

**LAPORAN PENELITIAN  
PROGRAM STUDI**

**PEMANFAATAN MACHINE LEARNING DENGAN  
ALGORITMA SUPPORT VECTOR MACHINE,  
POLYNOMIAL REGRESSION DAN BAYESIAN RIDGE  
REGRESSION UNTUK MEMPREDIKSI KASUS COVID-19  
DI INDONESIA**



**Ilka Zufria, M.Kom (Peneliti 1)**

**Dr. Mhd. Furqan, S.Si, M.Comp.Sc (Peneliti 2)**

**PROGRAM STUDI ILMU KOMPUTER  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI (UIN) SUMATERA UTARA  
MEDAN  
2020**

## LEMBAR PENGESAHAN

1. a. Judul Penelitian : Pemanfaatan *Machine Learning* Dengan *Support Vector Machine, Polynomial Regr Bayesian Ridge Regression* Untuk Mempred  
: Covid-19 Di Indonesia
- b. Kluster Penelitian : Penelitian Program Studi
- c. Bidang Keilmuan : Ilmu Komputer
- d. Kategori : Kelompok
2. Peneliti 1 : Ilka Zufria, M.Kom  
NIP Peneliti 1 : 198506042015031006
3. Peneliti 2 : Dr. Mhd. Furqan, S.Si, M.Comp.Sc  
NIP Peneliti 2 : 198008062006041003
4. Unit Kerja : Fakultas Sains dan Teknologi – UIN SU Mec
5. Waktu Penelitian : 3 bulan, tahun 2020

Disahkan oleh  
Dekan  
Fakultas Sains dan Teknologi



**Dr. H. M. Jamil, M.A**  
NIP. 196608101999031002

Medan, 04 Nopember 2020  
Peneliti,



**Ilka Zufria, M.Kom**  
NIP. 198506042015031006

## SURAT PERNYATAAN BEBAS PLAGIASI

Yang bertanda tangan di bawah ini :

Nama : Ilka Zufria, M.Kom  
Jabatan : Dosen / Lektor /Gol. III.c  
Unit Kerja : Fakultas Sains dan Teknologi  
                  UIN Sumatera Medan  
Alamat : Jalan IAIN No. 1 Medan

dengan ini menyatakan bahwa :

1. Judul penelitian “ Pemanfaatan *Machine Learning* Dengan Algoritma *Support Machine, Polynomial Regression, dan Bayesian Ridge Regression* Memprediksi Kasus Covid-19 Di Indonesia ” merupakan karya orisinal saya.
2. Jika dikemudian hari ditemukan fakta bahwa judul, hasil atau bagian dari penelitian saya merupakan karya orang lain dan/atau plagiasi, maka saya bertanggung jawab untuk mengembalikan 100% dana hibah penelitian yang telah diterima, dan siap mendapatkan sanksi sesuai ketentuan yang berlaku.

Demikian pernyataan ini dibuat untuk digunakan sebagaimana mestinya.

Medan, 04 Nopember 2020

**Yang Menyatakan,**



**Ilka Zufria, M.Kom**

NIP. 198506042015031006

## REKOMENDASI

Setelah membaca dan menelaah hasil penelitian yang berjudul “Pemanfaatan *Machine Learning* Dengan Algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* Untuk Memprediksi Kasus Covid-19 Di Indonesia”. Yang dilakukan oleh Ilka Zufria, M.Kom maka saya berkesimpulan bahwa hasil penelitian ini dapat diterima sebagai karya tulis berupa hasil penelitian. Demikianlah rekomendasi diberikan kepada yang bersangkutan untuk dapat dipergunakan sebagaimana mestinya.

Medan, 02 Nopember 2020

Konsultan



**Dr. Mhd. Furqan, S.Si, M.Comp.Sc**  
NIP. 198008062006041003

## ABSTRAK

Pandemi virus corona tahun 2019 yang disebut dengan Covid-19 telah membawa kegelisahan pada dunia termasuk negara Indonesia. Covid-19 pertama kali masuk di Indonesia pada tanggal 2 Maret 2020 dan angka kasus penyebaran terus melonjak tinggi dalam kurun 7 bulan terakhir sampai di bulan Oktober 2020. Berdasarkan data yang tersedia di laman website Kementerian Kesehatan Indonesia pada setiap provinsi di Indonesia, algoritma mesin pembelajaran dapat mempelajari pola dan memprediksi perkembangan banyaknya kasus yang dapat menjadi himbauan bagi warga agar semakin bersikap hati-hati. Data yang digunakan mulai dari tanggal 1 september sampai dengan 28 oktober 2020 untuk memprediksi banyak kasus selama 60 hari ke depan. Beberapa algoritma mesin pembelajaran yang digunakan pada penelitian ini yakni support vector machine dan polynomial regression. Ketiga algoritma ini memiliki rumusan dalam mempelajari set data dengan parameter yang berbeda-beda. Penggunaan algoritma pembelajaran tersebut telah banyak digunakan pada sejumlah penelitian dalam memprediksi, termasuk banyak kasus covid-19 di berbagai wilayah di dunia. Algoritma *Support Vector Machine* memprediksi akan ada 800.000 kasus covid-19 dalam 60 hari mendatang dengan  $RMSE = 4913.96$ . Sedangkan *Polynomial Regression* memprediksi sebanyak lebih dari 450.000 kasus covid-19 dengan  $RMSE = 560.37$  dan *Bayesian Ridge Regression* memprediksi sebanyak 2.000.000 kasus dengan  $RMSE = 44912.27$ .

**Kata kunci:** covid-19, *bayesian ridge regression*, *machine learning*, prediksi, *support vector machine*.

## ABSTRACT

*The coronavirus pandemic in 2019 called Covid-19 has brought anxiety to the world including Indonesia. Covid-19 first entered Indonesia on March 2, 2020 and the number of cases of spread continued to soar in the last 7 months until October 2020. Based on the data available on the website of the Indonesian Ministry of Health in every province in Indonesia, machine learning algorithms can learn patterns and predict the development of the number of cases that can be an exhortation for citizens to be more cautious. The data is used from September 1 to October 28, 2020 to predict many cases over the next 60 days. Some of the learning machine algorithms used in this study are support vector machine, polynomial regression, and bayesian ridge regression. These three algorithms have formulas for studying data sets with different parameters. The use of such learning algorithms has been widely used in a number of studies in predicting, including many cases of covid-19 in various regions of the world. Support Vector Machine algorithm predicts there will be 800.000 cases of covid-19 in the next 60 days with  $RMSE = 4913.96$ . While Polynomial Regression predicts as many as 450.000 cases with  $RMSE = 560.37$  and Bayesian Ridge Regression predicts as many as 2.000.000 cases with  $RMSE = 44912.27$ .*

**Keywords:** *covid-19, bayesian ridge regression, machine learning, polynomial regression, prediction, support vector machine*

## KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirobbil Alamiin, segala puji bagi Allah SWT. Atas berkat rahmat dan karuniaNya, saya dan tim penelitian dapat menyelesaikan penelitian ini dengan judul “**Pemanfaatan *Machine Learning* Dengan Algoritma *Support Vector Machine* Dan *Polynomial Regression* Untuk Memprediksi Kasus Covid-19 Di Indonesia**”. Sholawat dan salam senantiasa dipanjatkan kepada baginda Muhammad SAW beserta kerabat, sahabat, para pengikutnya sampai akhir zaman, adalah sosok yang telah membawa manusia dan seisi alam dari kegelapan ke cahaya sehingga kita menjadi manusia beriman, berilmu, dan tetap beramal shaleh agar menjadi manusia yang berakhlak mulia.

Penulisan laporan ini bertujuan untuk melengkapi persyaratan luaran penelitian. Laporan ini juga diharapkan dapat menambah wawasan ilmu pengetahuan, khususnya bidang ilmu komputer dalam instalasi nilai-nilai Islam yang terpadu dalam proses pembelajaran di lingkungan Universitas Islam Negeri Sumatera Utara.

Dalam penulisan laporan ini, saya sangat menyadari bahwa masih banyak kekurangan yang perlu perbaikan dan penyempurnaan, sumbangan pemikiran yang membangun sangat kami harapkan dari rekan-rekan sejawat terutama dari dosen-dosen senior. Semoga laporan penelitian ini dapat diperkaya melalui evaluasi terus menerus. Terimakasih kepada anggota peneliti dan tim penelitian yang sudah fokus dalam penyelesaian laporan ini dan pastinya sangat berperan dalam proses penelitian dari tahap awal hingga akhir. Dan kepada semua pihak yang tidak bisa disebutkan satu persatu, ribuan terimakasih diucapkan atas bantuan yang kami terima selama proses penelitian ini.

Medan, Nopember 2020  
Penulis



**Ilka Zufria, M.Kom**  
NIP. 198506042015031006

## DAFTAR ISI

LEMBAR PENGESAHAN .....	ii
SURAT PERNYATAAN BEBAS PLAGIASI .....	iii
ABSTRAK.....	v
ABSTRACT.....	vi
KATA PENGANTAR .....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xii
PENDAHULUAN .....	1
A. Latar Belakang .....	1
B. Rumusan Permasalahan .....	3
C. Tujuan Penelitian .....	3
D. Signifikansi .....	3
KAJIAN PUSTAKA.....	4
A. <i>Coronavirus Disease 2019 (COVID-19)</i> .....	4
B. <i>Machine Learning</i> .....	7
C. <i>Dataset</i> .....	8
D. <i>Support Vector Machine</i> .....	9
E. <i>Polynomial Regression</i> .....	11
F. <i>Bayesian Ridge Regression</i> .....	12
G. Python .....	13
H. Jupyter Notebook .....	15
I. <i>Flowchart</i> .....	15
METODE PENELITIAN .....	17
A. Perencanaan.....	17
B. Analisa Kebutuhan.....	18
C. Pemodelan dan Perancangan.....	18
D. Hasil Prediksi Algoritma.....	19
HASIL DAN PEMBAHASAN .....	21
A. Analisis Kebutuhan Sistem dan Informasi.....	21
1. Analisis Kebutuhan Data.....	21

2. Testing dan Implementasi .....	38
PENUTUP .....	56
A. Kesimpulan .....	56
B. Penutup.....	56
DAFTAR REFERENSI .....	58

## DAFTAR GAMBAR

Gambar 2. 1 Tahapan peristiwa COVID-19 di dunia .....	5
Gambar 2. 2 Prosedur penetapan pasien terjangkit COVID-19.....	6
Gambar 2. 3 Penentuan hyperplane optimal .....	10
Gambar 3. 1 Diagram Alur Metode Penelitian .....	20
Gambar 4. 1 Library Python pada Jupyter Notebook .....	29
Gambar 4. 2 Input Dataset Kasus Covid-19 di Indonesia.....	29
Gambar 4. 3 Membersihkan Nilai Kosong dari Data .....	29
Gambar 4. 4 Menetapkan Baris Key pada Data.....	30
Gambar 4. 5 Memisahkan Isi Data dengan Objek pada Data .....	30
Gambar 4. 6 Mendefinisikan Variabel Perhitungan .....	31
Gambar 4. 7 Fungsi Menghitung Rate Kasus .....	32
Gambar 4. 8 Fungsi Menghitung Peningkatan Kasus Harian.....	33
Gambar 4. 9 Membentuk Array yang Baru .....	34
Gambar 4. 10 Membentuk Array yang Baru untuk Prediksi .....	34
Gambar 4. 11 <i>Flowchart</i> SVM .....	35
Gambar 4. 12 <i>Flowchart</i> Polynomial .....	36
Gambar 4. 13 <i>Flowchart</i> Bayesian .....	37
Gambar 4. 14 Implementasi Visualisasi Plot Kasus Covid-19 .....	39
Gambar 4. 15 Visualisasi Kasus Covid-19 .....	39
Gambar 4. 16 Visualisasi Kasus Terkonfirmasi 34 Provinsi .....	40
Gambar 4. 17 Visualisasi Kasus Sembuh 34 Provinsi .....	41
Gambar 4. 18 Visualisasi Kasus Meninggal 34 Provinsi.....	42
Gambar 4. 19 Implementasi Visualisasi Plot 10 Provinsi Terbanyak .....	43
Gambar 4. 20 Visualisasi Kasus Covid-19 pada 10 Provinsi .....	43
Gambar 4. 21 Implementasi Visualisasi Covid-19 dalam Perawatan.....	44
Gambar 4. 22 Visualisasi Kasus Covid-19 dalam Perawatan.....	45
Gambar 4. 23 Implementasi Visualisasi Covid-19 yang Sembuh .....	45
Gambar 4. 24 Visualisasi Kasus Covid-19 yang Sembuh .....	46
Gambar 4. 25 Implementasi Visualisasi Covid-19 yang Meninggal .....	46
Gambar 4. 26 Visualisasi Kasus Covid-19 yang Meninggal .....	47
Gambar 4. 27 Implementasi Visualisasi Tingkat Kesembuhan .....	47

Gambar 4. 28 Visualisasi Tingkat Kesembuhan.....	48
Gambar 4. 29 Implementasi Visualisasi Tingkat Kematian .....	48
Gambar 4. 30 Visualisasi Tingkat Kesembuhan.....	49
Gambar 4. 31 Parameter SVM.....	49
Gambar 4. 32 Proses Prediksi SVM .....	49
Gambar 4. 33 Evaluasi SVM .....	50
Gambar 4. 34 Implementasi Visualisasi Prediksi SVM.....	51
Gambar 4. 35 Hasil Prediksi Kasus Covid-19 SVM .....	51
Gambar 4. 36 Parameter dan Transform Polynomial .....	52
Gambar 4. 37 Evaluasi Polynomial .....	52
Gambar 4. 38 Implementasi Visualisasi Prediksi Polynomial.....	52
Gambar 4. 39 Hasil Prediksi Kasus Covid-19 Polynomial.....	53
Gambar 4. 40 Mencari Parameter Bayesian .....	54
Gambar 4. 41 Parameter Bayesian.....	54
Gambar 4. 42 Evaluasi Bayesian .....	54
Gambar 4. 43 Hasil Prediksi Kasus Covid-19 Bayesian .....	55

## DAFTAR TABEL

Tabel 2. 1 Simbol <i>Flowchart</i> .....	16
Tabel 4. 1 Sampel Data Kasus Terkonfirmasi COVID-19 dari 34 Provinsi di Indonesia.....	21
Tabel 4. 2 Sampel Data Kasus Sembuh COVID-19 dari 34 Provinsi di Indonesia.....	23
Tabel 4. 3 Sampel Data Kasus Meninggal COVID-19 dari 34 Provinsi di Indonesia.....	26

# BAB I

## PENDAHULUAN

### A. Latar Belakang

Pada Desember 2019, kasus pneumonia misterius pertama kali dilaporkan di Wuhan, Provinsi Hubei. Sumber penularan kasus ini masih belum diketahui pasti, tetapi kasus pertama dikaitkan dengan pasar ikan di Wuhan. Tanggal 18 Desember hingga 29 Desember 2019, terdapat lima pasien yang dirawat dengan *Acute Respiratory Distress Syndrome* (ARDS).<sup>2</sup> Sejak 31 Desember 2019 hingga 3 Januari 2020 kasus ini meningkat pesat, ditandai dengan dilaporkannya sebanyak 44 kasus. Tidak sampai satu bulan, penyakit ini telah menyebar di berbagai provinsi lain di China, Thailand, Jepang, dan Korea Selatan. Sampel yang diteliti menunjukkan etiologi coronavirus baru. Awalnya, penyakit ini dinamakan sementara sebagai 2019 *novel coronavirus* (2019-nCoV), kemudian WHO mengumumkan nama baru pada 11 Februari 2020 yaitu *Coronavirus Disease* (COVID-19) yang disebabkan oleh virus *Severe Acute Respiratory Syndrome Coronavirus-2* (SARS-CoV-2)

Virus ini dapat ditularkan dari manusia ke manusia dan telah menyebar secara luas di China dan lebih dari 215 negara dan teritori lainnya. Pada 12 Maret 2020, WHO mengumumkan COVID-19 sebagai pandemik.<sup>6</sup> Hingga tanggal 28 Oktober 2020, terdapat total keseluruhan kasus positif corona di dunia kini mencapai angka 44.221.500. Adapun total 1.171.142 kasus kematian akibat Covid-19 yang terkonfirmasi pada pagi ini, dengan 32.429.336 di antaranya dinyatakan telah sembuh. Sementara di Indonesia sudah ditetapkan total kasus positif tercatat 400.483, sembuh 325.793, meninggal 13.612.

Dampak wabah Covid-19 terlihat hampir di seluruh sektor kehidupan masyarakat. Aktivitas sosial dilarang dan ditunda sementara waktu, melemahnya ekonomi, pelayanan transportasi dikurangi dan diatur dengan ketat, pariwisata ditutup, pusat perbelanjaan sepi pengunjung dan ditutup sektor informal seperti; ojek *online*, supir angkutan umum, pedagang kaki lima, pedagang keliling, UMKM dan kuli kasar penurunan pendapatan. Pusat-pusat perdagangan, seperti mal, pasar tanah abang yang biasanya ramai dikunjungi oleh masyarakat mendadak sepi dan saat ini ditutup sementara. Sektor pariwisata mengalami penurunan, pemerintah menutup

tempat wisata, tempat hiburan. Bekerja dan belajar pun dilakukan di rumah secara *online*.

Laju penularan yang sangat cepat oleh karena mobilitas masyarakat dunia yang tinggi menyebabkan kesiapan negara-negara di dunia dipandang tidak cukup dalam mengantisipasi penularan yang terjadi. Adanya pendataan yang cepat dari pihak pemerintahan seperti Kementerian Kesehatan membantu untuk mencermati penyebaran virus ini pada berbagai wilayah di Indonesia. Beberapa lembaga statistik di dunia juga melakukan pendataan yang selalu diperbaharui, misalkan (OrtizOspina, 2020). Demikian pula kepedulian sosial muncul atas pengaruh yang ditimbulkan akibat perkembangan laju COVID-19 yang meningkat sangat cepat. Pendataan secara cepat dan uptodate ini telah dilakukan dan diharapkan dapat memberikan penjelasan kepada semua pihak untuk dapat segera melakukan tindakan pencegahan dimana sekarang diberlakukan *social distancing* (SD) ataupun *physical distancing* (PD) yaitu pembatasan interaksi sosial antar manusia maupun kegiatan yang melibatkan kelompok.

Sejak 16 Maret 2020 di Indonesia SD atau PD diberlakukan dengan diberi nama PSBB yakni Pembatasan Sosial Berskala Besar. Telah direkomendasikan oleh penelitian bagaimana SD ataupun PD berpengaruh terhadap memperlambat laju kasus COVID-19 di Wuhan (Prem et al., 2020) serta di seluruh dunia turut melakukan SD (Pambuccian, 2020). Aksi SD ini telah dilakukan dan laju pertumbuhan pasien yang terkonfirmasi COVID-19 perlu dipelajari untuk melihat ada tidaknya pengaruh yang signifikan baik di Indonesia. Pada penelitian ini menunjukkan penelusuran pendataan kasus yang dikonfirmasi pada 34 provinsi di Indonesia dimulai dari 1 September 2020 sampai dengan 28 Oktober 2020. Dengan mempelajari data COVID-19 yang sudah tersedia hingga saat ini, maka laju total kasus di Indonesia dapat diprediksi dalam beberapa waktu ke depan. Oleh karena itulah penelitian ini dilakukan, yaitu mendapatkan prediksi pada banyaknya penambahan kasus COVID-19 di Indonesia selama 60 hari ke depan.

Metode yang digunakan adalah SVM (*Support Vector Machine*), *Polynomial Regression*, dan regresi *Bayesian Ridge*. Ketiga metode ini biasa digunakan dalam melakukan pengklasifikasian (Min-Wei et al., 2018; Lamidi, 2018; Mckinney, 2018) sebagai metode yang banyak digunakan pada *Machine Learning* untuk pengolahan *big data* (Demidova et al., 2016). Penelitian ini dilakukan dengan memanfaatkan penggunaan bahasa pemrograman Python 3 dengan *Jupyter Notebook*. Berdasarkan pemaparan yang telah dijelaskan sebelumnya dibuatlah penelitian ini yang berjudul **Pemanfaatan *Machine Learning* Dengan Algoritma *Support Vector***

## ***Machine Dan Polynomial Regression Untuk Memprediksi Kasus Covid-19 Di Indonesia.***

### **B. Rumusan Permasalahan**

Berdasarkan latar belakang di atas maka dapat dirumuskan permasalahan penelitian sebagai berikut:

- a. Bagaimanakah implementasi algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* dalam memprediksi banyak kasus covid-19 di Indonesia?
- b. Bagaimanakah implementasi algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* terhadap *machine learning*?
- c. Bagaimanakah implementasi algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* menggunakan bahasa pemrograman Python dengan *Jupyter Notebook*?

### **C. Tujuan Penelitian**

Penelitian ini bertujuan untuk:

- a. Mengetahui implementasi algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* dalam memprediksi banyak kasus covid-19 di Indonesia.
- b. Mengetahui implementasi algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* terhadap *machine learning*.
- c. Mengetahui implementasi algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* menggunakan bahasa pemrograman Python dengan *Jupyter Notebook*

### **D. Signifikansi**

Penelitian ini dianggap penting karena diharapkan akan memberi kontribusi sebagai berikut:

- a. Mengetahui laju perkembangan COVID-19 di negara Indonesia.
- b. Mendapatkan prediksi banyak penambahan kasus baru dari COVID-19 di Indonesia.
- c. *Outcome* berupa publikasi karya ilmiah hasil penelitian di prosiding international terindex scopus

## BAB II

### KAJIAN PUSTAKA

#### A. *Coronavirus Disease 2019 (COVID-19)*

Di awal tahun 2020 ini, seluruh dunia dikagetkan dengan kejadian infeksi berat dengan penyebab yang belum diketahui, yang berawal dari laporan dari Cina kepada *World Health Organization* (WHO) terdapat 44 pasien pneumonia yang berat di suatu wilayah yaitu Kota Wuhan, Provinsi Hubei, China, tepatnya di hari terakhir tahun 2019 Cina. Dugaan awal hal ini terkait dengan pasar basah yang menjual ikan, hewan laut dan berbagai hewan lain. Pada 10 Januari 2020 penyebabnya mulai teridentifikasi dan didapatkan kode genetiknya yaitu virus corona bau [<sup>1</sup>] Penelitian selanjutnya menunjukkan hubungan yang dekat dengan virus corona penyebab *Severe Acute Respiratory Syndrome* (SARS) yang mewabah di Hongkong pada tahun 2003, hingga WHO menamakannya sebagai novel corona virus (nCoV-19) [<sup>2</sup>]. Tidak lama kemudian mulai muncul laporan dari provinsi lain di Cina bahkan di luar Cina, pada orang-orang dengan riwayat perjalanan dari Kota Wuhan dan Cina yaitu Korea Selatan, Jepang, Thailand, Amerika Serikat, Makau, Hongkong, Singapura, Malaysia hingga total 25 negara termasuk Prancis, Jerman, Uni Emirat Arab, Vietnam dan Kamboja. Ancaman pandemik semakin besar ketika berbagai kasus menunjukkan penularan antar manusia (*human to human transmission*) pada dokter dan petugas medis yang merawat pasien tanpa ada riwayat berpergian ke pasar yang sudah ditutup [<sup>3</sup>].

Laporan lain menunjukkan penularan pada pendamping wisatawan Cina yang berkunjung ke Jepang disertai bukti lain terdapat penularan pada kontak serumah pasien di luar Cina dari pasien terkonfirmasi dan pergi ke Kota Wuhan kepada pasangannya di Amerika Serikat. Penularan langsung antar manusia (*human to human transmission*) ini menimbulkan

---

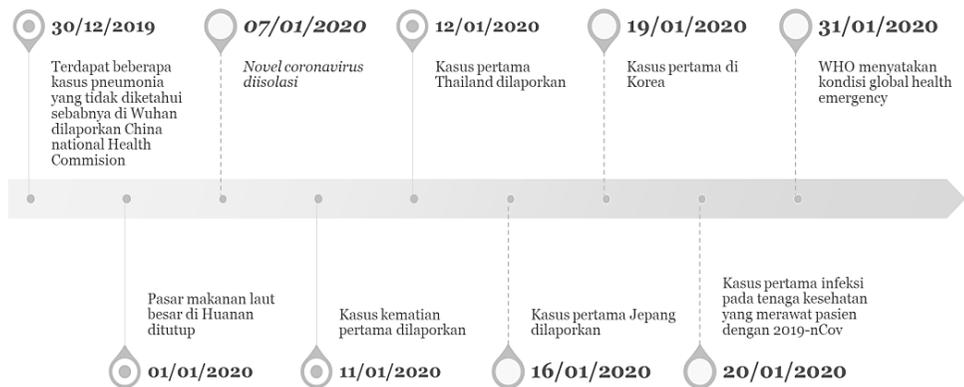
<sup>1</sup> D. Handayani, D. R. Hadi, F. Isbaniah, E. Burhan, and H. Agustin, "Penyakit Virus Corona 2019," *J. Respirologi Indones.*, vol. 40, no. 2, pp. 119–129, 2020, [Online].

Available: <https://jurnalrespirologi.org/index.php/jri/article/view/101>

<sup>2</sup> C. Ceraolo and F. M. Giorgi, "Genomic variance of the 2019-nCoV coronavirus," *J. Med. Virol.*, vol. 92, no. 5, pp. 522–528, 2020, doi: 10.1002/jmv.25700.

<sup>3</sup> Peng *et al.*, "A pneumonia outbreak associated with a new coronavirus of probable bat origin," *Nature*, vol. 579, no. 7798, pp. 270–273, 2020, [Online]. Available: <https://www.nature.com/articles/s41586-020-2012-7?rel=outbound>.

peningkatan jumlah kasus yang luar biasa hingga pada akhir Januari 2020 didapatkan peningkatan 2000 kasus terkonfirmasi dalam 24 jam. Pada akhir Januari 2020 WHO menetapkan status Global Emergency pada kasus virus Corona ini dan pada 11 Februari 2020 WHO menamakannya sebagai COVID-19 [4].

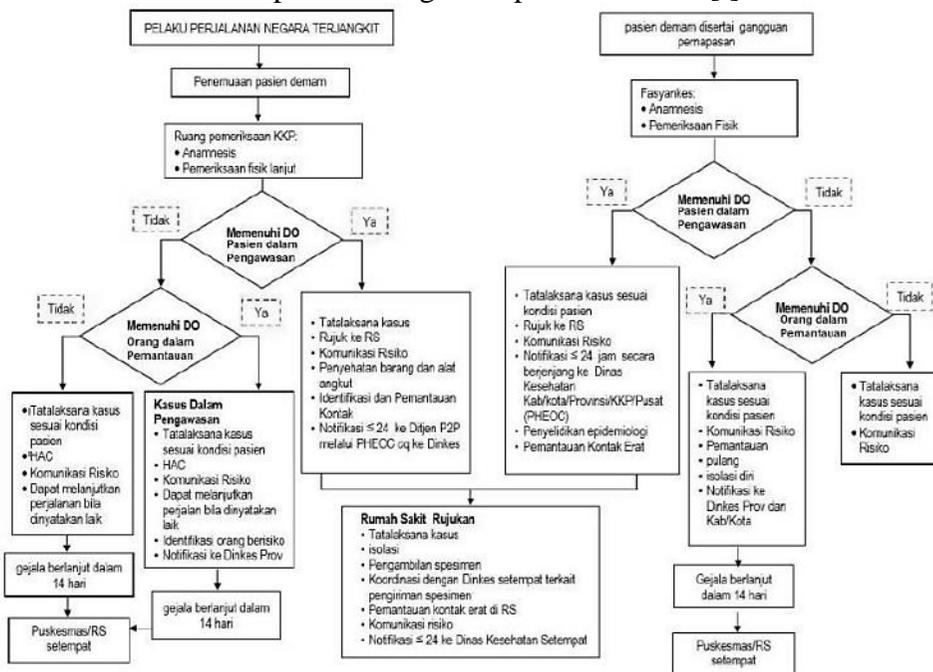


Gambar 2. 1 Tahapan peristiwa COVID-19 di dunia

Gambar di atas menunjukkan waktu peristiwa masuknya COVID-19 dan mengalami persebaran di beberapa negara di dunia. Virus corona merupakan zoonosis, sehingga terdapat kemungkinan virus berasal dari hewan dan ditularkan ke manusia. Pada COVID-19 belum diketahui dengan pasti proses penularan dari hewan ke manusia, tetapi data filogenetik memungkinkan COVID-19 juga merupakan zoonosis. Perkembangan data selanjutnya menunjukkan penularan antar manusia (*human to human*), yaitu diprediksi melalui tetesan dan kontak dengan virus yang dikeluarkan dalam droplet. Hal ini sesuai dengan kejadian penularan kepada petugas kesehatan yang merawat pasien COVID-19, disertai bukti lain penularan di luar Cina dari seorang yang datang dari Kota Shanghai, Cina ke Jerman dan diiringi penemuan hasil positif pada orang yang ditemui dalam kantor. Pada laporan kasus ini bahkan dikatakan penularan terjadi pada saat kasus indeks belum mengalami gejala (asimtomatik) atau masih dalam masa inkubasi. Laporan lain mendukung penularan antar manusia adalah laporan 9 kasus penularan langsung antar manusia di luar Cina dari kasus index ke orang kontak erat yang tidak memiliki riwayat perjalanan manapun.

Penularan ini terjadi umumnya melalui tetesan dan kontak dengan virus kemudian virus dapat masuk ke dalam mukosa yang terbuka. Suatu analisis mencoba mengukur laju penularan berdasarkan masa inkubasi, gejala dan durasi antara gejala dengan pasien yang diisolasi. Analisis

tersebut mendapatkan hasil penularan dari 1 pasien ke sekitar 3 orang di sekitarnya, tetapi kemungkinan penularan di masa inkubasi menyebabkan masa kontak pasien ke orang sekitar lebih lama sehingga risiko jumlah kontak tertular dari 1 pasien mungkin dapat lebih besar [4].



Gambar 2. 2 Prosedur penetapan pasien terjangkit COVID-19 [7]

Gambar 2 merupakan prosedur seseorang ditetapkan sebagai pasien COVID-19 menurut pedoman dari Kementerian Kesehatan RI yang diterapkan di Indonesia. Prinsip tatalaksana secara keseluruhan menurut rekomendasi WHO yaitu: Triase : identifikasi pasien segera dan pisahkan pasien dengan *severe acute respiratory infection* (SARI) dan dilakukan dengan memperhatikan prinsip pencegahan dan pengendalian infeksi (PPI) yang sesuai, terapi suportif dan monitor pasien, pengambilan contoh uji untuk diagnosis laboratorium, tata laksana secepatnya pasien dengan hipoksemia atau gagal nafas dan *acute respiratory distress syndrome* (ARDS), syok sepsis dan kondisi kritis lainnya [5].

<sup>4</sup> N. Zhu *et al.*, “A Novel Coronavirus from Patients with Pneumonia in China, 2019,” *N. Engl. J. Med.*, vol. 382, no. 8, pp. 727–733, 2020, doi: 10.1056/nejmoa2001017.

<sup>5</sup> WHO, *Clinical management of severe acute respiratory infection when novel coronavirus (2019-nCoV) infection is suspected*. Geneva: WHO Press, 2020.

Hingga saat ini tidak ada terapi spesifik anti virus COVID-19 dan anti virus corona lainnya. Beberapa peneliti membuat hipotesis penggunaan baricitinib, suatu inhibitor janus kinase dan regulator endositosis sehingga masuknya virus ke dalam sel terutama sel epitel alveolar. Pengembangan lain adalah penggunaan rendesivir yang diketahui memiliki efek antivirus RNA dan kombinasi klorokuin, tetapi keduanya belum mendapatkan hasil. Vaksinasi juga belum dapat dipastikan sampai sekarang sehingga tata laksana utama pada pasien adalah terapi suportif disesuaikan kondisi pasien, terapi cairan adekuat sesuai kebutuhan, terapi oksigen yang sesuai derajat penyakit mulai dari penggunaan kanul oksigen, masker oksigen. Bila dicurigai terjadi infeksi ganda diberikan antibiotika spektrum luas. Bila terdapat perburukkan klinis atau penurunan kesadaran pasien akan dirawat di ruang isolasi intensif (ICU) di rumah sakit rujukan dengan alur seperti prosedur pada Gambar 2 [6].

## **B. *Machine Learning***

*Machine Learning* (ML) merupakan bagian artificial intelligence sebagai suatu keilmuan yang memiliki cakupan serta rancangan dalam mengembangkan algoritma yang memberikan kemungkinan bahwa komputer dapat belajar berdasarkan data empirik. Sejak komputer pertama kali ditemukan, kemudian mengembangkan penggunaannya, para peneliti berpikir mengenai kemungkinan agar komputer dapat belajar sebagaimana manusia. Komputer belajar dari data-data yang ada kemudian menemukan cara baru untuk memecahkan suatu masalah. Pembelajaran dengan mesin membutuhkan set data yang benar bisa dilakukan penerapan terhadap tahapan pembelajaran. Data besar (*big data*) dapat membantu meningkatkan akurasi model pembelajaran mesin. Dengan data besar, dimungkinkan untuk memvirtualisasikan data sehingga dapat disimpan dengan cara yang paling efisien dan hemat. Selain itu, peningkatan kecepatan dan keandalan jaringan telah menghilangkan keterbatasan fisik lainnya karena dapat mengelola sejumlah besar data pada kecepatan yang dapat diterima [7]. *Machine Learning* memiliki banyak algoritma sesuai dengan kebutuhan seperti kluster, klasifikasi, dan prediksi. Sejumlah algoritma paling dasar antara lain *k-nearest neighbor*, *a simple classifier*, *k-means*, dan perceptron.

---

<sup>6</sup> Direktorat Jenderal P2P Kemenkes RI, “Pedoman Kesiapsiagaan Menghadapi Infeksi Novel Coronavirus (2019-nCoV).”

<https://covid19.kemkes.go.id/downloads/#.Xtva> (accessed Nov. 04, 2020).

<sup>7</sup> J. Hurwitz and D. Kirsch, *Machine learning*. Hoboken, 2016

Machine learning dikelompokkan menjadi tiga kategori utama, antara lain ialah [8]:

1. *Supervised Learning*, yaitu pembelajaran yang diawasi dan memerlukan penentuan target *ouput* yang dimana menggunakan data yang telah ada untuk diolah.
2. *Unsupervised Learning*, yaitu pembelajaran tanpa diawasi dan tanpa diperlukan target yang menjadi output. Metode ini tidak bisa menentukan suatu hal yang dihasilkan berdasarkan keinginan ketika tahapan pembelajaran. capaian metode ini untuk melakukan pengelompokan sejumlah unit yang memiliki kemiripan pada suatu lingkup.
3. *Reinforcement Learning*, yaitu pembelajaran dengan interaksi langsung pada lingkungan untuk mengambil keputusan yang akan memaksimalkan keuntungan dan meminimalkan risiko. Dalam prosesnya, agen belajar melalui pengalaman tentang lingkungan dan mengeksplorasi seluruh keadaan.

### C. Dataset

*Dataset* atau set data merupakan kumpulan instansi sebuah informasi yang isinya mengandung atribut. Set data dapat menyimpan informasi seperti catatan medis penyakit, catatan asuransi, perpustakaan makro, dan berbagai macam jenis data lainnya yang dibutuhkan oleh sistem itu sendiri. Kumpulan data ini yang kemudian di proses ke dalam algoritma mesin pembelajaran untuk melatih model yang dinamakan dengan *training dataset*. Namun, jika terdapat nilai yang hilang dalam suatu kumpulan data dapat menyebabkan masalah selama fase *training* dan klasifikasi. Beberapa kemungkinan alasan yang menyebabkan hilangnya data adalah data yang ada tidak relevan dengan entri, penghapusan data karena penyimpangan, dan adanya kerusakan data. Masalah adanya data yang hilang dapat diatasi dengan pendekatan seperti penambang data mengabaikan data yang dihilangkan, menukar seluruh nilai yang dihilangkan dengan global atau dengan rata-rata fitur kelas yang diberikan, mengamati sampel secara manual dengan menghilangkan nilai dan memasukkan nilai yang layak atau

---

<sup>8</sup>O. Pentakalos, *Introduction to machine learning*. New York: The Press Syndicate of The Univeristy of Cambridge, 2019

mungkin [<sup>9</sup>]. Set data penelitian ini diambil dari laman resmi Kementerian Kesehatan Republik Indonesia.

#### **D. Support Vector Machine**

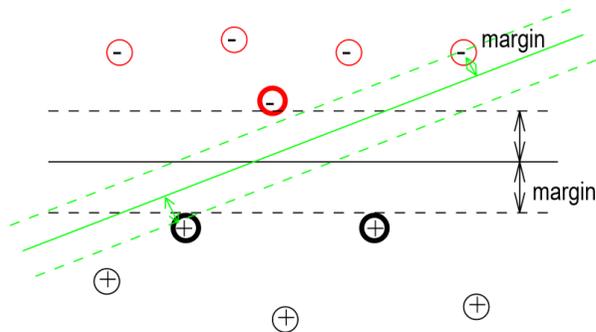
*Support Vector Machine* (SVM) atau mesin pendukung vektor dikembangkan dalam kerangka teori pembelajaran statistik oleh Vapnik pada tahun 1992 [<sup>10</sup>] dan telah berhasil diterapkan pada sejumlah aplikasi, mulai dari prediksi waktu proses, pengenalan wajah, pemrosesan data biologis untuk diagnosis medis. SVM merupakan algoritma pembelajaran dengan pengawasan yang penggunaannya dalam pengklasifikasian serta regresi dan belajar dengan cara memberikan contoh berupa data untuk menetapkan jenis label pada suatu objek. Misalnya, SVM dapat belajar untuk mengenali aktivitas kartu kredit yang melakukan kecurangan dengan memeriksa ratusan atau bahkan ribuan laporan aktivitas kartu kredit yang curang atau melakukan pemalsuan. SVM juga dapat belajar mengenali angka tulisan tangan dengan memeriksa banyak koleksi gambar yang dipindai dari angka tulisan tangan yang sudah dikenali. SVM juga telah digunakan pada banyak peneliti dalam memprediksi kasus COVID-19 di dunia. Properti khusus SVM ialah dapat meminimalisir kesalahan pengklasifikasian empirik serta memaksimalan terhadap margin geometrik. Dengan demikian, SVM dikatakan sebagai *Maximum Margin Classifiers*. SVM memiliki kelebihan diantaranya dalam menentukan jarak diantara dua set data yang berasal pada dua kelas yang memiliki perbedaan. SVM akan menemukan *hyperplane* sebagai pemisah maksimal seoptimal mungkin sebagai solusi untuk masalah pembelajaran. Formulasi SVM yang paling sederhana adalah linear, dimana *hyperplane* terletak pada ruang data input [<sup>11</sup>].

---

<sup>9</sup> M. Singh, S. Sharma, and A. Kaur, "Performance Analysis of Decision Trees," *Int. J. Comput. Appl.*, vol. 71, no. 19, pp. 975–8887, 2013, [Online]. Available: <https://www.ijcaonline.org/archives/volume71/number19/12593-9232>

<sup>10</sup> V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

<sup>11</sup> M. Furqan, "Face Recognition Using Eigenfaces and Smooth Support Vector Machine," Universiti Malaysia Pahang, 2011.



Gambar 2. 3 Penentuan hyperplane optimal [12]

Gambar 2.3 menunjukkan anggota pada dua buah kelas yakni *plus* (+) serta *minus* (-). Permasalahan pengklasifikasian bisa diselesaikan melalui menentukan garis *hyperplane* yang menjadi pemisah antar dua kelompok melalui penghitungan *margin* antar dua kelas tersebut kemudian menghasilkan titik optimum *hyperplane* tersebut. *Margin* adalah jarak dari *hyperplane* terhadap anggota paling dekat oleh tiap-tiap kelas. Anggota terdekat dikatakan *support vector*. Garis hitam di gambar 2.4 disebut dengan *hyperplane* optimal. Hasil dari penentuan *hyperplane* adalah pokok tahapan dari SVM [15].

Formulasi kelas yang memisah dengan baik oleh *hyperplane* terbentuk dengan formula :

$$\vec{w} \cdot \vec{x} + b = 0 \quad (2.1)$$

Anggota  $\vec{x}$  yang tergolong pada kelas *plus* (+) dapat dirumuskan :

$$\vec{w} \cdot \vec{x} + b \geq +1 \quad (2.2)$$

Anggota  $\vec{x}$  yang tergolong pada kelas *minus* (-) dapat dirumuskan :

$$\vec{w} \cdot \vec{x} + b \leq -1 \quad (2.3)$$

Formulasi dari SVM klasifikasi yaitu :

$$\min_f \|f\|_k^2 + C \sum_{i=1}^l |1 - y_i f(x_i)| \quad (2.4)$$

dimana C merupakan parameter regulasi yang berfungsi sebagai kontrol antara margin dan kesalahan klasifikasi [13]. SVM hanya

<sup>12</sup> J. Weston and A. Ben-Hur, "A User's Guide to Support Vector Machine," *Data Min. Tech. Life Sci.*, vol. 609, pp. 223–39, 2010, doi: 10.1007/978-1-60327-241-4\_13.

<sup>13</sup> T. Evgeniou and M. Pontil, "Support vector machines: Theory and applications," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell.*

menyimpan sebagian kecil data latih untuk digunakan pada saat prediksi. Hal inilah yang menjadi kelebihan SVM karena tidak semua data latih dilibatkan pada saat pelatihan [14].

### E. *Polynomial Regression*

Regresi adalah suatu metode analisis statistik yang digunakan untuk melihat pengaruh antara dua atau lebih variabel. Hubungan variabel tersebut bersifat fungsional yang diwujudkan dalam suatu model matematis. Pada analisis regresi, terbagi menjadi beberapa jenis yakni Regresi Sederhana (Linier Sederhana dan Nonlinier Sederhana) dan Regresi Berganda atau Nonlinier Berganda. Apabila dalam persamaan Regresi Linier mencakup lebih dari dua prediktor atau variabel bebas maka regresi tersebut dinamakan Regresi Linier Berganda (*Multiple Linear Regression*).

Regresi polinomial adalah suatu bentuk analisis regresi di mana hubungan antara variabel bebas  $x$  dan variabel terikat  $y$  dimodelkan sebagai polinomial derajat ke-  $n$  dalam  $x$ . Regresi polinomial cocok dengan hubungan nonlinier antara nilai  $x$  dan rata-rata bersyarat yang sesuai dari  $y$ , dilambangkan dengan  $E(y|x)$ . Meskipun regresi polinomial menyesuaikan model nonlinier dengan data, sebagai masalah estimasi statistik, ia linier, dalam arti bahwa fungsi regresi  $E(y|x)$  linier dalam parameter yang tidak diketahui yang diperkirakan dari data. Untuk alasan ini, regresi polinomial dianggap sebagai kasus khusus dari regresi linier berganda. Variabel penjelas (independen) yang dihasilkan dari ekspansi polinomial dari variabel “*baseline*” dikenal sebagai istilah tingkat yang lebih tinggi. Variabel semacam itu juga digunakan dalam pengaturan klasifikasi [15].

Regresi polinomial merupakan model regresi linier yang dibentuk dengan menjumlahkan pengaruh masing-masing variabel prediktor ( $X$ )

---

*Lect. Notes Bioinformatics*), vol. 2049 LNAI, pp. 249–257, 2001, doi: 10.1007/3-540-44673-7\_12.

<sup>14</sup> E. Prasetyo, *Data Mining Konsep dan Aplikasi Menggunakan Matlab*. Yogyakarta: Andi, 2012.

<sup>15</sup> Y. W. Chang, C. J. Hsieh, K. W. Chang, M. Ringgaard, and C. J. Lin, “Training and testing low-degree polynomial data mappings via linear SVM,” *J. Mach. Learn. Res.*, vol. 11, pp. 1471–1490, 2010, [Online]. Available: <https://www.jmlr.org/papers/volume11/chang10a/chang10a.pdf>.

yang dipangkatkan meningkat sampai orde ke- $n$ . Secara umum, model regresi polinomial ditulis dalam bentuk [<sup>16</sup>]:

$$y = b_0 + b_1X + b_2X^2 + \dots + b_nX^n + \varepsilon \quad (2.5)$$

#### F. *Bayesian Ridge Regression*

Model linear regresi Bayesian Ridge merupakan perpaduan antara regresi Ridge dan linear regresi Bayesian [<sup>17</sup>]. Regresi Ridge menggunakan regularisasi dengan norm L2 sedangkan regresi Bayesian model didefinisikan secara probabilistik dengan prior eksplisit untuk parameter-parameter. Pemilihan prior berpengaruh dalam regularisasi. Jadi keduanya tidak sama karena ridge regresi adalah suatu model regresi, dan pendekatan Bayesian adalah langkah yang lebih umum untuk mendefinisikan dan mengestimasi model statistik yang dapat diaplikasikan pada modelmodel yang berbeda. Diketahui bahwa regresi standar dalam 1 variabel dapat ditulis sebagai

$$y = \beta_0 + \beta_1X + \varepsilon \quad (2.6)$$

Untuk mengubah regresi standar ini dalam regresi Bayesian maka regresi standar tersebut ditulis dalam model probabilistik dalam bentuk

$$\begin{aligned} \mu_i &= \alpha + \beta x_i \\ y_i &\sim N(\mu_i, \sigma) \end{aligned} \quad (2.7)$$

yang berarti variabel dependen  $Y$  mengikuti distribusi normal dengan parameter  $\mu_i$  yaitu fungsi linear  $X$  yang diparameterisasi oleh  $\alpha$ ,  $\beta$  dan deviasi standard  $\sigma$ . Jika model linear dengan OLS (*Ordinary Least Square*) digunakan, probabilitas tidak perlu dipermasalahkan karena nilai optimal  $\alpha$ ,  $\beta$  yang dicari dengan meminimalkan kuadrat error dalam pencocokan data dengan pendekatan. Sebaliknya, model tersebut memaksimalkan fungsi *likelihood*.

$$\arg \max_{\alpha, \beta, \sigma} \prod_{i=1}^n \mathcal{N}(y_i; \alpha + \beta x_i, \sigma) \quad (2.8)$$

---

<sup>16</sup> J. S. Malensang, H. Komalig, and D. Hatidja, "Pengembangan Model Regresi Polinomial Berganda Pada Kasus Data Pemasaran," *J. Ilm. Sains*, vol. 12, no. 2, p. 149, 2013, doi: 10.35799/jis.12.2.2012.740.

<sup>17</sup> S. D. Permai and H. Tanty, "Linear regression model using bayesian approach for energy performance of residential building," *Procedia Comput. Sci.*, vol. 135, pp. 671–677, 2018, doi: 10.1016/j.procs.2018.08.219.

dimana  $\mathcal{N}$  adalah fungsi densitas dari distribusi normal yang dihitung pada titik-titik  $y_i$  dengan parameter  $\alpha + \beta x_i$  dan  $\sigma$ .

Pada pendekatan Bayesian, sebagai pengganti memaksimumkan fungsi likelihood, diasumsikan distribusi prior untuk parameter dan menggunakan teorema Bayesian posterior  $\propto$  likelihood x prior.

Fungsi likelihood sama dengan persamaan (3), tetapi yang berubah adalah bahwa distribusi prior diasumsikan untuk parameter-parameter  $\alpha$ ,  $\beta$ ,  $\sigma$  yang diduga dan melibatkannya dalam persamaan

$$f(\alpha, \beta, \sigma | Y, X) \propto \prod_{i=1}^n \mathcal{N}(y_i; \alpha\beta x_i, \sigma) f_{\alpha}(\alpha) f_{\beta}(\beta) f_{\sigma}(\sigma) \quad (2.9)$$

## G. Python

Python merupakan bahasa pemrograman tingkat tinggi (*high level language*) yang dikembangkan oleh Guido van Rossum pada tahun 1989 dan diperkenalkan untuk pertama kalinya pada tahun 1991. Python lahir atas dasar keinginan untuk mempermudah seorang programmer dalam menyelesaikan tugas-tugasnya dengan cepat. Python dirancang untuk memberikan kemudahan kepada programmer baik dari segi efisiensi waktu, maupun kemudahan dalam pengembangan program dan dalam hal kompatibilitas dengan sistem. Python dapat digunakan untuk membuat program *stand-alone* dan pemrograman skrip (*scripting programming*) [18].

Dalam bahasa Python sendiri terdapat bahasa tingkat rendah (*low level language*) yang berhubungan dengan bahasa mesin atau assembly. Python memiliki beberapa kelebihan dibandingkan dengan bahasa pemrograman lainnya, diantaranya ialah:

- a. Pemrograman level tinggi
- b. Kemudahan dalam mempelajari
- c. Kemudahan penggunaan
- d. Kemudahan untuk dikembangkan
- e. Pengelolaan memori yang memiliki sifat dinamis
- f. Pemrograman berbasis objek
- g. Memiliki sifat *open source*
- h. Gratis untuk diakses

---

<sup>18</sup> R. Sianipar and H. Wadi, *Pemrograman Python (Teori Dan Implementasi)*. Yogyakarta: Informatika, 2015.

Bahasa pemrograman Python memiliki beberapa fitur yang dapat digunakan oleh pengembang perangkat lunak. Berikut adalah beberapa fitur yang ada pada bahasa pemrograman Python:

- a. Desain multi paradigma
- b. *Open source*
- c. Kemudahan
- d. Memiliki tata bahasa yang jernih dan mudah dipelajari
- e. Memiliki aturan *layout source code* yang memudahkan pengecekan, pembacaan kembali, dan penulisan ulang *source code* tersebut
- f. Dukungan *library*, dalam distribusi Python telah disediakan modul-modul siap pakai untuk berbagai keperluan
- g. Memiliki modular, mudah dikembangkan dengan menciptakan modul-modul baru, dimana modul-modul tersebut dapat dibangun dengan bahasa Python maupun C/C++
- h. Memiliki fasilitas pengumpulan sampah secara auto, seperti halnya pada bahasa pemrograman Java, Python memiliki fasilitas pengaturan penggunaan memori komputer sehingga para programmer tidak perlu melakukan pengaturan memori komputer secara langsung
- i. Portabilitas
- j. Diperpanjang
- k. Skalabilitas

Pada penelitian ini untuk menganalisis data yang berjumlah sangat banyak menggunakan bantuan program Jupyter Notebook dengan bahasa pemrograman python. Python telah banyak digunakan oleh para *Data Scientist* untuk mengolah data. Python juga bahasa pemrograman yang populer dan relatif mudah penggunaannya [<sup>19</sup>]. Dalam proses analisa menggunakan konsep *Machine Learning* dimana mesin mempelajari data historik dari perkembangan kasus COVID-19 di Indonesia yang telah terjadi sebelumnya untuk memprediksi banyak pertambahan kasus COVID-19 di waktu mendatang.

---

<sup>19</sup> B. D. Prasetya, F. S. Pamungkas, and I. Kharisudin, "Pemodelan dan Peramalan Data Saham dengan Analisis Time Series menggunakan Python," *Prism. Pros. Semin. Nas. Mat.*, vol. 3, pp. 714–718, 2020, [Online]. Available: <https://journal.unnes.ac.id/sju/index.php/prisma/> ISSN.

## H. Jupyter Notebook

Jupyter Notebook sebelumnya dikenal sebagai IPython Notebook yang berbasis bahasa pemrograman Python. Dukungan yang massif dari komunitas, maka secara alami dalam waktu dekat Jupyter Notebook akan berevolusi menjadi JupyterLab, dengan dilengkapi fasilitas berbagai fitur yang lebih canggih. Jupyter Notebook (file yang berekstensi .ipynb) adalah dokumen yang dihasilkan oleh Jupyter Notebook App yang berisikan kode komputer dan *rich text element* seperti paragraf, persamaan matematik, gambar dan tautan (*links*) [20].

Jupyter Notebook adalah sistem yang paling banyak digunakan untuk pemrograman literasi interaktif. Jupyter Notebook pada saat yang sama adalah dokumen pemrograman melek interaktif dan aplikasi yang mengeksekusi dokumen [21]. Sistem dirancang untuk membuat analisis data lebih mudah didokumentasikan, dibagikan, dan direproduksi. Sistem ini dirilis pada tahun 2013, dan hari ini ada lebih dari 1 juta *notebook* di GitHub. Jupyter Notebook berasal dari IPython dan, selain Python, ia mendukung berbagai bahasa pemrograman, seperti Julia, R, Javascript, dan C. Jupyter Notebook tidak hanya menggabungkan kode dan teks, tetapi juga berbagai jenis media kaya, termasuk gambar, video, dan bahkan widget interaktif yang menggabungkan HTML dan JavaScript [22].

## I. Flowchart

*Flowchart* merupakan gambaran grafik yang berasal pada sebuah tahapan-tahapan sesuai urutan-urutan prosedur melalui pengekspresian pada rangkaian berbagai simbol tertentu. *Flowchart* dibuat pada tahapan perancangan untuk memudahkan penyelesaian masalah komputasi yang memerlukan evaluasi lebih lanjut. *Flowchart* dibuat menggunakan simbol-simbol tertentu seperti pada Tabel 2.4 [23].

---

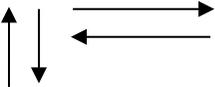
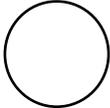
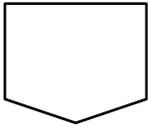
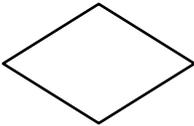
<sup>20</sup> D. Setiabudidaya, "Penggunaan Piranti Lunak Jupyter Notebook Dalam Upaya Mensosialisasikan Open Science," 2018, doi: 10.31227/osf.io/2h7q4.

<sup>21</sup> H. Shen, "Interactive notebooks: Sharing the code," *Nature*, vol. 515, no. 7525, pp. 151–152, 2014, doi: 10.1038/515151a.

<sup>22</sup> P. Fernando and E. G. Brian, "IPython: A System for Interactive Scientific Computing," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 21–29, 2007, [Online]. Available: <https://ieeexplore.ieee.org/document/4160251>.

<sup>23</sup> M. Furqan, R. Kurniawan, and K. HP, "Evaluasi Performa Support Vector Machine Classifier Terhadap Penyakit Mental," *JSINBIS*, vol. 10, no. 2, 2020

Tabel 2. 1 Simbol Flowchart

Simbol	Makna	Fungsi
	<i>Input/Output</i>	Simbol <i>input/output</i> sebagai perwakilan dari data <i>input/output</i>
	<i>Process</i>	Simbol <i>process</i> menggambarkan proses yang sedang terjadi
	<i>Flow lines</i>	Simbol <i>flowlines</i> ditujukan sebagai arah pada suatu proses
	<i>Connector</i>	Simbol <i>connector</i> menyambungkan <i>flowchart</i> dalam satu halaman
	<i>Connector</i>	Simbol <i>connector</i> yang ini untuk menyambungkan <i>flowchart</i> dalam halaman yang berbeda
	<i>Decission</i>	Simbol <i>decission</i> menyeleksi keadaan yang terjadi pada program tersebut
	<i>Predifined Process</i>	Simbol <i>predifined process</i> sebagai penunjuk pada suatu proses yang mana rincian proses tersebut berada pada tempat lain
	<i>Preparation</i>	Simbol <i>preparation</i> memberikan inisialisasi nilai
	<i>Terminal Point</i>	Simbol <i>terminal point</i> sebagai penunjuk dari berawal dan berakhirnya sebuah proses

## BAB III

### METODE PENELITIAN

Kerangka kerja dalam sebuah penelitian diperlukan untuk mencapai hasil yang optimal, karena hal itu maka peneliti akan membuat kerangka kerja menggunakan algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression*, dimana pada algoritma ini menjadi suatu bagian dari pembelajaran mesin yang diterapkan untuk mempelajari *big data* ataupun sekumpulan data yang jumlahnya sangat besar.

Adapun kerangka kerja dalam penelitian ini antara lain ialah:

#### A. Perencanaan

##### a. Ruang Lingkup masalah

Masalah pokok dalam penelitian adalah mengetahui prediksi penambahan kasus covid-19 di Indonesia melalui data dari tiap-tiap provinsi yakni sebanyak 34 provinsi. Prediksi dipelajari dengan menggunakan algoritma machine learning terhadap data kasus covid-19 di Indonesia mulai dari 1 September sampai dengan 28 Oktober 2020. Data ini didapatkan melalui laman resmi dari Kementerian Kesehatan Republik Indonesia. Algoritma pembelajaran yang dimanfaatkan terdapat tiga jenis algoritma yakni *support vector machine*, *polynomial regression*, dan *bayesian ridge regression*. Melalui hasil prediksi ini dapat diketahui seperti apa pola kemungkinan penambahan kasus persebaran virus covid 19 di Indonesia, agar masyarakat Indonesia semakin berhati-hati dan menaati protokol kesehatan yang telah diterapkan. Selain itu pada penelitian ini juga memvisualisasikan penambahan kasus terkonfirmasi, kasus yang sembuh, dan kasus meninggal berdasarkan data yang terdapat melalui laman kementerian kesehatan. Untuk menerapkan algoritma yang digunakan dengan memanfaatkan bahasa pemrograman python menggunakan jupyter notebook serta pemanfaatan sejumlah *library* yang telah tersedia untuk kasus prediksi.

##### b. Tujuan penelitian

Tujuan dari pelaksanaan penelitian ini adalah agar menerapkan pemanfaatan algoritma *machine learning* diantaranya *support vector machine*, *polynomial regression*, dan *bayesian ridge regression* untuk memprediksi kasus penambahan covid-19 di

Indonesia yang memiliki data dari 34 provinsi, di mana data tersebut didapatkan melalui laman resmi kementerian kesehatan Republik Indonesia. Melalui hasil prediksi dapat diketahui seperti apa pola kemungkinan besarnya penambahan kasus virus covid-19 di Indonesia untuk menambahkan rasa kehati-hatian bagi masyarakat Indonesia dalam upaya meminimalisir persebaran virus dan menekan angka penambahan kasus terkonfirmasi covid-19 tersebut. Menerapkan algoritma *support vector machine*, *polynomial regression*, dan *bayesian ridge regression* sebagai algoritma *machine learning* ke bahasa pemrograman python dengan menggunakan jupyter notebook.

c. Manfaat penelitian

Hasil dari penelitian ini yakni besar penambahan kasus covid-19 melalui prediksi ketiga algoritma yaitu *support vector machine*, *polynomial regression*, dan *bayesian ridge regression* selama 60 hari kedepan mulai dari data terakhir 28 Oktober 2020 dengan penggunaan data kasus covid-19 sejak 1 September 2020. Selain itu, divisualisasikan besar penambahan kasus covid-19 di Indonesia dari 1 September 2020 sampai dengan 28 Oktober 2020 untuk melihat grafik perkembangan angka kasus covid-19.

## **B. Analisa Kebutuhan**

Pada tahap analisa kebutuhan sistem adalah dengan menganalisa kebutuhan informasi fungsi sistem yaitu berupa proses – proses yang terjadi terhadap sistem dan kebutuhan non fungsi yaitu kebutuhan yang harus dipenuhi untuk perilaku sistem. Kebutuhan perilaku berkaitan dengan kebutuhan operasional, dan kinerja sistem. Untuk mendapatkan kebutuhan tersebut diperlukan teknik pengumpulan kebutuhan dengan studi literatur dan analisis data.

## **C. Pemodelan dan Perancangan**

a. Perancangan Algoritma

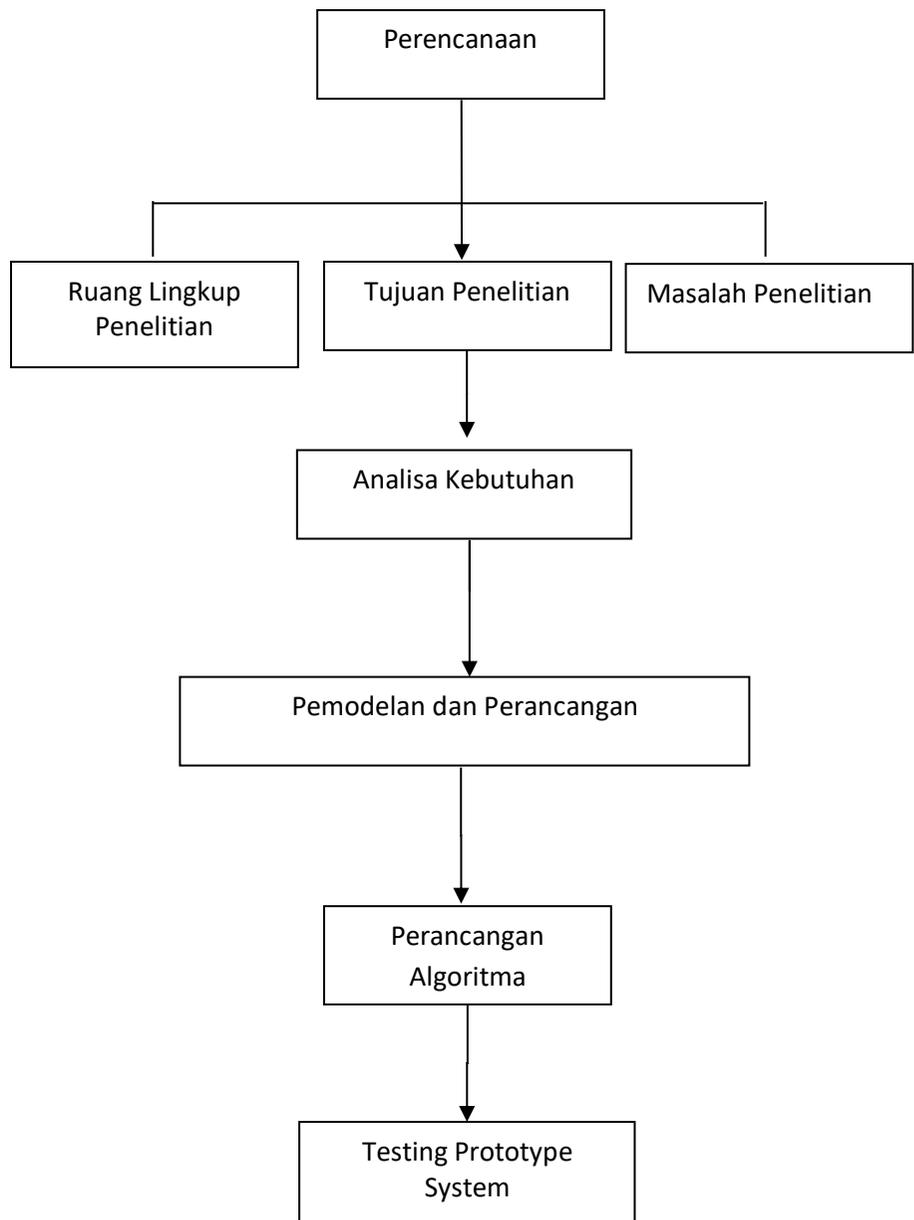
Perancangan yang dilakukan pada penelitian ini yakni melalui penggunaan pemodelan diagram flowchart yang menunjukkan kerja dari algoritma machine learning yang digunakan diantaranya *support vector machine*, *polynomial regression*, dan *bayesian ridge regression*. Di mana ketiga algoritma ini akan mempelajari data yang dimasukkan agar dapat memprediksi besar

pertambahan kasus covid-19 selama 60 hari ke depan dimulai dari 28 Oktober 2020.

#### **D. Hasil Prediksi Algoritma**

Pada tahapan ini akan memaparkan hasil serta membahas prediksi ketiga algoritma *machine learning* yang digunakan yang sudah diterapkan pada bahasa pemrograman python dengan menggunakan jupyter notebook. Memvisualisasikan pertambahan kasus covid-19 di Indonesia dalam bentuk grafik menggunakan python pada jupyter notebook. Hal ini menjadi informasi bagi peneliti dan masyarakat Indonesia agar semakin memiliki kehati-hatian dalam menghadapi pandemi yang sedang terjadi.

Kerangka kerja di atas digambarkan dengan diagram metode penelitian.



*Gambar 3. 1 Diagram Alur Metode Penelitian*

## BAB IV

### HASIL DAN PEMBAHASAN

#### A. Analisis Kebutuhan Sistem dan Informasi

##### 1. Analisis Kebutuhan Data

Pada tahap analisis kebutuhan informasi yang dilakukan adalah menganalisis data yang telah diperoleh dari hasil pengumpulan data melalui laman website resmi yang dipublikasikan oleh Kementerian Kesehatan Republik Indonesia secara *realtime* setiap hari dengan akses <https://www.kemkes.go.id/article/view/20031900002/Dashboard-Data-Kasus-COVID-19-di-Indonesia.html>. Data yang didapatkan melalui situs Kemenkes RI merupakan data penambahan banyak kasus covid-19 di Indonesia dari tiap provinsi yang terdiri dari data kasus terkonfirmasi, data kasus yang sembuh, dan data kasus meninggal mulai dari tanggal 1 September 2020 sampai dengan tanggal 28 Oktober 2020. Data-data ini akan dilakukan prediksi melalui penggunaan 3 algoritma machine learning yakni *support vector machine*, *polynomial regression*, dan *bayesian ridge regression*.

Adapun data yang didapatkan melalui situs Kementerian Kesehatan Republik Indonesia merupakan data penambahan banyak kasus covid-19 di Indonesia dari tiap provinsi dari tanggal 1 September 2020 sampai dengan tanggal 28 Oktober 2020. Sampel dari data tiap provinsi selama 5 hari terakhir dari data yang digunakan dapat dilihat melalui tabel di bawah ini.

*Tabel 4. 1 Sampel Data Kasus Terkonfirmasi COVID-19 dari 34 Provinsi di Indonesia*

<b>Provinsi</b>	<b>Saturday, October 24, 2020</b>	<b>Sunday, October 25, 2020</b>	<b>Monday, October 26, 2020</b>	<b>Tuesday, October 27, 2020</b>	<b>Wednesday, October 28, 2020</b>
DKI Jakarta	100220	100991	101897	102678	103522
Jawa Timur	50653	50921	51217	51506	51752
Jawa Tengah	31586	31942	32098	32098	32732

Sulawesi Selatan	17893	17936	17950	18155	18275
Jawa Barat	33568	33924	34355	34745	35148
Kalimantan Selatan	11629	11648	11650	11662	11732
Sumatera Utara	12472	12562	12655	12745	12837
Bali	11279	11388	11455	11520	11588
Sumatera Selatan	7500	7522	7538	7581	7667
Kalimantan Timur	12944	13119	13255	13348	13562
Sulawesi Utara	5176	5209	5232	5243	5287
Papua	8569	8618	8785	8785	8910
Banten	8563	8794	8914	9038	9208
Nusa Tenggara Barat	3836	3858	3884	3913	3940
Kalimantan Tengah	4233	4251	4274	4284	4288
Sumatera Barat	12432	12791	13060	13373	13649
Gorontalo	2982	3001	3003	3005	3007
Riau	13257	13547	13752	13998	14251
Maluku	3718	3739	3739	3772	3785
Maluku Utara	2188	2188	2192	2193	2196
Aceh	7125	7212	7252	7265	7339

Sulawesi Tenggara	4622	4665	4675	4713	4856
D I Yogyakarta	3506	3550	3576	3617	3662
Kepulauan Riau	3207	3362	3500	3603	3738
Papua Barat	3879	3885	3891	3974	4080
Kalimantan Barat	1564	1564	1577	1606	1620
Lampung	1551	1601	1646	1686	1729
Sulawesi Barat	978	987	988	997	1000
Kalimantan Utara	766	780	782	791	805
Bengkulu	969	995	995	1017	1040
Jambi	1118	1140	1149	1156	1188
Sulawesi Tengah	799	817	821	838	848
Kep. Bangka Belitung	549	553	554	566	573
Nusa Tenggara Timur	649	652	653	667	669

*Tabel 4. 2 Sampel Data Kasus Sembuh COVID-19 dari 34 Provinsi di Indonesia*

<b>Provinsi</b>	<b>Saturday, October 24, 2020</b>	<b>Sunday, October 25, 2020</b>	<b>Monday, October 26, 2020</b>	<b>Tuesday, October 27, 2020</b>	<b>Wednesday, October 28, 2020</b>
-----------------	---	---	---	--	--

DKI Jakarta	85492	86721	87883	89017	90064
Jawa Timur	44665	44907	45203	4545	45683
Jawa Tengah	25968	26331	26634	26951	27278
Sulawesi Selatan	15564	15664	15744	16124	16230
Jawa Barat	22676	23143	23585	2414	24507
Kalimantan Selatan	10373	10425	10481	10513	10593
Sumatera Utara	10126	10201	10286	10376	10464
Bali	10144	10226	10285	10361	10397
Sumatera Selatan	5704	5885	6015	6075	6113
Kalimantan Timur	9682	9832	9959	10175	10267
Sulawesi Utara	4341	4374	4405	4430	4445
Papua	4533	4533	4598	4598	4637
Banten	6455	6574	6835	6936	7030
Nusa Tenggara Barat	3118	3134	3156	3188	3220
Kalimantan Tengah	3684	3710	3744	3776	3793
Sumatera Barat	7248	7420	7605	8085	8283
Gorontalo	2817	2835	2838	2839	2839
Riau	9097	9654	9905	10203	10467

Maluku	2926	3024	3083	3120	3158
Maluku Utara	1928	1928	1928	1928	1928
Aceh	5037	5037	5037	5037	5037
Sulawesi Tenggara	3259	3330	3392	3447	3557
D I Yogyakarta	2842	2910	2939	2971	3013
Kepulauan Riau	2418	2471	2541	2541	2564
Papua Barat	2836	3075	3093	3358	3478
Kalimantan Barat	1288	1288	1300	1306	1328
Lampung	939	941	941	941	977
Sulawesi Barat	763	784	784	816	816
Kalimantan Utara	654	662	664	672	679
Bengkulu	760	768	773	785	788
Jambi	447	501	521	542	574
Sulawesi Tengah	519	558	590	603	613
Kep. Bangka Belitung	473	475	482	495	510
Nusa Tenggara Timur	443	443	443	449	463

*Tabel 4. 3 Sampel Data Kasus Meninggal COVID-19 dari 34 Provinsi di Indonesia*

<b>Provinsi</b>	<b>Saturday, October 24, 2020</b>	<b>Sunday, October 25, 2020</b>	<b>Monday, October 26, 2020</b>	<b>Tuesday, October 27, 2020</b>	<b>Wednesday, October 28, 2020</b>
DKI Jakarta	2146	2164	2177	2188	2204
Jawa Timur	3647	3663	3683	3704	3724
Jawa Tengah	1665	1673	1689	1701	1711
Sulawesi Selatan	447	451	451	451	410
Jawa Barat	668	679	694	708	714
Kalimantan Selatan	468	471	474	477	479
Sumatera Utara	514	516	523	524	526
Bali	371	372	374	378	380
Sumatera Selatan	407	407	409	410	410
Kalimantan Timur	456	459	461	465	466
Sulawesi Utara	195	195	195	197	199
Papua	119	119	124	124	129
Banten	250	258	261	261	262
Nusa Tenggara Barat	217	219	219	219	219
Kalimantan Tengah	148	148	151	151	151

Sumatera Barat	220	226	229	238	242
Gorontalo	81	85	85	85	85
Riau	300	303	307	313	319
Maluku	45	45	45	47	47
Maluku Utara	74	74	74	74	75
Aceh	250	251	256	260	266
Sulawesi Tenggara	77	79	79	80	80
D I Yogyakarta	88	88	88	89	90
Kepulauan Riau	86	87	89	89	90
Papua Barat	61	61	63	63	65
Kalimantan Barat	10	10	12	12	16
Lampung	61	61	64	68	70
Sulawesi Barat	12	12	12	12	14
Kalimantan Utara	10	7	7	7	7
Bengkulu	47	48	48	48	48
Jambi	21	21	21	22	23
Sulawesi Tengah	33	33	33	33	34
Kep. Bangka Belitung	7	7	7	7	7

Nusa Tenggara Timur	7	7	7	7	7
---------------------	---	---	---	---	---

**a. Analisa Algoritma**

Pada tahapan ini akan dianalisa algoritma *machine learning* yang digunakan yakni *support vector machine*, *polynomial regression*, dan *bayesian ridge regression* yang memanfaatkan bahasa pemrograman python dengan jupyter notebook. Adapun hal yang pertama dilakukan yakni penggunaan *library* python pada jupyter notebook untuk dapat menerapkan ketiga algoritma machine learning dalam pembelajaran dan memprediksi data kasus covid-19 di Indonesia. Berikut adalah analisisnya antara lain yaitu :

a) Algoritma *machine learning*

Algoritma *machine learning* atau pembelajaran mesin yang digunakan pada penelitian ini terdapat tiga algoritma antara lain *support vector machine*, *polynomial regression*, dan *bayesian ridge regression* yang digunakan dalam mempelajari data kasus covid-19 dari 34 provinsi di Indonesia agar mendapatkan prediksi angka kasus ketika 60 hari yang akan datang. Data kasus covid-19 yang dikumpulkan mulai dari tanggal 1 September 2020 sampai dengan 28 Oktober 2020. Data kasus yang terdiri dari kasus terkonfirmasi covid-19, data kasus sembuh, dan data kasus meninggal akan divisualisasikan dalam bentuk plot grafik menggunakan *library* dari python. Masing-masing dari data akan di visualisasi dalam bentuk grafik plot bar secara umum melalui data, kemudian grafik prediksi perkembangan covid 19 yang telah dilakukan pembelajaran melalui penggunaan tiga jenis algoritma dengan parameter yang berbeda-beda, dan menghasilkan besar banyak kasus covid-19 pada 60 hari yang akan datang dengan angka yang berbeda-beda pula. Untuk dapat menampilkan data, memvisualisasikan data, menerapkan algoritma, dan melakukan prediksi dengan menggunakan *library* yang disediakan oleh bahasa pemrograman python dengan jupyter notebook seperti pada Gambar 4.1 di bawah ini.

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import matplotlib.colors as mcolors
import pandas as pd
import random
import math
import time
from sklearn.linear_model import LinearRegression, BayesianRidge
from sklearn.model_selection import RandomizedSearchCV, train_test_split, cross_val_score
from sklearn.preprocessing import PolynomialFeatures
from sklearn.tree import DecisionTreeRegressor
from sklearn.svm import SVR
from sklearn.metrics import mean_squared_error, mean_absolute_error, accuracy_score, confusion_matrix
import datetime
import operator
plt.style.use('fivethirtyeight')
%matplotlib inline
```

*Gambar 4. 1 Library Python pada Jupyter Notebook*

Untuk memasukkan data ke dalam bahasa pemrograman dengan memanfaatkan library pandas yang didefinisikan sebagai pd untuk membaca dataset yang tersedia dengan bentuk format csv, di mana data set terdiri dari data kasus baru yang terkonfirmasi, data kasus sembuh, dan data kasus meninggal. Penginputan data pada program dilakukan seperti pada Gambar 4.2

```
In [2]: kasusbarudata = pd.read_csv('konfirmasisept.csv')
sembuhdata = pd.read_csv('sembuhsept.csv')
meninggaldata = pd.read_csv('meninggalsept.csv')
```

*Gambar 4. 2 Input Dataset Kasus Covid-19 di Indonesia*

Ketika data baru dimasukkan terdapat penambahan baris baru yang bernilai kosong, sehingga diperlukan kode untuk menghapus data yang kosong tersebut dengan menggunakan fungsi .dropna seperti pada Gambar 4.3

```
In [3]: kasusbarudata = kasusbarudata.dropna()
sembuhdata = sembahdata.dropna()
meninggaldata = meninggaldata.dropna()
```

*Gambar 4. 3 Membersihkan Nilai Kosong dari Data*

Untuk menetapkan baris pertama sebagai *key* pada data maka dilakukan dengan menggunakan fungsi dari keys sebagaimana pada gambar 4.4

```
In [4]: cols = kasusbarudata.keys()
        cols

out[4]: Index(['Provinsi', 'Tuesday, September 1, 2020',
              'Wednesday, September 2, 2020', 'Thursday, September 3, 2020',
              'Friday, September 4, 2020', 'Saturday, September 5, 2020',
              'Sunday, September 6, 2020', 'Monday, September 7, 2020',
              'Tuesday, September 8, 2020', 'Wednesday, September 9, 2020',
              'Thursday, September 10, 2020', 'Friday, September 11, 2020',
              'Saturday, September 12, 2020', 'Sunday, September 13, 2020',
              'Monday, September 14, 2020', 'Tuesday, September 15, 2020',
              'Wednesday, September 16, 2020', 'Thursday, September 17, 2020',
              'Friday, September 18, 2020', 'Saturday, September 19, 2020',
              'Sunday, September 20, 2020', 'Monday, September 21, 2020',
              'Tuesday, September 22, 2020', 'Wednesday, September 23, 2020',
              'Thursday, September 24, 2020', 'Friday, September 25, 2020',
              'Saturday, September 26, 2020', 'Sunday, September 27, 2020',
              'Monday, September 28, 2020', 'Tuesday, September 29, 2020',
              'Wednesday, September 30, 2020', 'Thursday, October 1, 2020',
              'Friday, October 2, 2020', 'Saturday, October 3, 2020',
              'Sunday, October 4, 2020', 'Monday, October 5, 2020',
              'Tuesday, October 6, 2020', 'Wednesday, October 7, 2020',
              'Thursday, October 8, 2020', 'Friday, October 9, 2020',
              'Saturday, October 10, 2020', 'Sunday, October 11, 2020',
              'Monday, October 12, 2020', 'Tuesday, October 13, 2020',
              'Wednesday, October 14, 2020', 'Thursday, October 15, 2020',
              'Friday, October 16, 2020', 'Saturday, October 17, 2020',
              'Sunday, October 18, 2020', 'Monday, October 19, 2020',
              'Tuesday, October 20, 2020', 'Wednesday, October 21, 2020',
              'Thursday, October 22, 2020', 'Friday, October 23, 2020',
              'Saturday, October 24, 2020', 'Sunday, October 25, 2020',
              'Monday, October 26, 2020', 'Tuesday, October 27, 2020',
              'Wednesday, October 28, 2020'],
             dtype='object')
```

*Gambar 4. 4 Menetapkan Baris Key pada Data*

Agar dapat melakukan perhitungan maupun pelatihan terhadap isi dari data maka diperlukan pemisahan dari objek yang dimiliki data terhadap isi data tersebut untuk dapat dikenali secara jelas oleh algoritma. Dalam memisahkan isi data tersebut dengan menggunakan kode seperti gambar 4.5

```
In [5]: cases = kasusbarudata.loc[:, cols[1]:cols[-1]]
        recoveries = sembuhdata.loc[:, cols[1]:cols[-1]]
        deaths = meninggaldata.loc[:, cols[1]:cols[-1]]
```

*Gambar 4. 5 Memisahkan Isi Data dengan Objek pada Data*

Dalam melakukan perhitungan terhadap banyak kasus covid-19 di Indonesia melalui data 34 provinsi yang dimiliki, maka terlebih dahulu dilakukan pendefinisian terhadap variabel yang akan dihitung.

```
In [7]: dates = kasusbaru.keys()

new_cases = []
total_deaths = []
mortality_rate = []
recovery_rate = []
total_recovered = []
total_active = []

jkt_cases = []
jatim_cases = []
jateng_cases = []
susel_cases = []
jabar_cases = []
kalsel_cases = []
sumut_cases = []
bali_cases = []
sumsel_cases = []
kaltim_cases = []
sulut_cases = []
papua_cases = []
banten_cases = []
ntb_cases = []
kalteng_cases = []
```

*Gambar 4. 6 Mendefinisikan Variabel Perhitungan*

Setelah variabel didefinisikan, lalu dibuat suatu fungsi yang dapat menghitung isi dari data yang diperlukan seperti banyaknya kasus baru, banyak kasus sembuh, banyak kasus meninggal, banyak kasus pasien yang masih dalam perawatan, rasio tingkat kesembuhan, dan rasio tingkat kematian dari pasien covid-19 di Indonesia.

```
for i in dates:
    kasusbaru_sum = kasusbaru[i].sum()
    sembuh_sum = sembuh[i].sum()
    meninggal_sum = meninggal[i].sum()

    # kasusbaru, sembuh, meninggal, dan dalam perawatan
    new_cases.append(kasusbaru_sum)
    total_deaths.append(meninggal_sum)
    total_recovered.append(sembuh_sum)
    total_active.append(kasusbaru_sum-sembuh_sum-meninggal_sum)

    # hitung rate
    mortality_rate.append(meninggal_sum/kasusbaru_sum)
    recovery_rate.append(sembuh_sum/kasusbaru_sum)
```

*Gambar 4. 7 Fungsi Menghitung Rate Kasus*

Untuk memvisualisasikan data ke dalam bentuk plot, dilakukan penghitungan terhadap kasus harian yang mengalami peningkatan dari seluruh provinsi di Indonesia dengan fungsi seperti pada gambar 4.8.

```
In [8]: def daily_increase(data):
        d = []
        for i in range(len(data)):
            if i == 0:
                d.append(data[0])
            else:
                d.append(data[i]-data[i-1])
        return d

indo_daily_increase = daily_increase(new_cases)
jkt_daily_increase = daily_increase(jkt_cases)
jatim_daily_increase = daily_increase(jatim_cases)
jateng_daily_increase = daily_increase(jateng_cases)
sulsel_daily_increase = daily_increase(sulsel_cases)
jabar_daily_increase = daily_increase(jabar_cases)
kalsel_daily_increase = daily_increase(kalsel_cases)
sumut_daily_increase = daily_increase(sumut_cases)
bali_daily_increase = daily_increase(bali_cases)
sumsel_daily_increase = daily_increase(sumsel_cases)
kaltim_daily_increase = daily_increase(kaltim_cases)
sulut_daily_increase = daily_increase(sulut_cases)
papua_daily_increase = daily_increase(papua_cases)
banten_daily_increase = daily_increase(banten_cases)
ntb_daily_increase = daily_increase(ntb_cases)
kalteng_daily_increase = daily_increase(kalteng_cases)
sumbar_daily_increase = daily_increase(sumbar_cases)
grtl_daily_increase = daily_increase(grtl_cases)
riau_daily_increase = daily_increase(riau_cases)
```

*Gambar 4. 8 Fungsi Menghitung Peningkatan Kasus Harian*

Agar dapat melakukan peramalan atau prediksi terhadap data maupun kasus covid-19 di Indonesia selama 60 hari kedepan sejak tanggal 28 Oktober 2020 sampai dengan 27 Desember 2020. Dengan demikian diperlukan pembentukan array yang baru dalam proses prediksi dengan menggunakan fungsi reshape.

```
In [9]: days_since_1_sept = np.array([i for i in range(len(dates))]).reshape(-1, 1)
new_cases = np.array(new_cases).reshape(-1, 1)
total_deaths = np.array(total_deaths).reshape(-1, 1)
total_recovered = np.array(total_recovered).reshape(-1, 1)
```

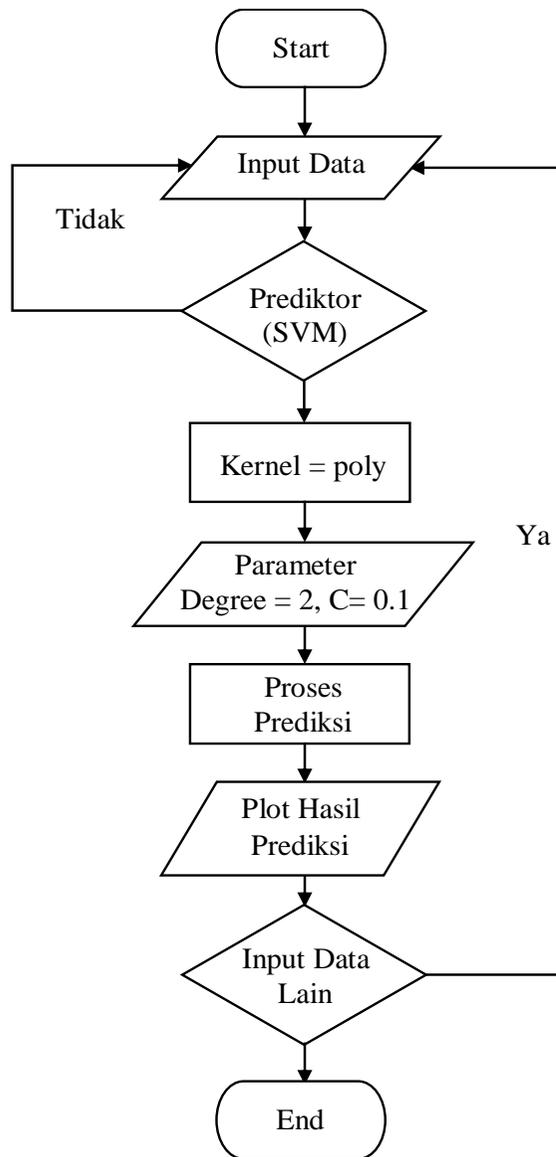
*Gambar 4. 9 Membentuk Array yang Baru*

```
In [69]: days_in_future = 60
future_forecast = np.array([i for i in range(len(dates)+days_in_future)]).reshape(-1, 1)
adjusted_dates = future_forecast[:-60]
```

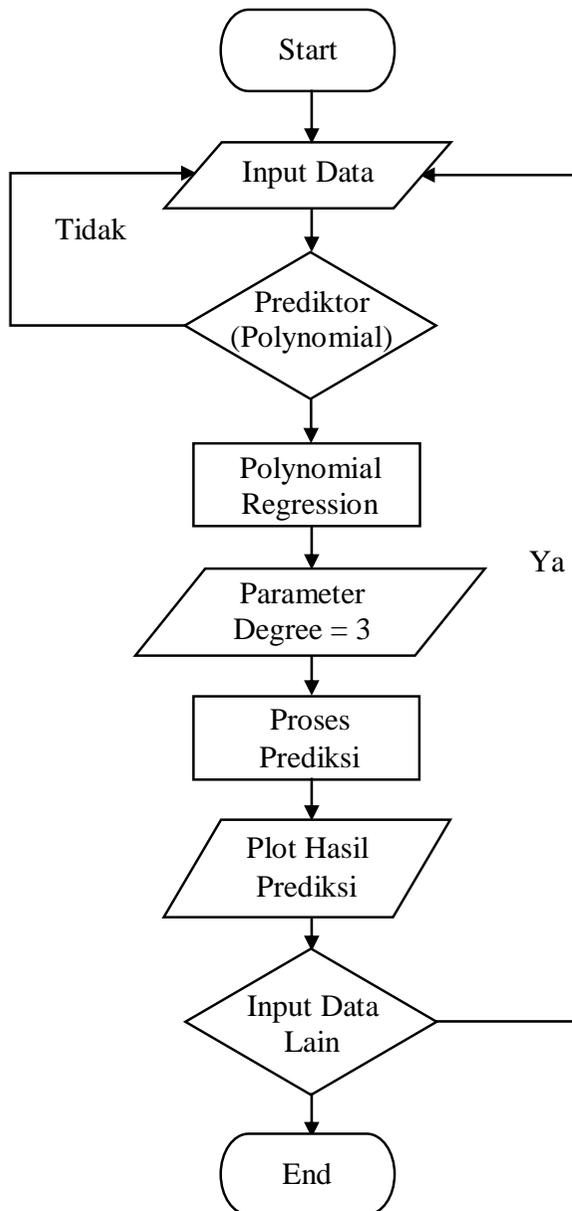
*Gambar 4. 10 Membentuk Array yang Baru untuk Prediksi*

#### b) *Flowchart* Algoritma

*Flowchart* dari algoritma berfungsi untuk menunjukkan tahapan-tahapan dari algoritma dalam mempelajari suatu data dengan didasari oleh parameter yang berbeda-beda bentar sajikan dalam bentuk diagram, sebagaimana pada gambar 4.11 merupakan *flowchart* dari algoritma *support vector machine*



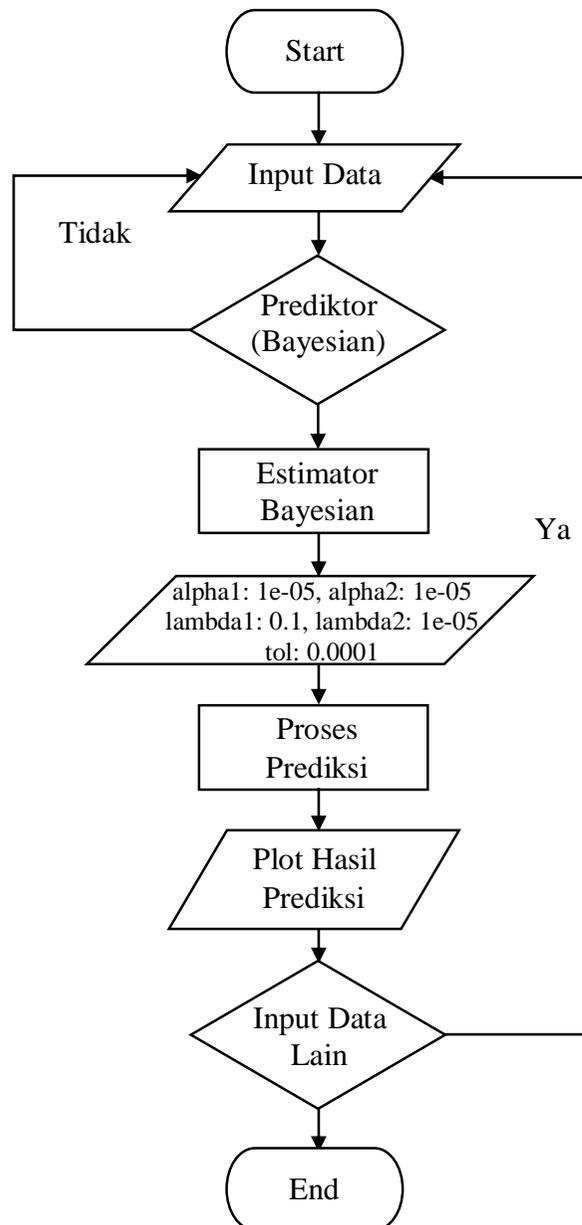
Gambar 4. 11 Flowchart SVM



*Gambar 4. 12 Flowchart Polynomial*

*Flowchart* dari algoritma berfungsi untuk menunjukkan tahapan-tahapan dari algoritma dalam mempelajari suatu data dengan didasari oleh parameter yang berbeda-beda bentar sajikan dalam

bentuk diagram, sebagaimana pada gambar 4.12 merupakan *flowchart* dari algoritma *polynomial regression*.



Gambar 4. 13 Flowchart Bayesian

*Flowchart* dari algoritma berfungsi untuk menunjukkan tahapan-tahapan dari algoritma dalam mempelajari suatu data dengan didasari oleh parameter yang berbeda-beda bentar sajian dalam bentuk diagram, sebagaimana pada gambar 4.13 merupakan *flowchart* dari algoritma *bayesian ridge regression*.

## 2. Testing dan Implementasi

### a) Kebutuhan perangkat lunak

Adapun kebutuhan perangkat lunak dalam menerapkan algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* dalam memprediksi banyak kasus covid-19 di Indonesia menggunakan bahasa pemrograman Python dengan *Jupyter Notebook* diantaranya adalah:

- 1) Bahasa Pemrograman Python 3
- 2) Jupyter Notebook
- 3) Anaconda Navigator

### b) Kebutuhan perangkat keras

Untuk kebutuhan akan perangkat keras untuk menerapkan algoritma *Support Vector Machine*, *Polynomial Regression*, dan *Bayesian Ridge Regression* dalam memprediksi banyak kasus covid-19 di Indonesia menggunakan bahasa pemrograman Python dengan *Jupyter Notebook* diantaranya kebutuhan akan

- 1) PC dengan spesifikasi:
  - a) Processor Core i3
  - b) RAM 4GB
  - c) HDD 500GB
  - d) Perangkat *networking*

### c) Implementasi

#### 1) Implementasi Visualisasi Data COVID-19 di Indonesia

Visualisasi dari suatu data berguna untuk lebih memahami serta mengetahui isi maupun makna dari data dalam bentuk plot yang disajikan. Untuk mengimplementasikan visualisasi plot kasus covid-19 di Indonesia, dengan menggunakan kode seperti pada gambar 4.14

```
In [144]: plt.figure(figsize=(20,8))
plt.plot(adjusted_dates, new_cases)
plt.title('Jumlah Kasus Covid-19 di Indonesia', size=30)
plt.xlabel('Tanggal/Bulan (2020)', size =15)
plt.ylabel('Banyak Kasus', size=20)
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
```

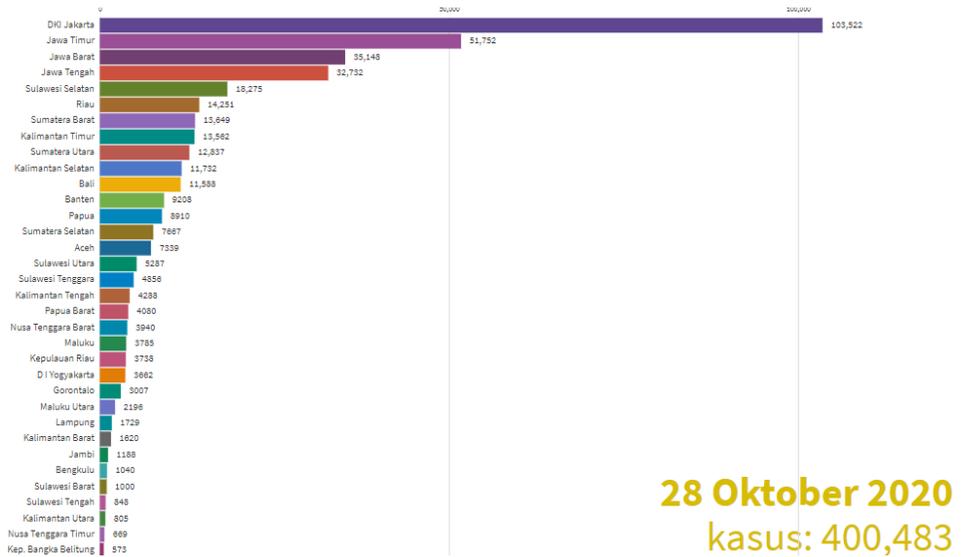
Gambar 4. 14 Implementasi Visualisasi Plot Kasus Covid-19

Setelah kode program pada python diimplementasikan, akan menghasilkan tampilan plot perkembangan dari kasus covid-19 di Indonesia mulai sejak tanggal 1 September 2020 sampai dengan 28 Oktober 2020,



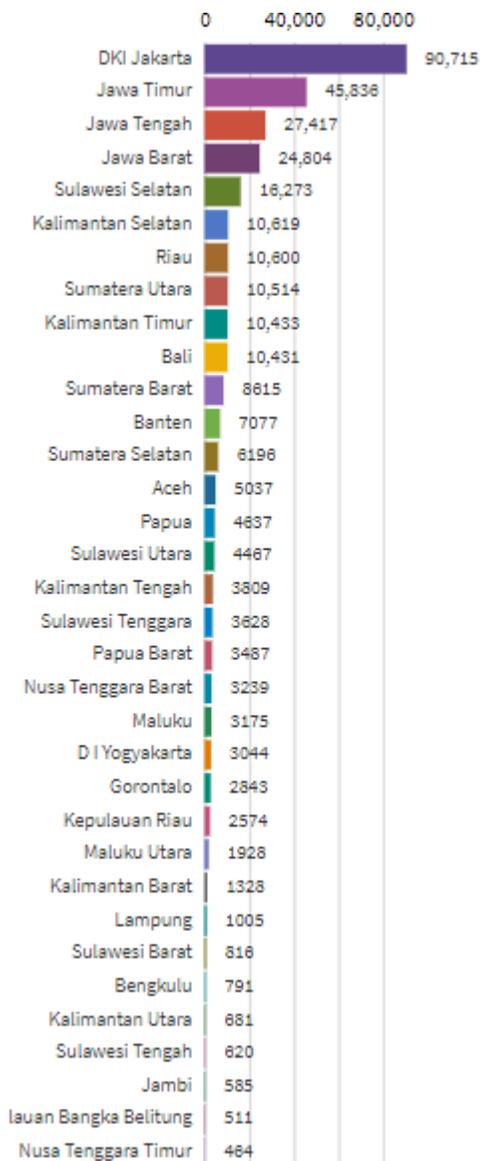
Gambar 4. 15 Visualisasi Kasus Covid-19

Pada gambar 4.16 memperlihatkan visualisasi dari banyaknya kasus yang sudah terkonfirmasi pada tiap-tiap provinsi di Indonesia yakni sebanyak 34 provinsi sampai dengan tanggal 28 Oktober 2020 yaitu sebesar 400483 kasus.

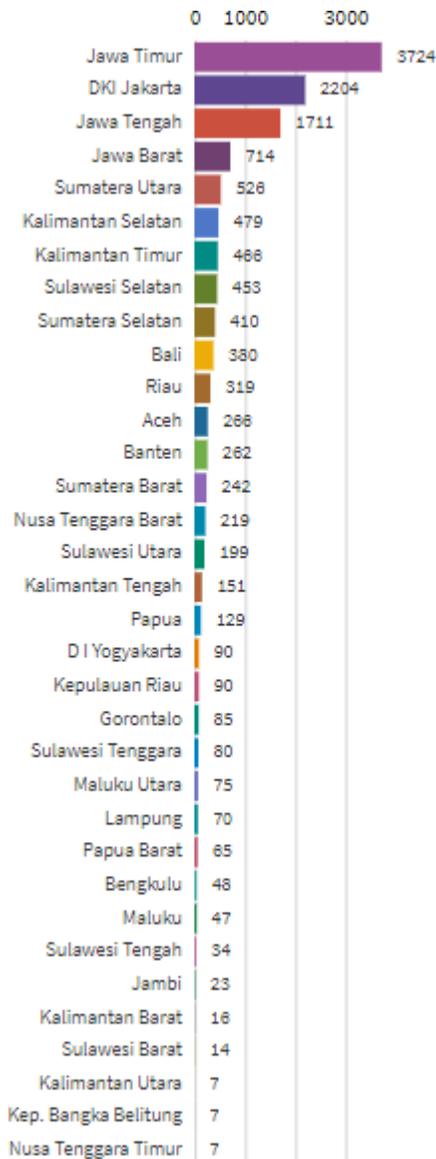


Gambar 4. 16 Visualisasi Kasus Terkonfirmasi 34 Provinsi

Pada gambar 4.17 memperlihatkan visualisasi dari banyaknya kasus yang sembuh pada tiap-tiap provinsi di Indonesia yakni sebanyak 34 provinsi sampai dengan tanggal 28 Oktober 2020.



Gambar 4. 17 Visualisasi Kasus Sembuh 34 Provinsi



*Gambar 4. 18 Visualisasi Kasus Meninggal 34 Provinsi*

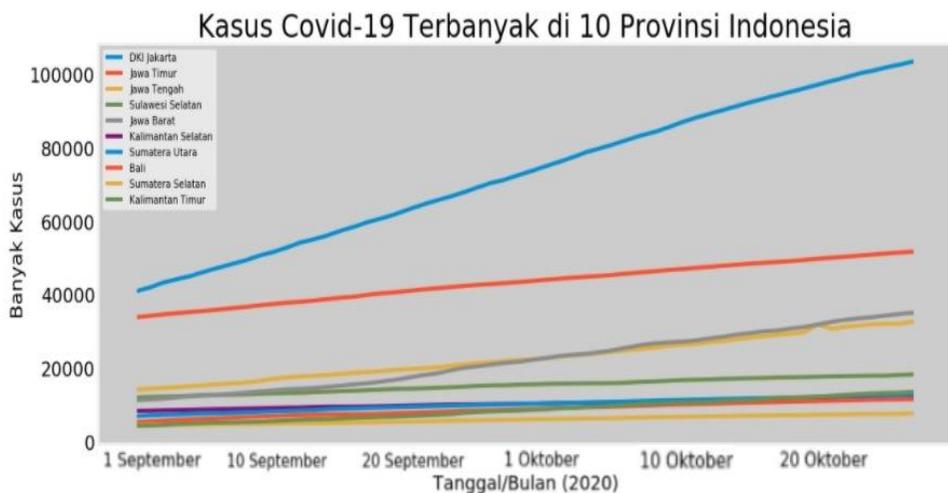
Pada gambar 4.18 memperlihatkan visualisasi dari banyaknya kasus yang meninggal pada tiap-tiap provinsi di Indonesia yakni sebanyak 34 provinsi sampai dengan tanggal 28 Oktober 2020.

## 2) Implementasi Visualisasi Data Covid-19 pada 10 Provinsi Terbanyak

Agar dapat melihat secara lebih jelas 10 provinsi di Indonesia yang paling banyak menyumbang angka kasus covid-19 yaitu dengan memvisualisasikannya menggunakan kode seperti pada gambar 4.19

```
In [155]: plt.figure(figsize=(16, 7))
plt.plot(adjusted_dates, jkt_cases)
plt.plot(adjusted_dates, jatim_cases)
plt.plot(adjusted_dates, jateng_cases)
plt.plot(adjusted_dates, susel_cases)
plt.plot(adjusted_dates, jabar_cases)
plt.plot(adjusted_dates, kalsel_cases)
plt.plot(adjusted_dates, sumut_cases)
plt.plot(adjusted_dates, bali_cases)
plt.plot(adjusted_dates, sumsel_cases)
plt.plot(adjusted_dates, kaltim_cases)
ax = plt.gca()
ax.set_facecolor('0.80')
plt.title('Kasus Covid-19 Terbanyak di 10 Provinsi Indonesia', size=30)
plt.xlabel('Tanggal/Bulan (2020)', size=20)
plt.ylabel('Banyak Kasus', size=20)
plt.legend(['DKI Jakarta', 'Jawa Timur', 'Jawa Tengah', 'Sulawesi Selatan',
           'Jawa Barat', 'Kalimantan Selatan', 'Sumatera Utara', 'Bali',
           'Sumatera Selatan', 'Kalimantan Timur'], size=20)
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

Gambar 4. 19 Implementasi Visualisasi Plot 10 Provinsi Terbanyak



Gambar 4. 20 Visualisasi Kasus Covid-19 pada 10 Provinsi

Melalui gambar 4.20 dapat diketahui jika provinsi yang memiliki angka kasus covid-19 paling tinggi jauh diantara yang lainnya adalah DKI Jakarta, kemudian disusul oleh Jawa Timur, dan provinsi lainnya yang memiliki angka kasus covid-19 yang hampir mirip besarnya.

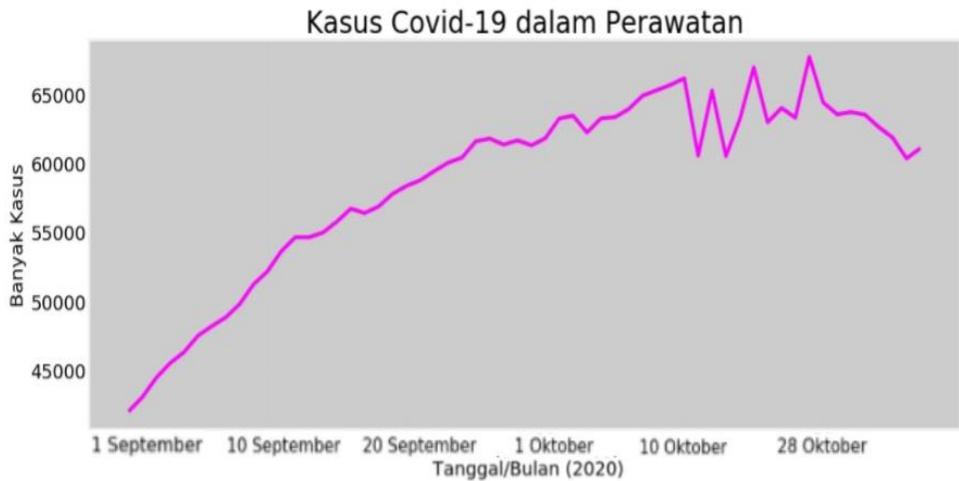
### 3) Implementasi Visualisasi Data Covid-19 dalam Perawatan

Untuk mengetahui visualisasi dari para pasien covid-19 yang masih berada dalam perawatan sampai dengan tanggal 28 Oktober 2020 yakni dengan menerapkan kode seperti pada gambar 4.21

```
In [62]: plt.figure(figsize=(16, 7))
plt.plot(adjusted_dates, total_active, color='magenta')
ax = plt.gca()
ax.set_facecolor('0.80')
plt.title('Kasus Covid-19 dalam Perawatan', size=30)
plt.xlabel('Tanggal/Bulan (2020)', size=20)
plt.ylabel('Banyak Kasus', size=20)
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

*Gambar 4. 21 Implementasi Visualisasi Covid-19 dalam Perawatan*

Hasil visualisasi dari kasus covid-19 oleh pasien yang masih berada dalam perawatan menunjukkan bahwa sampai tanggal 28 Oktober sebanyak kurang lebih 60.000 pasien yang masih dirawat di pusat perawatan pada tiap-tiap daerah yang menangani covid-19.

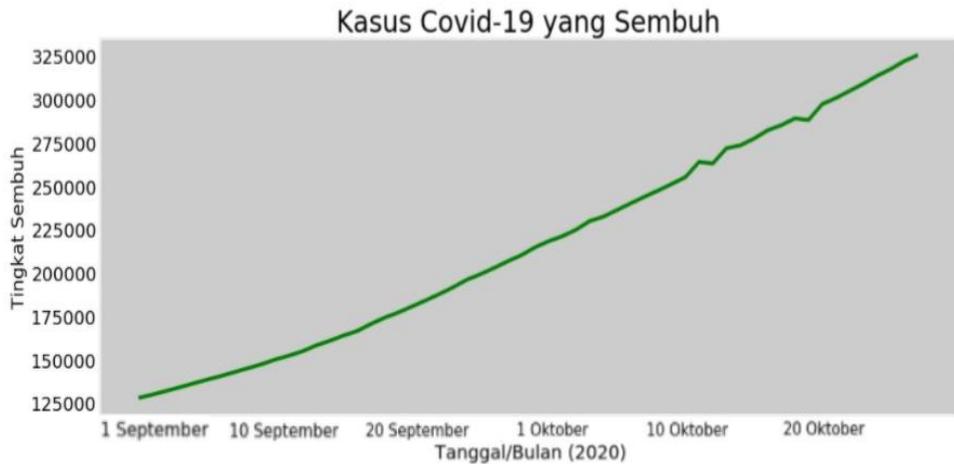


*Gambar 4. 22 Visualisasi Kasus Covid-19 dalam Perawatan*

- 4) Implementasi Visualisasi Data Covid-19 yang Sembuh  
 Untuk mengetahui visualisasi dari para pasien covid-19 yang sudah sembuh sampai dengan tanggal 28 Oktober 2020 yakni dengan menerapkan kode seperti pada gambar 4.23

```
In [63]: plt.figure(figsize=(16, 7))
plt.plot(adjusted_dates, total_recovered, color='green')
ax = plt.gca()
ax.set_facecolor('0.80')
plt.title('Kasus Covid-19 yang Sembuh', size=30)
plt.xlabel('Tanggal/Bulan (2020)', size=20)
plt.ylabel('Tingkat Sembuh', size=20)
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

*Gambar 4. 23 Implementasi Visualisasi Covid-19 yang Sembuh*



*Gambar 4. 24 Visualisasi Kasus Covid-19 yang Sembuh*

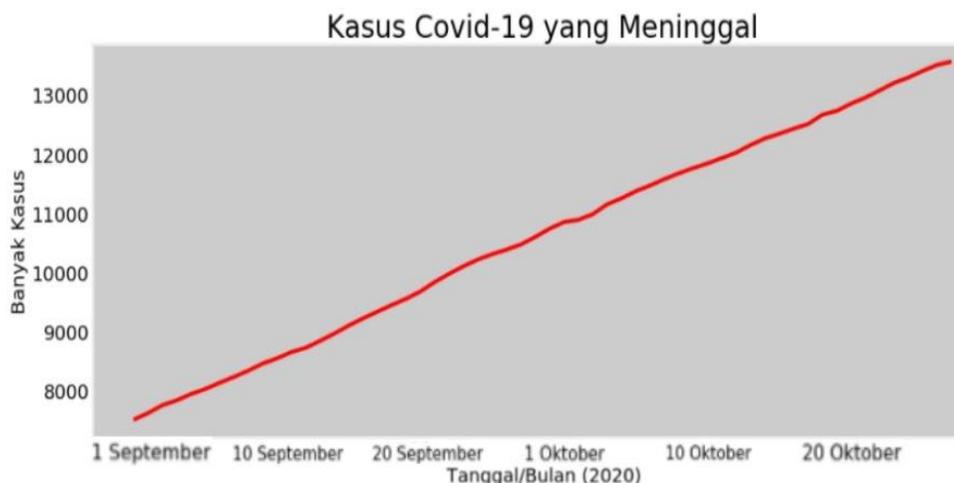
Hasil visualisasi dari kasus covid-19 oleh pasien yang sudah sembuh menunjukkan bahwa sampai tanggal 28 Oktober sebanyak kurang lebih 325000 pasien.

#### 5) Implementasi Visualisasi Data Covid-19 yang Meninggal

Untuk mengetahui visualisasi dari para pasien covid-19 yang meninggal sampai dengan tanggal 28 Oktober 2020 yakni dengan menerapkan kode seperti pada gambar 4.25

```
In [65]: plt.figure(figsize=(16, 7))
plt.plot(adjusted_dates, total_deaths, color='red')
ax = plt.gca()
ax.set_facecolor('0.80')
plt.title('Kasus Covid-19 yang Meninggal', size=30)
plt.xlabel('Tanggal/Bulan (2020)', size=20)
plt.ylabel('Banyak Kasus', size=20)
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

*Gambar 4. 25 Implementasi Visualisasi Covid-19 yang Meninggal*



Gambar 4. 26 Visualisasi Kasus Covid-19 yang Meninggal

Hasil visualisasi dari kasus covid-19 oleh pasien yang meninggal menunjukkan bahwa sampai tanggal 28 Oktober sebanyak lebih dari 13000 pasien.

#### 6) Implementasi Visualisasi Data Tingkat Kesembuhan

Untuk mengetahui visualisasi dari tingkat kesembuhan yang dimiliki pasien covid-19 di Indonesia sampai dengan tanggal 28 Oktober 2020 yakni dengan menerapkan kode seperti pada gambar 4.27

```
In [67]: mean_recovery_rate = np.mean(recovery_rate)
plt.figure(figsize=(16, 7))
plt.plot(adjusted_dates, recovery_rate, color='blue')
ax = plt.gca()
ax.set_facecolor('0.80')
plt.axhline(y = mean_recovery_rate, linestyle='--', color='black')
plt.title('Tingkat Sembuh dari Covid-19 di Indonesia', size=30)
plt.legend(['tingkat sembuh', 'y='+str(mean_recovery_rate)], prop={'size': 20})
plt.xlabel('Tanggal/Bulan (2020)', size=20)
plt.ylabel('Tingkat Kesembuhan', size=20)
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

Gambar 4. 27 Implementasi Visualisasi Tingkat Kesembuhan



*Gambar 4. 28 Visualisasi Tingkat Kesembuhan*

Hasil visualisasi dari tingkat kesembuhan kasus covid-19 para pasien di Indonesia menunjukkan bahwa sampai tanggal 28 Oktober 2020 memiliki tingkat kesembuhan sebesar 75%.

#### 7) Implementasi Visualisasi Data Tingkat Kematian

Untuk mengetahui visualisasi dari tingkat kematian yang dimiliki pasien covid-19 di Indonesia sampai dengan tanggal 28 Oktober 2020 yakni dengan menerapkan kode seperti pada gambar 4.29

```
In [66]: mean_mortality_rate = np.mean(mortality_rate)
plt.figure(figsize=(16, 7))
plt.plot(adjusted_dates, mortality_rate, color='orange')
ax = plt.gca()
ax.set_facecolor('0.80')
plt.axhline(y = mean_mortality_rate, linestyle='--', color='black')
plt.title('Tingkat Kasus Kematian di Indonesia', size=30)
plt.legend(['Tingkat Kematian', 'y='+str(mean_mortality_rate)], prop={'size': 20})
plt.xlabel('Tanggal/Bulan (2020)', size=20)
plt.ylabel('Tingkat Rasio Kematian', size=20)
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

*Gambar 4. 29 Implementasi Visualisasi Tingkat Kematian*



Gambar 4. 30 Visualisasi Tingkat Kesembuhan

Hasil visualisasi dari tingkat kematian kasus covid-19 para pasien di Indonesia menunjukkan bahwa sampai tanggal 28 Oktober 2020 memiliki tingkat kematian sebesar 3,7%.

- 8) Implementasi Prediksi Kasus Covid-19 di Indonesia menggunakan algoritma *support vector machine* sampai 27 Desember 2020

```
In [136]: svm_confirmed = SVR(shrinking=True, kernel='poly', degree = 2, C=0.1)
```

```
In [137]: svm_confirmed.fit(X_train_confirmed, y_train_confirmed)
svm_confirmed.score(X_test_confirmed, y_test_confirmed)
```

Gambar 4. 31 Parameter SVM

Pada penggunaan algoritma support vector machine untuk memprediksi banyak kasus covid-19 di Indonesia selama 60 hari kedepan mulai dari 28 Oktober 2020, dengan menggunakan kernel polinomial,  $d = 2$ , serta  $C = 0.1$  yang diimplementasikan seperti pada gambar 4.31

```
In [138]: svm_pred = svm_confirmed.predict(future_forcast)
```

Gambar 4. 32 Proses Prediksi SVM

Untuk memproses prediksi menggunakan algoritma *support vector machine* dengan fungsi *future forecast* yang telah didefinisikan sebelumnya dan diterapkan pada kode seperti pada gambar 4.32

Langkah selanjutnya adalah mengevaluasi model yang telah terbentuk. Salah satu metode yang banyak digunakan untuk penilaian model statistik adalah *Root Mean Square Error* (RMSE). RMSE mengukur kinerja dari model regresi. Dengan mengukur akar rata-rata kesalahan kuadrat dan dihitung.

Nilai RMSE yang lebih rendah menunjukkan bahwa prediksi mendekati nilai aktual, sehingga memiliki akurasi prediktif yang lebih baik.

Setelah dievaluasi dengan menggunakan fungsi RMSE, maka didapatkan nilai RMSE dari algoritma support vector machine yang diterapkan terhadap data yakni sebesar 4913.96

```
In [143]: svm_test_pred = svm_confirmed.predict(X_test_confirmed)
plt.plot(svm_test_pred)
plt.plot(y_test_confirmed)
plt.legend(['Test Data', 'SVM Predictions'])
print('RMSE:', np.sqrt(mean_squared_error(svm_test_pred, y_test_confirmed)))
```

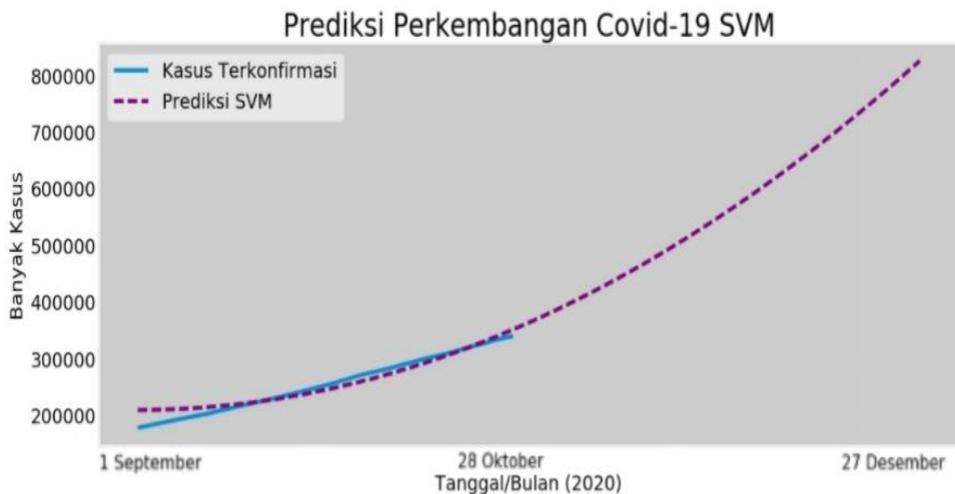
*Gambar 4. 33 Evaluasi SVM*

Untuk mengetahui visualisasi dari hasil prediksi SVM terhadap banyak kasus covid-19 di Indonesia sampai dengan tanggal 27 Desember 2020 yakni dengan menerapkan kode seperti pada gambar 4.34

```
In [156]: plt.figure(figsize=(16, 7))
plt.plot(adjusted_dates, new_cases)
plt.plot(future_forcast, svm_pred, linestyle='dashed', color='purple')
ax = plt.gca()
ax.set_facecolor('0.80')
plt.title('Prediksi Perkembangan Covid-19 SVM', size=30)
plt.xlabel('Tanggal/Bulan (2020)', size=20)
plt.ylabel('Banyak Kasus', size=20)
plt.legend(['Kasus Terkonfirmasi', 'Prediksi SVM'], prop={'size': 20})
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

Gambar 4. 34 Implementasi Visualisasi Prediksi SVM

Hasil visualisasi dari prediksi SVM terhadap banyak kasus covid-19 para pasien di Indonesia menunjukkan bahwa sampai tanggal 27 Desember 2020 memiliki banyak kasus mencapai 800000 kasus.



Gambar 4. 35 Hasil Prediksi Kasus Covid-19 SVM

- 9) Implementasi Prediksi Kasus Covid-19 di Indonesia menggunakan algoritma *polynomial regression* sampai 27 Desember 2020

```
In [145]: # mengubah data menjadi polynomial
poly = PolynomialFeatures(degree=3)
poly_X_train_confirmed = poly.fit_transform(X_train_confirmed)
poly_X_test_confirmed = poly.fit_transform(X_test_confirmed)
poly_future_forecast = poly.fit_transform(future_forecast)
```

Gambar 4. 36 Parameter dan Transform Polynomial

Pada penggunaan algoritma *polynomial regression* untuk memprediksi banyak kasus covid-19 di Indonesia selama 60 hari kedepan mulai dari 28 Oktober 2020, dengan menggunakan  $d = 3$ , yang diimplementasikan seperti pada gambar 4.36

```
In [146]: # polynomial regression
linear_model = LinearRegression(normalize=True, fit_intercept=False)
linear_model.fit(poly_X_train_confirmed, y_train_confirmed)
test_linear_pred = linear_model.predict(poly_X_test_confirmed)
linear_pred = linear_model.predict(poly_future_forecast)
print('RMSE:', np.sqrt(mean_squared_error(test_linear_pred, y_test_confirmed)))
```

Gambar 4. 37 Evaluasi Polynomial

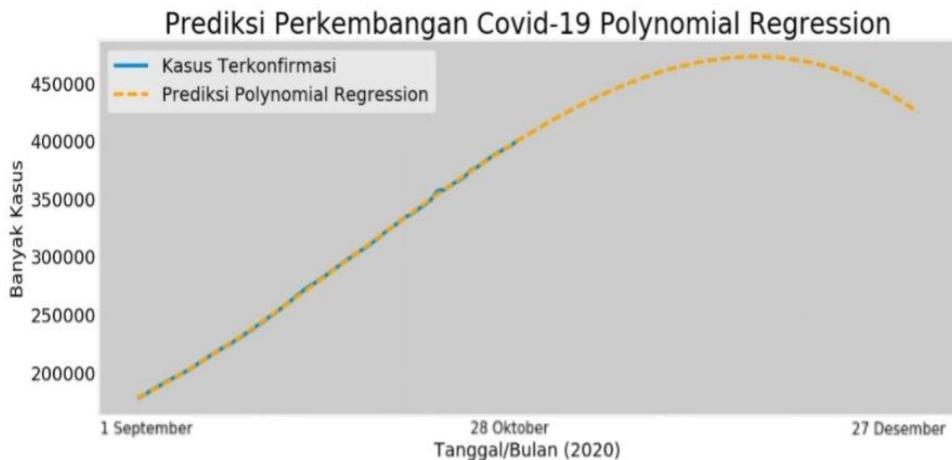
Setelah dievaluasi dengan menggunakan fungsi RMSE, maka didapatkan nilai RMSE dari algoritma polynomial regression yang diterapkan terhadap data yakni sebesar 560.37

```
In [157]: plt.figure(figsize=(16, 7))
plt.plot(adjusted_dates, new_cases)
plt.plot(future_forecast, linear_pred, linestyle='dashed', color='orange')
ax = plt.gca()
ax.set_facecolor('0.80')
plt.title('Prediksi Perkembangan Covid-19 Polynomial Regression', size=30)
plt.xlabel('Tanggal/Bulan (2020)', size=20)
plt.ylabel('Banyak Kasus', size=20)
plt.legend(['Kasus Terkonfirmasi', 'Prediksi Polynomial Regression'], prop={'size': 20})
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

Gambar 4. 38 Implementasi Visualisasi Prediksi Polynomial

Untuk mengetahui visualisasi dari hasil prediksi algoritma *polynomial regression* terhadap banyak kasus covid-19 di Indonesia sampai dengan tanggal 27 Desember 2020 yakni dengan menerapkan kode seperti pada gambar 4.38

Hasil visualisasi dari prediksi algoritma *polynomial regression* terhadap banyak kasus covid-19 para pasien di Indonesia menunjukkan bahwa sampai tanggal 27 Desember 2020 memiliki banyak kasus menuju puncak pada pertengahan Desember sebanyak 450000 kasus kemudian mengalami penurunan banyak kasus sampai pada tanggal 27 Desember 2020.



Gambar 4. 39 Hasil Prediksi Kasus Covid-19 Polynomial

- 10) Implementasi Prediksi Kasus Covid-19 di Indonesia menggunakan algoritma *bayesian ridge regression* sampai 27 Desember 2020

```
In [149]: # bayesian ridge polynomial regression
tol = [1e-3, 1e-2, 1e-4]
alpha_1 = [1e-5, 1e-4, 1e-2, 1e-1]
alpha_2 = [1e-5, 1e-4, 1e-2, 1e-1]
lambda_1 = [1e-4, 1e-3, 1e-2, 1e-1]
lambda_2 = [1e-5, 1e-4, 1e-2, 1e-1]

bayesian_grid = {'tol': tol, 'alpha_1': alpha_1, 'alpha_2': alpha_2, 'lambda_1': lambda_1, 'lambda_2': lambda_2}

bayesian = BayesianRidge(fit_intercept=False, normalize=True)
bayesian_search = RandomizedSearchCV(bayesian, bayesian_grid, scoring='neg_mean_squared_error', cv=2, return_train_score=True)
bayesian_search.fit(poly_X_train_confirmed, y_train_confirmed)
```

Gambar 4. 40 Mencari Parameter Bayesian

```
In [150]: bayesian_search.best_params_

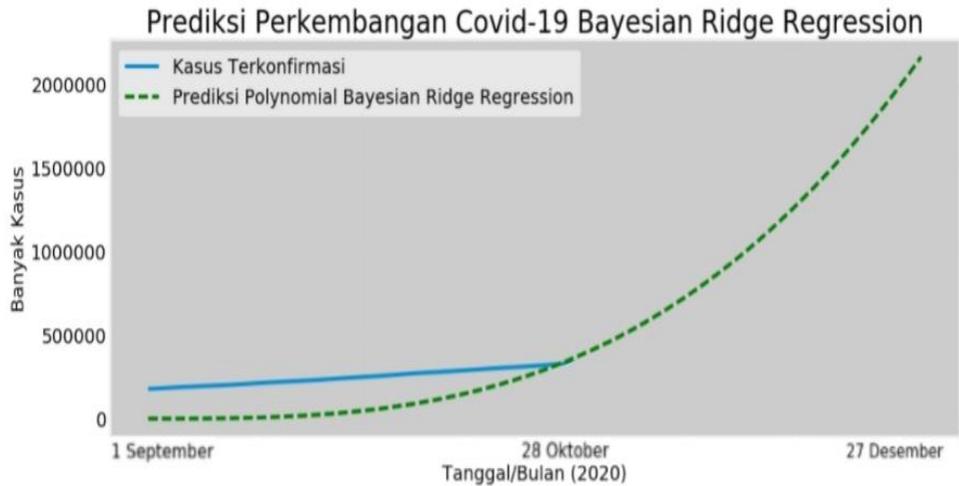
Out[150]: {'alpha_1': 0.0001,
           'alpha_2': 0.01,
           'lambda_1': 0.1,
           'lambda_2': 1e-05,
           'tol': 0.01}
```

Gambar 4. 41 Parameter Bayesian

Pada penggunaan algoritma *bayesian ridge regression* untuk memprediksi banyak kasus covid-19 di Indonesia selama 60 hari kedepan mulai dari 28 Oktober 2020, dengan menggunakan kernel  $\alpha_1 = 0.0001$ ,  $\alpha_2 = 0.01$ ,  $\lambda_1 = 0.1$ ,  $\lambda_2 = 0.00001$ , serta  $\text{tol} = 0.1$  yang diimplementasikan seperti pada gambar 4.41

```
In [151]: bayesian_confirmed = bayesian_search.best_estimator_
test_bayesian_pred = bayesian_confirmed.predict(poly_X_test_confirmed)
bayesian_pred = bayesian_confirmed.predict(poly_future_forecast)
print('RMSE:', np.sqrt(mean_squared_error(test_bayesian_pred, y_test_confirmed)))
```

Gambar 4. 42 Evaluasi Bayesian



Gambar 4. 43 Hasil Prediksi Kasus Covid-19 Bayesian

Setelah dievaluasi dengan menggunakan fungsi RMSE, maka didapatkan nilai RMSE dari algoritma *bayesian ridge regression* yang diterapkan terhadap data yakni sebesar 44912.27

Hasil visualisasi dari prediksi algoritma *bayesian ridge regression* terhadap banyak kasus covid-19 para pasien di Indonesia menunjukkan bahwa sampai tanggal 27 Desember 2020 memiliki banyak kasus mencapai 2000000 kasus.

## BAB V

### PENUTUP

#### A. Kesimpulan

Berdasarkan penelitian dan pembahasan yang ketua dan tim penelitian lakukan, dapat ditarik kesimpulan sebagai berikut :

1. Algoritma *machine learning* telah banyak digunakan dalam melakukan prediksi terhadap sesuatu yang dibutuhkan sebagai suatu informasi melalui data yang telah tersedia sebagaimana untuk memprediksi kasus covid-19 di Indonesia menggunakan data selama 2 bulan yakni mulai dari 1 September - 28 Oktober 2020 dengan memanfaatkan algoritma *support vector machine*, *polynomial regression*, dan *bayesian ridge regression*. Hasil dari ketiga algoritma ini melakukan prediksi terhadap banyak kasus dengan jumlah yang berbeda-beda dikarenakan perbedaan parameter yang digunakan dalam mempelajari suatu data
2. Sesudah algoritma diterapkan untuk mempelajari data didapatkan hasil prediksi dari masing-masing algoritma terhadap banyak kasus covid-19 di Indonesia. Berdasarkan nilai evaluasi menggunakan RMSE bahwa algoritma *polynomial regression* memiliki nilai RMSE paling kecil yang berarti mempunyai keakuratan paling baik diantara ketiga algoritma yang digunakan. Algoritma *polynomial regression* memprediksi bahwa pada pertengahan Desember 2020 angka kasus covid-19 akan berada di puncak sebesar lebih dari 450.000 kasus, kemudian mengalami penurunan sampai pada prediksi di tanggal 27 Desember 2020
3. Sistem pembelajaran mesin ini dibangun dengan menggunakan bahasa pemrograman python 3 dengan jupyter notebook yang dilengkapi berbagai *library* dengan kelengkapan untuk mengimplementasikan algoritma *machine learning* serta memvisualisasikan data yang dimiliki maupun data yang telah dimodelkan dengan algoritma.

#### B. Penutup

Dengan selesainya penelitian ini dilaksanakan hingga tahap akhir dan laporan dihasilkan, maka:

1. Prediksi kasus covid-19 di Indonesia ini menjadi suatu gambaran maupun informasi terhadap masyarakat Indonesia serta pemerintahan agar lebih memperhatikan upaya dalam menekan angka kasus covid-19 agar mengalami penurunan.
2. Pada model mesin pembelajaran ini memerlukan banyak algoritma lainnya yang lebih baik dalam prediksi agar menemukan akurasi yang maksimal untuk mempelajari data
3. Data pelatihan yang digunakan dapat ditambahkan dengan jangka waktu yang lebih lama agar algoritma ataupun mesin dapat melakukan pembelajaran secara lebih dalam terhadap data sehingga didapatkan hasil prediksi yang lebih baik

## DAFTAR REFERENSI

- [1] D. Handayani, D. R. Hadi, F. Isbaniah, E. Burhan, and H. Agustin, "Penyakit Virus Corona 2019," *J. Respirologi Indones.*, vol. 40, no. 2, pp. 119–129, 2020, [Online]. Available: <https://jurnalrespirologi.org/index.php/jri/article/view/101>.
- [2] C. Ceraolo and F. M. Giorgi, "Genomic variance of the 2019-nCoV coronavirus," *J. Med. Virol.*, vol. 92, no. 5, pp. 522–528, 2020, doi: 10.1002/jmv.25700.
- [3] Peng *et al.*, "A pneumonia outbreak associated with a new coronavirus of probable bat origin," *Nature*, vol. 579, no. 7798, pp. 270–273, 2020, [Online]. Available: <https://www.nature.com/articles/s41586-020-2012-7?rel=outbound>.
- [4] N. Zhu *et al.*, "A Novel Coronavirus from Patients with Pneumonia in China, 2019," *N. Engl. J. Med.*, vol. 382, no. 8, pp. 727–733, 2020, doi: 10.1056/nejmoa2001017.
- [5] WHO, *Clinical management of severe acute respiratory infection when novel coronavirus (2019-nCoV) infection is suspected*. Geneva: WHO Press, 2020.
- [6] Direktorat Jenderal P2P Kemenkes RI, "Pedoman Kesiapsiagaan Menghadapi Infeksi Novel Coronavirus (2019-nCoV)." <https://covid19.kemkes.go.id/downloads/#.Xtva> (accessed Nov. 04, 2020).
- [7] J. Hurwitz and D. Kirsch, *Machine learning*. Hoboken, 2016.
- [8] O. Pentakalos, *Introduction to machine learning*. New York: The Press Syndicate of The Univeristy of Cambridge, 2019.
- [9] M. Singh, S. Sharma, and A. Kaur, "Performance Analysis of Decision Trees," *Int. J. Comput. Appl.*, vol. 71, no. 19, pp. 975–8887, 2013, [Online]. Available: <https://www.ijcaonline.org/archives/volume71/number19/12593-9232>.
- [10] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
- [11] M. Furqan, "Face Recognition Using Eigenfaces and Smooth

Support Vector Machine,” Universiti Malaysia Pahang, 2011.

- [12] J. Weston and A. Ben-Hur, “A User’s Guide to Support Vector Machine,” *Data Min. Tech. Life Sci.*, vol. 609, pp. 223–39, 2010, doi: 10.1007/978-1-60327-241-4\_13.
- [13] T. Evgeniou and M. Pontil, “Support vector machines: Theory and applications,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2049 LNAI, pp. 249–257, 2001, doi: 10.1007/3-540-44673-7\_12.
- [14] E. Prasetyo, *Data Mining Konsep dan Aplikasi Menggunakan Matlab*. Yogyakarta: Andi, 2012.
- [15] Y. W. Chang, C. J. Hsieh, K. W. Chang, M. Ringgaard, and C. J. Lin, “Training and testing low-degree polynomial data mappings via linear SVM,” *J. Mach. Learn. Res.*, vol. 11, pp. 1471–1490, 2010, [Online]. Available: <https://www.jmlr.org/papers/volume11/chang10a/chang10a.pdf>.
- [16] J. S. Malensang, H. Komalig, and D. Hatidja, “Pengembangan Model Regresi Polinomial Berganda Pada Kasus Data Pemasaran,” *J. Ilm. Sains*, vol. 12, no. 2, p. 149, 2013, doi: 10.35799/jis.12.2.2012.740.
- [17] S. D. Permai and H. Tanty, “Linear regression model using bayesian approach for energy performance of residential building,” *Procedia Comput. Sci.*, vol. 135, pp. 671–677, 2018, doi: 10.1016/j.procs.2018.08.219.
- [18] R. Sianipar and H. Wadi, *Pemrograman Python (Teori Dan Implementasi)*. Yogyakarta: Informatika, 2015.
- [19] B. D. Prasetya, F. S. Pamungkas, and I. Kharisudin, “Pemodelan dan Peramalan Data Saham dengan Analisis Time Series menggunakan Python,” *Prism. Pros. Semin. Nas. Mat.*, vol. 3, pp. 714–718, 2020, [Online]. Available: <https://journal.unnes.ac.id/sju/index.php/prisma/> ISSN.
- [20] D. Setiabudidaya, “Penggunaan Piranti Lunak Jupyter Notebook Dalam Upaya Mensosialisasikan Open Science,” 2018, doi: 10.31227/osf.io/2h7q4.
- [21] H. Shen, “Interactive notebooks: Sharing the code,” *Nature*, vol. 515,

no. 7525, pp. 151–152, 2014, doi: 10.1038/515151a.

- [22] P. Fernando and E. G. Brian, “IPython: A System for Interactive Scientific Computing,” *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 21–29, 2007, [Online]. Available: <https://ieeexplore.ieee.org/document/4160251>.
- [23] M. Furqan, R. Kurniawan, and K. HP, “Evaluasi Performa Support Vector Machine Classifier Terhadap Penyakit Mental,” *JSINBIS*, vol. 10, no. 2, 2020.
- [24] M. Hoffmann, H. Kleine-Weber, N. Krüger, M. Müller, C. Drosten, and S. Pöhlmann, “The novel coronavirus 2019 (2019-nCoV) uses the SARS-coronavirus receptor ACE2 and the cellular protease TMPRSS2 for entry into target cells,” 2020, doi: 10.1101/2020.01.31.929042.
- [25] L. E. Gralinski and V. D. Menachery, “Return of the coronavirus: 2019-nCoV,” *Viruses*, vol. 12, no. 2, 2020, doi: 10.3390/v12020135.
- [26] J. Hartono, *Analisis dan Desain Sistem Informasi*. Yogyakarta: Andi Offset, 2007