0.959318	0.941306	0.890786	0.780419	1.000000	0.851737	0.725933	0.830030	0.726346	0.745591
PDP Total	0.907743	0.930961	0.933564	0.973033	0.984560	0.851737	1.000000	0.965988	0.988787	0.971340	0.978293
Positif Meninggal	0.778710	0.826443	0.815595	0.888101	0.994183	0.725933	0.965988	1.000000	0.972239	0.997622	0.996273
Positif Sembuh	0.874489	0.902118	0.903582	0.949523	0.984182	0.830030	0.988787	0.972239	1.000000	0.977959	0.983916
Positif dirawat	0.797446	0.834255	0.829927	0.902513	0.993642	0.726346	0.971340	0.997622	0.977959	1.000000	0.999122
Positif Total	0.813834	0.848688	0.846559	0.913845	0.994966	0.745591	0.978293	0.996273	0.983916	0.999122	1.000000

Gambar 8. Tabel korelasi antar fitur [9]

B. Berikan penjelasan yang dapat dinarasikan untuk dipahami masyarakat awal bisa dalam bentuk naratif atau visualisasi

- Setelah kita mengetahui gambaran umum dari data/tabel yang kita miliki, kini saatnya kita beralih untuk pengolahan lebih jauh. Muncul pertanyaan, data mana saja yang butuh untuk diolah? Dari 11 features yang tersedia dalam tabel data covid-19 ini, data manakah yang paling memberikan pengaruh terhadap data features lainnya.

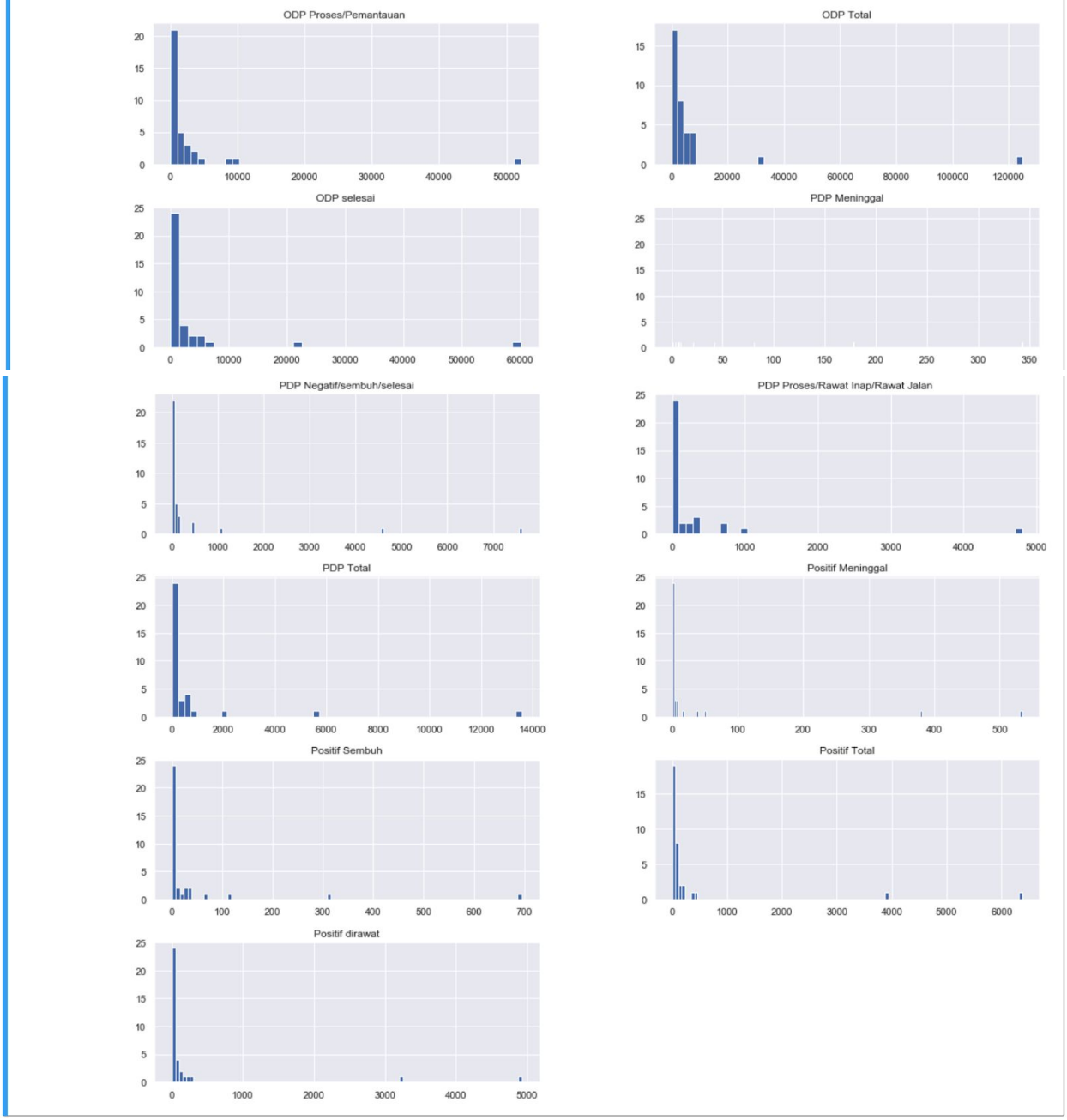


Gambar 9. Visualisasi HeatMap untuk korelasi antar fitur [9]

- Membuat histogram untuk melihat persentase perolehan suatu fitur terhadap fitur lainnya

```
In [31]: # Membuat histogram
         df.hist(bins='auto', figsize=(18, 22), layout=(6, 2))

Out[31]: array([[<matplotlib.axes._subplots.AxesSubplot object at 0x1a1d68c0f0>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x1a20b67860>],
                [<matplotlib.axes._subplots.AxesSubplot object at 0x1a20bc3748>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x1a20c413c8>],
                [<matplotlib.axes._subplots.AxesSubplot object at 0x1a20c56908>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x1a215fce48>],
                [<matplotlib.axes._subplots.AxesSubplot object at 0x1a216363c8>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x1a21663d30>],
                [<matplotlib.axes._subplots.AxesSubplot object at 0x1a21663d68>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x1a216d40b8>],
                [<matplotlib.axes._subplots.AxesSubplot object at 0x1a21702a58>,
                 <matplotlib.axes._subplots.AxesSubplot object at 0x1a2173c438>]],
               dtype=object)
```



Gambar 10. Contoh visualisasi menggunakan histogram [9]

REFERENSI

[1] http://covid19.gamabox.id/analysis#

[2] https://medium.com/labtek-indie/exploratory-data-analysis-7b9b0234ba05

[3] https://towardsdatascience.com/exploratory-data-analysis-in-python-c9a77dfa39ce

[4] https://towardsdatascience.com/ways-to-detect-and-remove-the-outliers-404d16608dba

[5] https://towardsdatascience.com/a-gentle-introduction-to-exploratory-data-analysis-f11d843b8184

[6] https://cloud.google.com/blog/products/ai-machine-learning/building-ml-models-with-eda-feature-selection

[7] https://www.analyticsvidhya.com/blog/2017/08/catboost-automated-categorical-data/

[8] https://towardsdatascience.com/https-medium-com-talperetz24-mastering-the-new-generation-of-gradient-boosting-db04062a7ea2

[9] https://github.com/dhifaaans/uas_abd/blob/master/uas_nadhifasofia_KKPMDD.ipynb