ANALISIS KESUKSESAN FILM DENGAN DATA MINING

Teuku Hashrul–2016730067

1 Data Skripsi

Pembimbing utama/tunggal: **Kristopher David Harjono**
Pembimbing pendamping: -
Kode Topik : **KDH4701**
Topik ini sudah dikerjakan selama : **1** semester
Pengambilan pertama kali topik ini pada : Semester **7 - Ganjil 19/20**
Pengambilan pertama kali topik ini di kuliah : **Skripsi 1**
Tipe Laporan : **B -** Dokumen untuk reviewer pada presentasi dan **review Skripsi 1**

2 Latar Belakang

Film merupakan media komunikasi yang bersifat audio visual untuk menyampaikan suatu pesan kepada penontonnya. Keberadaan film membuat masyarakat menjadikan film sebagai media hiburan. Beragam cara dapat dilakukan untuk menikmati sebuah film, yaitu datang ke bioskop, membeli kaset DVD dan *streaming* menggunakan aplikasi *desktop* dan *smartphone*.

Film yang dibuat ada karena ada kumpulan orang dibalik layar yang bekerja untuk membuatnya. Terdapat beragam perusahaan produksi film yang berlomba-lomba untuk membuat film yang dapat memperoleh keuntungan maksimum. Dengan menciptakan film yang sesuai dengan keinginan penontonnya,maka peluang keuntungan yang diperoleh pun akan semakin meningkat dan dapat menutup*budget* yang digunakan sebelumnya untuk biaya produksi.

Berdasarkan penelitian analisis data film yang ada sebelumnya pada *kaggle*, data film memiliki beberapa atribut umum yaitu judul,*genre*, *rating*, keuntungan, *budget*, nilai *review* dari situs internet, aktor yang terlibat dan lama tayang. Data film yang ada digunakan untuk membantu menganalisis sifat dari film. Penelitian yang dilakukan adalah analisis prediksi nilai IMDB *score*, yaitu situs yang berisi data film.

Genre adalah sebutan untuk membedakan berbagai jenis film. Sebuah film memiliki satu atau beragam *genre*. Terdapat banyak film yang menggunakan kombinasi dari beberapa *genre*. *Genre* yang ada berupa *action, adventure, animation ,drama ,comedy ,horror ,romance* dan lain-lain.

Terdapat banyak kemungkinan faktor yang dapat dijadikan sebuah film dapat memperoleh keuntungan maksimum. Faktor kesuksesan film berupa faktor *rating* dari situs *review* film, nama aktor yang terlibat, nama sutradara yang terlibat dan jumlah *budget* yang dikeluarkan. Salah satu faktor dan kombinasi beberapa faktor dapat memengaruhi kesuksesan film.

Berdasarkan uraian diatas,akan dilakukan sebuah penelitian mengenai data film. Penelitian ini adalah analisis kesuksesan film dengan *data mining* untuk memperoleh faktor-faktor yang ada dapat memprediksi kesuksesan sebuah film. Dari faktor yang diperoleh, maka akan diprediksi *revenue*/pendapatan sebuah film berdasarkan data film yang sudah ada sebelumnya.

Pada penelitian ini dibuat sekumpulan perangkat lunak yang digunakan untuk mengumpulkan, membersihkan data, analisis ,pembuatan model, evaluasi kerja model dan visualisasi data.Perangkat lunak yang dibuat akan membantu menganalisis data film yang digunakan. Pembuatan perangkat lunak akan menggunakan bahasa pemrograman *Python* dan memanfaatkan beberapa *library* dari *Python.Pandas* digunakan untuk integrasi data. *Sci-kit learn* digunakan untuk implementasi regresi, teknik *clustering* dan *classification*

untuk memprediksi keuntungan sebuah film. Penelitian ini akan melakukan eksperimen untuk membandingkan beberapa metode *machine learning* dalam memprediksi kesuksesan sebuah film.

3 Rumusan Masalah

Berkaitan dengan identifikasi masalah yang ada pada deskripsi diatas, masalah-masalah yang ada dapat dirumuskan sebagai berikut.

- Apa saja faktor yang dapat digunakan untuk menentukan kesuksesan sebuah film ?
- Bagaimana langkah dalam melakukan analisis kesuksesan film dengan *data mining* ?
- Bagaimana hasil pengujian pada penelitian ini ?

4 Tujuan

Tujuan yang ingin dicapai dari penelitian ini adalah:

- Mengeksplorasi data yang dikumpulkan
- Membuat perangkat lunak yang dapat menggunakan metode *data mining* untuk melakukan analisis faktor-faktor yang dapat berpengaruh pada kesuksesan film
- Menguji metode-metode yang digunakan pada penelitian ini

5 Detail Perkembangan Pengerjaan Skripsi

Detail bagian pekerjaan skripsi sesuai dengan rencana kerja/laporan perkembangan terakhir :

1. **Melakukan studi literatur dengan mencari jurnal, *paper* mengenai penelitian sejenis dari berbagai sumber untuk membantu penulis dalam menulis.**
   Status : Ada sejak rencana kerja skripsi.
   Hasil :

   Sebagai referensi utama dalam penelitian, penulis mempelajari *data mining* dari artikel situs *kaggle* yang berjudul *"Analyze IMDB score with data mining algorithms"*. Artikel ini membahas penerapan *data mining* pada data film untuk dibandingkan 3 metode *classification* dan performanya. Bahasa pemrograman yang digunakan untuk eksperimen adalah R.

   Di dalam artikel tersebut, terdapat langkah-langkah *data mining* yang diterapkan. Penelitian dimulai dengan *Data Exploration* untuk memahami data. *Data cleaning* untuk membersihkan data agar dapat digunakan saat *Data mining*. *Data visualization* untuk memvisualisasikan data sehingga mempermudah dalam melihat korelasi dan hubungan tiap data. *Data preprocessing* untuk mengubah data sesuai kebutuhan. *Implement algorithm* untuk memasukkan data yang sudah diproses ke algoritma yang relevan yaitu *classification*. *Classification* merupakan salah satu algoritma *supervised machine learning* yang digunakan untuk memrediksi nilai sebuah kelas.

   Dataset yang digunakan pada artikel *kaggle* memiliki 5043 data film dengan 28 variabel. Data film berasal dari 100 negara. *Dataset* memiliki 2399 nama sutradara. Nama-nama variabel beserta deskripsinya adalah sebagai berikut :

   - **movie _title** : judul film
   - **duration** : durasi film

- **director _name**: nama sutradara
- **director _facebook _likes** : jumlah *likes* pada *facebook page* sutradara
- **actor _1 _name** : nama pemeran utama
- **actor _1 _facebook _likes** : jumlah *likes* pada *facebook page* artis
- **actor _2 _name** : nama pemeran pendukung pertama
- **actor _2 _facebook _likes** : jumlah *likes* pada *facebook page* artis
- **actor _3 _name** : nama pemeran pendukung kedua
- **actor _3 _facebook _likes** : jumlah *likes* pada *facebook page* artis
- **num_user_for_reviews** : jumlah *user* yang memberikan *review*
- **num_critic_for_reviews** : jumlah *user* yang memberikan *critical review*
- **num_voted_user** : jumlah *user* yang mendukung film
- **cast_total_facebook_likes** : jumlah *like* dari setiap pemeran film di *facebook*
- **movie_facebook_likes** : jumlah *likes* film di *facebook*
- **plot_keywords** : kata kunci yang mendeskripsikan film
- **facenumber_in_poster** : jumlah wajah pemain di poster film
- **color** : jenis warna film ('Black and white' atau 'Color')
- **genres** : kategori film
- **title_year** : tahun rilis film (dari tahun 1916 sampai 2016)
- **language** : bahasa film
- **country** : negara asal film
- **content_rating** : *nilai* konten dalam film
- **aspect_ratio** : perbandingan panjang dan lebar layar resolusi film
- **movie_imdb_link** : tautan film pada imdb
- **gross** : pendapatan kotor
- **budget** : biaya produksi film
- **imdb_score** : nilai film yang diberikan oleh imdb

Pada tahap eksplorasi data, beberapa hal dilakukan untuk melakukan *data cleaning* seperti menghilangkan film yang memiliki *missing value*, menghilangkan film yang duplikat, melakukan normalisasi pada judul film. Tahap eksplorasi dilakukan untuk memahami data dan mengubah data ke bentuk yang lebih relevan untuk tahap pengujian model

Pada tahap visualisasi data, terdapat beberapa teknik yang dilakukan untuk membantu menganalisis data menggunakan grafik seperti memunculkan histogram jumlah film setiap tahun dari 1916 sampai 2016. Pengurutan data dilakukan untuk memunculkan informasi-informasi yang dibutuhkan seperti 20 besar film berdasarkan keuntungan dan 20 besar nama sutradara yang menghasilkan film dengan nilai IMDB tertinggi

Eksperimen yang dilakukan diartikel akan memproses *dataset* yang sudah dibersihkan untuk dimasukkan ke beberapa algoritma *classification* yaitu *Decision Tree*, *K-nearest neighbors* dan *Random Forest*. Berdasarkan hasil pengujian ternyata *Random Forest* memiliki akurasi yang paling tinggi dibanding algoritma lain yaitu 0,76 atau 76 persen. Sehingga, *model* yang dibuat dapat dipercaya untuk memprediksi seberapa bagus film berdasarkan skor IMDB.

2. **Melakukan studi literatur mengenai ilmu statistika dasar khususnya dalam memahami persebaran data**
   Status : ditambah semenjak skripsi 1
   Hasil :

5.1 Measuring the Central Tendency

Central Tendency adalah cara untuk mengukur persebaran tiap nilai pada kumpulan data. Kumpulan data yang besar sulit untuk dibaca sehingga membutuhkan suatu cara untuk memahaminya. *Central Tendency* dapat menghitung sebuah nilai yang dapat merepresentasikan kumpulan data. Terdapat beberapa cara untuk mengukur persebaran data yaitu menggunakan *mean* / rata-rata, median (Q2) dan modus.

5.1.1 Mean

Mean / rata-rata adalah bilangan yang mewakili sekumpulan data. Rata-rata dapat dihitung dengan menjumlahkan setiap data dibagi dengan jumlah elemen pada data tersebut. Sebuah kumpulan data X dengan elemen x1,x2....xn memiliki N elemen. Berikut adalah rumus *mean* yaitu :

$$\bar{x} = \frac{\sum_i^N x_i}{N} = \frac{x1 + x2 + ... + x_N}{N} \tag{1}$$

X pada persamaan 1 merupakan kumpulan data. x1 sampai xn merepresentasikan tiap nilai dari kumpulan data. Penjumlahan dari setiap elemen dapat dibagi dengan jumlah elemen untuk mendapatkan rata-rata dari data tersebut. Untuk mempermudah penjelasan, diberikan contoh yaitu sebuah kumpulan data nilai ujian suatu kelas yaitu 90,85,75 dan 80. Jumlah elemen pada data nilai yaitu 4. Menggunakan *mean* pada 1, didapatkan :

$$\bar{x} = \frac{90 + 85 + 75 + 80}{4} = \frac{330}{4} = 82.5$$

5.1.2 Median

Median adalah nilai tengah yang dapatkan dari sebuah data yang terurut. Jika banyaknya data genap, maka rumus mediannnya adalah :

$$Me(Q2) = \frac{(X_{n/2} + X_{(n/2)+1})}{2} \tag{2}$$

$X_n dan X_{n+1}$ pada persamaan 3 merupakan data urutan ke $n/2$ yang sudah terurut menaik. Jika banyaknya data ganjil, maka rumus mediannya adalah :

$$Me(Q2) = \frac{X_{(n+1)/2}}{2} \tag{3}$$

$X_{(n+1)/2}$ pada persamaan 3 merupakan data urutan ke $n/2$ yang sudah terurut menaik. Untuk mempermudah penjelasan, diberikan sebuah contoh yaitu kumpulan data nilai ujian yang sudah terurut menaik yaitu 75,80,85 dan 90. Maka didapatkan perhitungan *median* yaitu :

$$Me(Q2) = \frac{X_{n/2} + X_{n/2} + 1}{2} = \frac{X_2 + X_3}{2} = \frac{80 + 85}{2} = 82.5$$

5.1.3 Modus

Modus adalah elemen pada kumpulan data yang paling sering muncul. Terdapat beberapa jenis *modus* yaitu *unimodal* , *bimodal* dan *trimodal*. *Modus unimodal* yaitu elemen dengan frekuensi terbanyak berjumlah 1. Berikut adalah contoh kumpulan data nilai ujian sekolah yang terurut adalah 75,80,85,85 dan 90. *Modus* pada kumpulan data ini adalah 85 karena frekuensi kemunculan elemen 85 adalah 2 kali.

3. **Melakukan studi literatur mengenai proses *data mining***
   Status : Ada sejak rencana kerja skripsi.
   Hasil :

5.2 Data Mining

Data Mining / *Knowledge Discovery Process* (KDD) adalah proses menemukan suatu pola dari kumpulan data yang besar. Dengan *data mining*, manusia dapat menemukan sebuah informasi / pemahaman baru dari data. Sumber data objek yang dapat diproses untuk *data mining* adalah dari *database*, *data warehouse* atau data yang didapatkan dari sebuah proses.

Suatu data objek yang digunakan di *data mining* merupakan sebuah entitas. Kumpulan dari data objek disebut *data set*. Contoh nama lain *data set* adalah *data points* dan *objects*. Contoh data objek adalah data pelanggan di salon, transaksi pembelian di supermarket, data pasien di rumah sakit, data mahasiswa di kampus dan lain-lain.

Atribut adalah sebuah karakteristik dan kondisi dari data objek. Data objek dapat memiliki satu atau lebih atribut. Atribut dapat disebut juga sebagai dimensi, fitur atau variabel. Contoh atribut adalah sebuah data objek transaksi pembelian di supermarket memiliki kumpulan atribut yaitu *customer_ID*, *name* dan *address*. Sebuah atribut memiliki beberapa jenis yaitu nominal, biner, ordinal, atau numerik.

Atribut numerik adalah jenis nilai kuantitatif. Atribut numerik dapat diukur kuantitasnya. Nilai dari atribut numerik dapat berupa bilangan bulat (*integer*) atau bilangan *real*. Atribut numerik dapat dibagi lagi menjadi interval dan rasio.

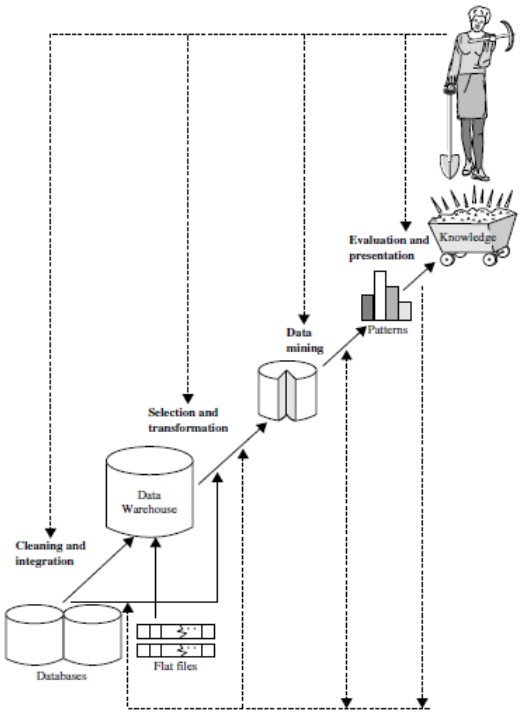
Atribut nominal adalah tipe atribut yang tiap nilainya merupakan sebuah kategori/kondisi/kode. Atribut nominal memiliki isi berupa sebuah nama, simbol dari objek yang direpresentasikan. Nilai nominal tidak dapat dipengaruhi oleh perhitungan karena setiap angka/kode menunjukkan kondisi. Contoh atribut nominal adalah sebuah objek data pelanggan supermarket memiliki atribut jenis kelamin yang bernilai 'Pria' atau 'Wanita'.

Atribut biner adalah tipe atribut yang serupa dengan nominal tetapi hanya memiliki 2 jenis nilai. Biasanya, nilai biner dikodekan menjadi 0 dan 1. Contoh dari nilai biner adalah atribut pembayaran pada data objek transaksi memiliki nilai *true* atau *false* yang menyatakan pernyataan pembayaran sudah diselesaikan.

Atribut ordinal adalah tipe atribut yang memiliki hubungan keterurutan dari setiap nilainya. Atribut ordinal dapat dihasilkan dari mengubah nilai numerik yang disebut *discretization*. Contoh dari atribut ordinal adalah nilai survei kepuasan penduduk yang awalnya memiliki rentang nilai 0 sampai 10 lalu dikonversikan menjadi ordinal berupa buruk, puas dan sangat puas.

Proses dari *data mining* yaitu

- ***Data cleaning*** : menghilangkan noise dan data yang tidak konsisten
- ***Data integration*** : menggabungkan data dari beberapa sumber jika ada
- ***Data transformation*** :mengubah bentuk data menjadi lebih mudah dan relevan untuk kebutuhan analisis
- ***Data selection*** : memilih data yang relevan untuk melakukan analisis
- ***Data mining*** :proses menggunakan metode *machine learning* untuk menemukan pola dari sebuah data
- ***Pattern evaluation*** : untuk memeriksa dari pola yang dihasilkan apakah dapat menghasilkan kebenaran mengenai pola yang ditemukan



Gambar 1: proses data mining

5.2.1 *Data Cleaning*

Data Cleaning merupakan salah satu tahap *data Preprocessing*. *Data cleaning* adalah kegiatan untuk membersihkan data. Data kotor adalah data dengan yang memiliki *missing value*, mengubah atau menghilangkan *noisy data.Noisy data* adalah data yang seharusnya tidak berada dikumpulan *dataset*. *Noisy data* dapat muncul dikarenakan beberapa hal seperti kesalahan proses saat perangkat keras membaca data, salah input,*human error* atau kesalahan saat diproses menggunakan perangkat lunak. *Noisy data* harus dihilangkan dikarenakan dapat memengaruhi hasil proses *data mining*.

Proses *Data Cleaning* membantu analisis data yang lebih valid. Terdapat beberapa cara yang dapat dilakukan jika pada data terdapat nilai yang hilang / *missing value* seperti :

- **Mengabaikan bagian data tersebut:** metode ini dapat dengan cara mengabaikan *missing value*. *Missing value* dapat diabaikan jika data yang *missing value* tidak terlalu banyak. Sedikit data yang diabaikan tidak akan mempengaruhi hasil *data mining*

- **Mengisi nilai yang hilang secara manual:** metode ini tidak dapat direkomendasikan karena prosesnya yang memakan banyak waktu.
- **Menggunakan nilai konstan untuk mengisi nilai yang hilang:** *missing value* dapat diubah dengan membuat nilai konstan. Contoh yang dapat dipahami adalah mengubah *value* yang kosong menjadi *'unknown'* pada suatu atribut. Data yang sudah ditandai *'unknown'* akan membantu komputer mengidentifikasinya.
- **Menggunakan nilai tengah:** nilai yang hilang dapat diubah dengan nilai tengah yang tidak akan mempengaruhi distribusi data. Cara yang dapat digunakan adalah mengubah data tersebut menjadi *mean* /rata-rata dari *data set*.

5.2.2 *Data Integration*

Data integration adalah proses menggabungkan data dari beberapa sumber data. *Data integration* dapat menghasilkan data baru yang akan membantu proses *data mining*. Kebutuhan baru akan tambahan deskripsi data membuat proses *data integration* perlu dilakukan. Cara yang dapat dilakukan pada tahap *Data integration* adalah menggabungkan beberapa tabel dari basisdata, *web crawling* yaitu mengekstrak data dari *web pages*.

5.2.3 *Data Reduction*

Data reduction adalah salah satu *data preprocessing* yang dilakukan untuk mengurangi *data set* menjadi jumlah yang lebih sedikit. *Data reduction* diterapkan untuk mempercepat proses *data mining* dan meningkatkan akurasi dalam proses *data mining*. Metode *Data reduction* dibagi menjadi dua yaitu *dimensionality reduction* dan *numerosity reduction*.

Dimensionality Reduction adalah proses memilih atribut tertentu dan menghilangkan atribut data yang tidak dibutuhkan. Teknik yang dapat dilakukan adalah menggunakan *feature selection*. *Feature selection* adalah teknik untuk memilih atribut yang relevan untuk *predictive analysis* menggunakan *machine learning*.

Numerosity Reduction adalah proses memilih baris data tertentu dan menghilangkan sisa baris yang tidak digunakan. Cara yang dapat dilakukan adalah *sampling* yaitu mengambil *subset* dari data secara *random*. *Sampling* akan mengambil sebagian data yang mewakili kumpulan dari *dataset*.

5.2.4 *Data Transformation*

Data Transformation adalah proses mengubah data yang lebih sesuai saat digunakan saat proses *data mining*. Terdapat beberapa metode yang dapat digunakan untuk *data transformation* yaitu :

- **Smoothing** : *smoothing* adalah kegiatan untuk memproses *noisy data*. *Noisy data* dapat diproses dengan berbagai cara seperti mengubah menjadi *mean* data tersebut dan mengabaikannya.
- **Attribute Construction** : menambah atribut baru berdasarkan atribut yang sudah ada demi membantu proses *data mining*.
- **Aggregation**: menghitung dan menyimpan data laporan seperti nilai maksimum, minimum dan rata-rata dari *data set*
- **Normalization**: atribut data tertentu diskalakan pada rentang nilai tertentu yang lebih distandarkan.

- **Discretization**: proses mengubah atribut numerik menjadi diskret. Contohnya adalah mengubah atribut 'umur'. Umur dapat diubah menjadi kelompok nilai yang memiliki jarak berdasarkan jenis. Contohnya adalah 'balita' direntang 2-5 tahun, anak kecil direntang 5-12 dan 'remaja' rentang 12-17 tahun.

5.2.5 *Data Selection*

Data Selection adalah proses memilih data yang tepat untuk melakukan analisis dengan *machine learning*. Sebuah *data set* memiliki dua atau lebih atribut. Data perlu dipilih dengan mengambil atribut yang relevan dan mengabaikan atribut yang tidak relevan.*Machine Learning* akan melakukan prediksi nilai atribut yang disebut *response* / dependen berdasarkan atribut prediktor/independen Pemilihan atribut yang tepat akan meningkatkan akurasi dalam pembuatan *model*. Terdapat beberapa teknik yang dapat digunakan untuk memilih fitur/prediktor yang tepat.

5.2.5.1 Pearson Correlation

Pearson Correlation adalah teknik yang dapat digunakan untuk memeriksa korelasi antara 2 atribut numerik yaitu A dan B. Berikut adalah rumus dari penghitungan koefisien korelasi :

$$^{r}A, B = \frac{\sum_{i=1}^{n}(ai - \overline{A})(bi - \overline{B})}{n\sigma A\sigma B} \tag{4}$$

Persamaan 4 adalah rumus *pearson*. n adalah jumlah baris, ai dan bi adalah nilai A baris i dan nilai B baris i, $\overline{A}$ dan $\overline{B}$ adalah nilai rata-rata dari A dan B, σA dan σB adalah nilai standar deviasi dari A dan B. Jika hasil perhitungan koefisien korelasi lebih dari 0, maka A dan B memiliki hubungan **korelasi positif**. Hubungan korelasi positif adalah kondisi ketika naiknya nilai A maka nilai B juga akan naik.

Jika hasil perhitungan koefisien korelasi adalah 0, maka A dan B adalah atribut **independen** dan tidak memiliki korelasi. Jika hasil perhitungan koefisien korelasi adalah lebih kecil dari 0, maka A dan B memiliki hubungan **korelasi negatif**. Korelasi negatif adalah hubungan dimana ketika satu atribut nilainya semakin bertambah, maka atribut lain akan berkurang. *Scatter plot* juga dapat digunakan untuk melihat korelasi antara 2 atribut.

5.2.5.2 Chi Square (X^2)

Chi Square adalah teknik yang dapat digunakan untuk memeriksa korelasi antara 2 atribut kategori. Berikut adalah rumus dari perhitungan *chi square* yaitu :

$$X_c^2 = \sum \frac{(O_i - E_i)^2}{E_i} \tag{5}$$

Persamaan 5 adalah rumus untuk menghitung *chi Square*. c adalah tiap kejadian atribut respon dan prediktor. O_i adalah jumlah *observed value* yaitu jumlah kemunculan prediktor i pada respon i. *Expected value* dapat dihitung dengan :

$$E1 = n * p \tag{6}$$

$$p = P(predictor_i) * P(response_i) \tag{7}$$

Persamaan 7 adalah perhitungan untuk mendapatkan nilai *Expected*. Penjumlahan dari setiap perhitungan kemungkinan pasangan atribut prediktor dan respon akan menjadi *chi square*. Semakin tinggi nilai *chi square*, maka semakin relevan sebuah pasangan atribut prediktor dan respon digunakan.

4. **Melakukan studi literatur mengenai metode metode *machine learning* yaitu *clustering* dan *classification* yang relevan**
   Status : Ada sejak rencana kerja skripsi.
   Hasil :

5.3 Machine Learning

Machine learning adalah metode yang dapat dilakukan komputer untuk belajar berdasarkan data. Dengan *machine learning*, komputer dapat mengambil sebuah keputusan atau memrediksi. Komputer akan mencari pola dari kumpulan sampel data yang disebut dengan *training data*. *Machine learning* juga disebut sebagai *predictive analysis* karena dapat membantu komputer untuk mengambil sebuah keputusan. *Machine Learning* dapat dibedakan berdasarkan jenis *input* dan *output* yang dihasilkan.Jenis-jenis kategori *Machine Learning* yaitu :

- ***Supervised Learning*** : algoritma yang menerima kumpulan sampel data yang dijadikan *training data* dan menghasilkan *output* berupa jenis kelas dari data tersebut. *Supervised learning* menerima data yang sudah memiliki label agar dapat memrediksi data baru berdasarkan *training data*. Algoritma *supervised learning* antara lain adalah *Regression* dan *Classification*. *Supervised learning* disebut sebagai *predictive analysis* karena kemampuannya untuk memprediksi nilai
- ***Unsupervised Learning*** : algoritma yang menerima kumpulan sampel data *training* yang belum memiliki label. *Unsupervised learning* dapat digunakan untuk mengelompokkan kumpulan data berdasarkan nilai dan kesamaan. Algoritma *unsupervised learning* adalah *clustering*. *Unsupervised Learning* disebut sebagai *descriptive analysis* karena kemampuannya untuk mengelompokkan data sesuai kemiripan dan mendeskripsikannnya.

5.3.1 Regression

Regression adalah teknik *supervised learning* yang digunakan untuk memrediksi nilai kontinu. *Regression* menerima sampel data numerik sebagai input untuk menghasilkan sebuah persamaan yang dapat digunakan untuk memrediksi nilai yang dibutuhkan. *Regression* memrediksi nilai variabel bergantung (*response*) berdasarkan nilai variabel independen. Terdapat beberapa jenis *regression* yaitu *linear regression* dan *polynomial regression*.

5.3.1.1 Linear Regression

Linear Regression adalah algoritma *regression* yang digunakan untuk menghasilkan persamaan linear. *Linear regression* juga dapat digunakan untuk menguji sejauh mana hubungan sebab akibat antara variabel dependen dengan variabel independen. *Linear Regression* dapat digunakan untuk memrediksi nilai kontinu. Persamaan *linear regression* yaitu :

$$Y = a + bX \tag{8}$$

Nilai Y pada persamaan (8) merupakan atribut variabel dependen (*response*). X merupakan atribut variabel independen (*predictor*). a merupakan konstanta dan b merupakan koefisien regresi (kemiringan). Koefisien regresi merupakan besaran *response* yang ditimbulkan oleh *predictor*. Nilai-nilai a dan b dapat dihitung dengan menggunakan rumus yaitu :

$$a = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2} \tag{9}$$

$$b = \frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2} \tag{10}$$

Berdasarkan uraian rumus (10) dan rumus (9), seberapa kuatnya pengaruh variabel atribut independen terhadap variabel atribut dependen dapat dihitung menggunakan koefisien determinasi ($\mathbf{R^2}$). **Koefisien determinasi** adalah nilai yang menunjukkan kuat/tidaknya hubungan antara dua variabel. Berikut adalah cara menghitung koefisien determinasi :

$$R^2 = \frac{((n)(\sum XY) - (\sum X)(\sum Y))^2}{(n(\sum X^2) - (\sum X)^2(n(\sum Y^2)) - (\sum Y)^2)} \tag{11}$$

Uraian rumus (11) diatas merupakan cara untuk menghitung koefisien determinasi. Terdapat 3 kemungkinan dalam hasil perhitungan koefisien determinasi. Jika nilai $R^2 > 0$, maka kedua atribut memiliki korelasi positif. Korelasi positif terjadi saat suatu nilai atribut meningkat (X), maka atribut lainnya (Y) juga meningkat. Jika nilai $R^2 < 0$, maka kedua atribut memiliki korelasi negatif. Korelasi negatif terjadi saat nilai atribut meningkat, maka atribut lainnya (Y) juga akan menurun. Jika nilai $R^2 = 0$, maka kedua atribut tidak memiliki korelasi sama sekali.

Untuk mempermudah penjelasan, maka diberikan contoh perhitungan menggunakan *linear regression*. Terdapat *dataset* dengan 2 variabel yaitu suhu ruangan dan jumlah cacat produksi. Berikut adalah isi dari *data set* :

Biaya Promosi (X)	Volume Penjualan (Y)
12	56
14	62
13	60
12	61
15	65
13	66
14	60
15	63
13	65
14	62

Tabel 1: tabel dataset

Biaya promosi pada Tabel 1 merupakan variabel prediktor / independen (X). Volume penjualan pada tabe 1 merupakan varibel dependen / respon (Y). Tujuan *linear regression* adalah memprediksi volume penjualan berdasarkan biaya promosi yang dikeluarkan. Menggunakan persamaan *linear regression*, maka kita harus mencari a (konstanta) dan b(koefisien) pada persamaan 8. Berikut adalah tabel perhitungan $\sum y$, $\sum x^2$, $\sum xy$, $(\sum x)^2$ pada Tabel 2 adalah :

no	x	y	x^2	xy
1	12	56	144	672
2	14	62	196	868
3	13	60	169	780
4	12	61	144	732
5	15	65	225	975
6	13	66	169	858
7	14	60	196	840
8	15	63	225	945
9	13	65	169	845
10	14	62	196	868
$\sum$	135	620	1833	8383

Tabel 2: perhitungan komponen konstanta dan koefisien

Setelah menghitung komponen untuk konstanta dan koefisien, maka akan diterapkan pada rumus konstanta dan koefisien pada persamaan 10 yaitu :

$$a = \frac{(620)(1833) - (135)(8383)}{10(1833) - 18255} = 45.29 \tag{12}$$

$$b = \frac{10(8383) - (135)(620)}{10(1833) - 18255} = 1.24 \tag{13}$$

Koefisien dan konstanta yang diperoleh pada persamaan 13 dan 12 dapat diterapkan pada rumus *linear regression* pada persamaan 8. Persamaan setelah didapatkan konstanta dan koefisiennya adalah :

$$Y = 45.29 + 1.24X \tag{14}$$

Menggunakan persamaan 14 yang sudah diperoleh , maka prediksi volume penjualan menggunakan *linear regression* dapat dihitung menggunakan biaya promosi. Berikut adalah tabel perbandingan antara volume penjualan asli (Y) dan volume penjualan prediksi *linear regression* ($\hat{Y}$) pada Tabel 3

Biaya Promosi (X)	Volume Penjualan (Y)	Volume Penjualan Prediksi ($\hat{Y}$)
12	56	60.14
14	62	62.62
13	60	61.38
12	61	60.14
15	65	63.86
13	66	61.38
14	60	62.61
15	63	63.86
13	65	61.38
14	62	62.61

Tabel 3: perbandingan volume penjualan prediksi dan volume penjualan asli

5.3.1.2 Polynomial Regression

Polynomial Regression adalah algoritma *regression* yang digunakan untuk menghasilkan persamaan polinomial (suku banyak) berdasarkan data yang diinput. Persamaan akan berubah sesuai dengan *degree* / orde yang ditentukan. Persamaan yang dihasilkan dapat digunakan untuk menghitung nilai variabel bergantung menggunakan nilai variabel independen dan koefisien yang ditemukan. Persamaan *polynomial regression* adalah :

$$y = a0 + a_1 x + a_2 x^2 + .. + a_n x^n \tag{15}$$

Nilai Y pada persamaan 15 merupakan variabel dependen yang ingin diprediksi. Nilai a0 sampai aN adalah nilai koefisien. Nilai X adalah atribut variabel independen. Sebelum menggunakan persamaan diatas, kita perlu untuk mencari koefisien dari setiap suku perpangkatan. Untuk persamaan polinomial orde 2 didapatkan hubungan yaitu :

$$\begin{cases} na_0 + (\sum_{i=1}^{n} xi)a_1 + (\sum_{i=1}^{n} xi^2)a_2 & = \sum_{i=1}^{n} yi \\ (\sum_{i=1}^{n} xi)a_0 + (\sum_{i=1}^{n} xi^2)a_1 + (\sum_{i=1}^{n} xi^3)a_2 & = \sum_{i=1}^{n}(x_i y_i) \\ (\sum_{i=1}^{n} xi^2)a_0 + (\sum_{i=1}^{n} xi^3)a_1 + (\sum_{i=1}^{n} xi^4)a_2 & = \sum_{i=1}^{n}(x_i^2 y_i) \end{cases} \tag{16}$$

Berdasarkan persamaan 16, dapat diubah dalam bentuk matriks persamaan untuk mendapatkan koefisien yaitu :

$$\begin{bmatrix} n & \sum_{i=1}^{n} xi & \sum_{i=1}^{n} xi^2 \\ \sum_{i=1}^{n} xi & \sum_{i=1}^{n} xi^2 & \sum_{i=1}^{n} xi^3 \\ \sum_{i=1}^{n} xi^2 & \sum_{i=1}^{n} xi^3 & \sum_{i=1}^{n} xi^4 \end{bmatrix} \begin{bmatrix} a0 \\ a1 \\ a2 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} yi \\ \sum_{i=1}^{n}(x_i y_i) \\ \sum_{i=1}^{n}(x_i^2 y_i) \end{bmatrix} \tag{17}$$

Persamaan (17) dapat menemukan masing-masing koefisien sehingga dapat dimasukan pada persamaan *Polynomial Regression*. Hasil perhitungan koefisien dapat diaplikasikan pada rumus persamaan sehingga bisa menghitung prediksi nilai atribut respon (dependen). Untuk mempermudah penjelasan,

maka diberikan contoh perhitungan menggunakan *polynomial regression*. Terdapat *dataset* dengan 2 variabel yaitu suhu ruangan dan jumlah cacat produksi. Berikut adalah isi dari *data set* :

Biaya Promosi (X)	Volume Penjualan (Y)
12	56
14	62
13	60
12	61
15	65
13	66
14	60
15	63
13	65
14	62

Tabel 4: tabel dataset

Biaya promosi pada tabel 4 merupakan variabel prediktor / independen (X). Volume penjualan pada tabe 1 merupakan varibel dependen / respon (Y). Tujuan *polynomial* adalah memprediksi volume penjualan berdasarkan biaya promosi yang dikeluarkan. Menggunakan persamaan *polynomial regression*, maka perlu menghitung komponen untuk persamaan (16) hubungan berupa $\sum x$, $\sum y$, $\sum x^2$, $\sum x^3$, $\sum x^4$, $\sum xy$ dan $\sum x^2y$ pada Tabel 5 yaitu :

no	x	y	x^2	x^3	x^4	xy	x^2y
1	12	56	144	1728	20736	672	9064
2	14	62	196	2744	48416	868	12152
3	13	60	169	2197	28561	780	10140
4	12	61	144	1728	20736	732	8784
5	15	65	225	3375	50625	975	14625
6	13	66	169	2197	28561	858	11154
7	14	60	196	2744	38416	840	11760
8	15	63	225	3375	50625	945	14175
9	13	65	169	2197	28561	845	10985
10	14	62	196	2744	38416	868	12152
$\sum$	135	620	1833	25029	343653	8383	113991

Tabel 5: perhitungan komponen matriks hubungan polinom orde 2

Perhitungan Tabel 5 akan dimasukkan ke persamaan hubungan polinom yaitu :

$$\begin{cases} (10)a_0 + (135)a_1 + (1833)a_2 & = 620 \\ (135)a_0 + (1833)a_1 + (25049)a_2 & = 8383 \\ (1833)a_0 + (25049)a_1 + (343653)a_2 & = 113991 \end{cases}$$

Menghitung koefisien a0,a1 dan a2 dapat menggunakan operasi matriks perkalian invers yaitu :

$$\begin{bmatrix} 10 & 135 & 1833 \\ 135 & 1833 & 25049 \\ 1833 & 25049 & 343653 \end{bmatrix} \begin{bmatrix} a0 \\ a1 \\ a2 \end{bmatrix} = \begin{bmatrix} 620 \\ 8383 \\ 113991 \end{bmatrix}$$

Berdasarkan perhitungan matriks, maka didapatkan a0 = -67.96428571 , a1 = 18.11309524 dan a2 = -0.625. Hasil perhitungan tiap koefisien dapat dimasukkan pada persamaan 15 sehingga membentuk persamaan untuk menghitung volume penjualan (Y) menggunakan biaya promosi (X) dengan :

$$Y = -67.96 + 18.11X + (-0.63)X^2$$

Menggunakan persamaan yang sudah diperoleh, maka prediksi volume penjualan (Y) menggunakan *polynomial regression* dapat dihitung menggunakan biaya promosi. Berikut adalah perbandingan antara volume penjualan asli (Y) dan volume penjualan prediksi *polynomial regression* ($\hat{Y}$) pada Tabel 6 yaitu :

Biaya Promosi (X)	Volume Penjualan (Y)	Prediksi Volume Penjualan ($\hat{Y}$)
12	56	59.39
14	62	63.11
13	60	61.88
12	61	59.39
15	65	63.10
13	66	61.88
14	60	63.11
15	63	63.10
13	65	61.88
14	62	63.11

Tabel 6: perbandingan prediksi volume penjualan (Y) polynomial regression

5.3.1.3 Evaluasi Regresi Linear

Metode yang dapat digunakan untuk menguji seberapa baik hasil prediksi atribut respon yang ingin diperoleh adalah dengan membandingkan selisih antara atribut respon pada *test set* (y true) dengan atribut respon yang diprediksi (y pred) menggunakan model ini. Selisih antara prediksi dan yang asli disebut dengan *error*. *Mean Squared Error* (MSE) adalah rata-rata dari nilai *error* setiap data objek. Rumus dari MSE (18) adalah :

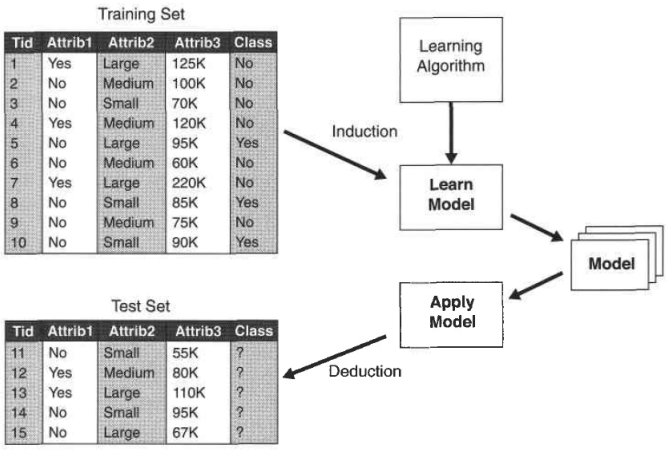
$$\frac{1}{N}\sum_{i=1}^{n}(ytrue_i - ypred_i) \tag{18}$$

Root Mean Squared Error (RMSE) adalah akar pangkat dari MSE yang akan memberikan nilai yang lebih dapat dinormalkan. Berikut adalah rumus dari RMSE (19) yaitu :

$$\sqrt{\frac{1}{N}\sum_{i=1}^{n}(ytrue_i - ypred_i)} \tag{19}$$

5.3.2 Classification

Classification merupakan metode *supervised learning* selain regresi. *Classification* dilakukan untuk menetapkan label data / kategori pada sebuah *data object*. *Classification* merupakan proses untuk memetakan variabel prediktor / independen (X) terhadap variabel respon / dependen (Y). Terdapat beberapa algoritma *classification* yang dapat digunakan untuk memrediksi nilai label pada *data set*.



Gambar 2: *classification model*

Gambar 5.3.2 menunjukkan sebuah proses dalam melakukan *classification*. *Classifier* / *classification technique* adalah sebuah cara untuk membuat model dari *input data set*. Pada awalnya, *dataset* dapat dibagi menjadi 2 bagian yaitu *training set* dan *test set*. Terdapat *training set* yaitu kumpulan data / *record* yang sudah memiliki label data akan di*train* modelnya dengan menggunakan algoritma *classification*. Model yang sudah dibuat akan digunakan untuk memrediksi nilai label dari data yang ingin diprediksi yaitu *test set* untuk diuji keakuratannya. Hasil nilai label yang sudah diprediksi dapat dinilai dengan cara membandingkan hasil prediksi label dengan label asli pada *test set*.

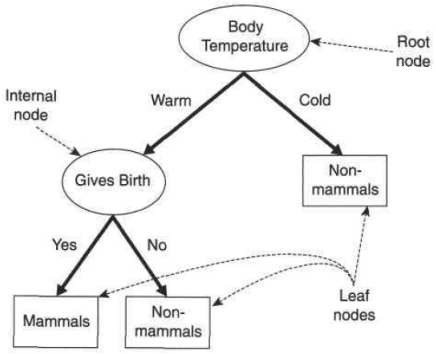
Dalam memanfaatkan *classification* untuk memrediksi nilai label, terdapat beberapa algoritma yang dapat dimanfaatkan untuk melakukan *classification* yaitu *Decision Tree* ,*k Nearest Neighbors* (kNN), dan *Naive Bayes Classifier*.

5.3.2.1 Decision Tree

Decision tree adalah teknik *classifier* dapat digunakan untuk menentukan label data pada *test set*. *Decision tree* akan membentuk sebuah pohon keputusan berdasarkan atribut prediktor yang akan digunakan untuk membuat model. Pohon keputusan yang dihasilkan akan menjadi aturan bagaimana menentukan label dari *data set*. Membuat *Decision tree* akan memanfaatkan *training set* untuk menentukan berdasarkan apa pohon dapat di*split*.

Struktur sebuah pohon keputusan memiliki 3 *node* yaitu :

- ***Root node*** : tidak memiliki *incoming edge* dan tidak atau banyak *edges*
- ***Internal nodes*** : memiliki satu *incoming edge* dan banyak *outgoing edges*
- ***Leaf / terminal nodes*** memiliki satu *incoming edge* dan tidak ada *outgoing edges*



Gambar 3: ilustrasi Decision Tree

Gambar 5.3.2.1 menunjukkan sebuah contoh bagaimana penerapan pohon keputusan pada sebuah *dataset* binatang. Tujuan dari pohon keputusan adalah untuk menentukan aturan apakah sebuah binatang termasuk *mammals* atau *non-mammals*. Atribut prediktor yang terdapat ada *root node*dan *internal node*seperti *Body temperature* dan *Given Birth* digunakan untuk memecah *data* agar dapat menentukan label.

Teknik ini dapat menghasilkan berbagai pohon keputusan berdasarkan attribut yang dipilih sebagai atribut pembagi. Untuk menghasilkan *model* yang akurat, dibutuhkan cara untuk menentukan atribut yang dapat membagi pohon. Cara memilih atribut adalah dengan mengukur tingkat *impurity* / ketidakmurnian dari *node*. Tujuan dari pembuatan pohon keputusan ini adalah untuk membuat semua *node pure* / murni. Sebuah *node* dapat dikatakan murni jika sudah memiliki label data.

Entropy adalah ukuran *randomness* / ketidakpastian sebuah data. Semakin rendah nilai *entropy*, maka semakin murni *node* tersebut. Rumus dari entropi adalah yaitu :

$$Entropy(t) = -\sum_{i=0}^{c-1} p(i|t) \log_2(i|t) \tag{20}$$

Berdasarkan uraian rumus (20) , perhitungan entropi merupakan penjumlahan peluang tiap kelas i terhadap semua jumlah data t. Untuk menentukan atribut mana yang dapat digunakan sebagai pemecah, kita butuh untuk membandingkan tingkat ketdakmurnian *parent node* (sebelum dipecah) dengan tingkat ketidakmurnian dari *child node* (setelah dipecah). ***Information Gain*** (Δ info) adalah sebuah kriteria yang dapat digunakan untuk menentukan atribut pembagi dalam pembentukan pohon. Rumus dari *information gain* yaitu :

$$\Delta info = I(parent) - \sum_{j=1}^{k} \frac{N(vj)}{N} I(jv) \tag{21}$$

rumus (21) I adalah nilai ketidakmurnian yang dapat diperoleh menggunakan *entropy*. N adalah total observasi data / jumlah *record* dari *parent node*, k adalah jumlah nilai dari suatu atribut, dan vj adalah jumlah *record* pada nilai atribut j. *Decision* tree akan memilih atribut prediktor mana dengan memilih nilai Δ (info) yang maksimum.

5.3.2.2 k-Nearest Neighbor Classifiers

k-Nearest Neighbor adalah algoritma *classification* untuk menentukan label sebuah data. *k-Nearest Neighbor* akan menentukan label data baru berdasarkan *dataset* yang sudah memiliki label pada *training set*. Menentukan label untuk data baru dapat ditentukan dengan mencari *training data* yang

terdekat dengan data baru. Label pada *k*-jarak terdekat menjadi label bagi data baru. Setiap data pada *dataset* dapat dianalogikan sebagai titik yang akan dibandingkan jaraknya. Menghitung jarak antara titik data baru dan setiap titik pada *dataset* dapat menggunakan *euclidean distance*. Menghitung *Euclidean distance* adalah dengan rumus :

$$dist(X,Y) = \sqrt{\sum_{n}^{i=1}(xi - yi)^2} \tag{22}$$

Uraian rumus (22) diatas merupakan cara menghitung jarak antara titik baru dengan suatu titik pada *training set*. X merupakan data baru dan Y merupakan suatu titik pada *training set*. Xi merupakan setiap atribut variabel pada data baru dan Yi merupakan setiap atribut variabel pada titik *training set*. Algoritma *k-Nearest Neighbor* akan menghitung *euclidean distance* sebanyak N yaitu jumlah *row* pada *training set* untuk membandingkan jarak data baru dengan setiap data pada *training set*.

Menggunakan *euclidean distance* dapat diterapkan untuk atribut dengan tipe data numerik. Tetapi, untuk atribut nominal seperti jenis warna (biru,merah, hijau dan lain-lain) membutuhkan mekanisme berbeda untuk menentukan jarak terdekat. Menentukan angkanya adalah jika nilai atribut data baru sama dengan data pada *training set* maka jaraknya adalah 0. Jika nilai atirbut data baru tidak sama dengan sebuah data pada *training set*, maka jaraknya adalah 1.

Tahap pertama dalam mencari label untuk data baru menggunakan *k-Nearest Neighbor* adalah dengan menentukan k tetangga terdekat. Cara menentukan k terbaik adalah dengan mencoba menghitung jarak pada setiap k. K terbaik dapat ditentukan dengan mencari akurasi tertinggi dari setiap percobaan k.

Untuk mempermudah penjelasan, akan diberikan contoh perhitungan menggunakan *k-Nearest Neighbor* untuk memprediksi nilai kategori pada data baru. Berikut adalah contoh *data set* yang digunakan sebagai *training set* untuk membuat model *k-Nearest Neighbor* :

x	y	kategori
7	6	Bad
6	6	Bad
6	5	Bad
1	3	Good
2	4	Good
2	2	Good

Tabel 7: tabel data set k-nearest neighbor

Tabel 7 berisi 2 variabel prediktor / independen yaitu x dan y. Kategori merupakan variabel respon / dependen. Diberikan sebuah data objek baru dengan x = 3 dan y = 5. Algoritma *k-Nearest Neighbors* memilih k tetangga terdekat. Nilai k yang ditentukan adalah 3. Menggunakan *euclidean distance*, berikut adalah perhitungan jarak setiap data pada *data set* dengan data baru yaitu :

x	y	kategori	euclidean distance dengan data baru	perhitungan
7	6	Bad	4.12	$\sqrt{(7-3)^2+(6-5)^2}$
6	6	Bad	3.16	$\sqrt{(6-3)^2+(6-5)^2}$
6	5	Bad	3	$\sqrt{(6-3)^2+(5-5)^2}$
1	3	Good	2.82	$\sqrt{(1-3)^2+(3-5)^2}$
2	4	Good	1.41	$\sqrt{(2-3)^2+(3-5)^2}$
2	2	Good	3.16	$\sqrt{(2-3)^2+(2-5)^2}$

Tabel 8: tabel perhitungan euclidean distance dengan data baru

Setelah menghitung setiap jarak dengan data baru, maka *k-Nearest Neighbors* akan memilih tetangga terdekat sebanyak k. Karena k = 3, maka akan dipilih kategori berdasarkan tiga tetangga dengan selisih jarak *euclidean distance* minimum. Berikut adalah 3 tetangga terdekat dengan data baru pada Tabel 9 yaitu :

x	y	kategori	Jarak dengan data baru
2	4	Good	1.41
1	3	Good	2.82
6	5	Bad	3

Tabel 9: 3 tetangga terdekat berdasarkan perhitungan euclidean distance

Tabel 9 menunjukkan bahwa terdapat 3 tetangga terdekat. Algoritma *k-Nearest Neighbors* akan menentukan kategori dari data baru berdasarkan jarak terdekat dan paling banyak. Pada 3 jarak terdekat, terdapat 2 kategori Good dan 1 kategori Bad. Karena jumlah Good lebih banyak daripada Bad, maka data baru akan memiliki kategori Good

5.3.2.3 Evaluasi Classification

Cara yang dapat digunakan untuk melakukan evaluasi terhadap model yang sudah dibuat adalah dengan menghitung jumlah perbandingan data yang benar dan salah. *Accuracy* adalah sebuah metode *performance metric* yang dapat digunakan untuk menghitung performa model. Cara menghitung akurasi yaitu :

$$Accuracy = \frac{Jumlahprediksiyangbenar}{Jumlahsemuaprediksi}$$

Error rate adalah cara lain yang dapat digunakan sebagai menilai performa model berdasarkan nilai *error* yang dihasilkan. Cara menghitung *error rate* adalah sebagai berikut :

$$Errorrate = \frac{Jumlahprediksiyangsalah}{Jumlahsemuaprediksi}$$

5.3.3 Clustering

Clustering merupakan teknik *unsupervised learning*. *Clustering* akan mengelompokkan tiap data objek pada *data set* menjadi beberapa kelompok. *Clustering* mengelompokkan data objek yang memiliki kemiripan menjadi satu kelompok dan data objek yang tidak memiliki kemiripan menjadi kelompok yang berbeda. Kemiripan dan ketidakmiripan dari data objek dapat ditentukan menggunakan atribut prediktor pada *dataset*. Ukuran yang dapat ditentukan untuk kemiripan tiap data objek adalah berdasarkan perhitungan jarak. Terdapat beberapa metode *clustering* yaitu :

- **Partitioning methods** : metode *clustering* dengan membuat k partisi dimana jumlah partisi (k) <= N (jumlah data objek). Tiap partisi harus minimal berisi satu data objek. Contoh algoritma *partitioning* adalah *k-Means*
- **Hierarchical methods** : metode *clustering* dengan cara melakukan proses dekomposisi menjadi hirarki.Hirarki yang dihasilkan adalah tingkatan jumlah kelompok yang dihasilkan. Terdapat dua jenis *Hierarchical methods* yaitu *agglomerative* dan *divisive*. *Agglomerative* menerapkan *bottom-up approach*. *Agglomerative* berawal dari tiap data objek pada *dataset* menjadi data yang terpisah. Masing-masing data objek digabung satu per satu sehingga membentuk satu kelompok yang besar. *Divisive* menerapkan *top-bottom approach* yaitu setiap data objek yang dijadikan satu

kelompok besar. Setiap iterasi akan dibagi menjadi kelompok-kelompok kecil sampai tiap data objek merupakan kelompok yang terpisah.

- **Density-based methods** : metode *clustering* dengan cara membagi kumpulan data objek menjadi bentuk kelompok yang lebih kompleks. Contoh algoritma *density* adalah *Density Based Spatial Clustering* (DBSCAN). Cara kerjanya adalah setiap data objek yang merupakan titik secara acak diambil untuk ditentukan jenisnya. Suatu titik merupakan *core points* jika masih memiliki tetangga yang jaraknya lebih kecil dari epsilon.Jika tidak terdapat titik tetangga terdekat, titik dijadikan sebuah *border points*. Setelah itu berpindah ke titik tetangga *core points* terdekat lalu diulangi lagi sampai semua menjadi mempunyai *cluster* atau *outlier*

5.3.3.1 k-Means

k-Means adalah algoritma *clustering* jenis *partitioning*. Algoritma ini mengelompokkan data objek dengan memaksimalkan kemiripan. Ukuran kemiripan antar data objek dapat diukur menggunakan jarak. Jumlah kelompok akan ditentukan berdasarkan nilai k yang ditentukan. Setelah nilai k telah ditentukan, maka akan dibuat data objek secara *random* atau mengambil salah satu data pada *data set* yang disebut sebagai *centroid*. *Centroid* merupakan nilai tengah yang merepresentasikan tiap *cluster*. Tiap data objek akan dihitung jaraknya dengan setiap *centroid* yang diinisialisasikan. Data objek akan dimasukkan ke kelompok *cluster* yang memiliki jarak terkecil dengan *centroid*. Jarak data objek dengan centroid dapat dihitung dengan menggunakan *euclidean distance*:

$$dist(X,Y) = \sqrt{\sum_{n}^{i=1}(xi - yi)^2} \tag{23}$$

Persamaan (24) menghitung jarak antara 2 data objek. X adalah data objek pertama dan Y adalah data objek kedua. $\sum$ adalah melakukan iterasi dan penjumlahan setap atribut. Setiap atribut masing-masing data objek akan dikurang lalu dipangkatkan. Hasil perpangkatan selisih setiap atribut akan diakarpangkat dua sehingga menjadi jarak dari kedua data objek

Langkah-langkah dalam melakukan *k-Means clustering* adalah :

(a) Tentukan jumlah K (jumlah *cluster* / kelompok)

(b) Untuk tiap *cluster*, buat satu data objek sebagai titik tengah yang disebut sebagai *centroid*. Inisialisasi *centroid* dapat ditentukan dengan *random* / mengambil salah satu titik dari *data objek*

(c) Untuk setiap data objek, hitung jarak menggunakan *euclidean distance* dengan setiap *centroid*. Data objek akan masuk ke kelompok yang memiliki *centroid* dengan jarak terpendek

(d) Atur ulang posisi *centroid* dengan menghitung rata-rata tiap atribut yang termasuk dalam *cluster*

(e) Ulangi tahap tiga sampai tiap data objek tidak mengalami perubahan dalam menetapkan *cluster*

Untuk mempermudah penjelasan, diberikan sebuah contoh perhitungan *k-Means* untuk menetapkan kelompok data. Berikut adalah contoh *data set* nilai

Tabel 10 memiliki informasi / atribut berupa nilai UTS,ART dan UAS. K atau jumlah *cluster* yang ditentukan adalah K=2. Titik awal *centroid* yaitu c1 dan c2. *Centroid* ditentukan secara *random* adalah :

- c1 : UTS(80) , ART(80),UAS(80)
- c2 : UAS(50) , ART(50), UAS(50)

Nama	UTS	ART	UAS
Jonathan	75	40	95
James	90	95	100
Pedro	55	90	75
Luna	85	65	85
Harry	85	80	80
Chloe	55	50	51

Tabel 10: dataset k-Means

Berikut adalah perhitungan tiap data objek dengan menggunakan *euclidean distance* pada persamaan (24) :

Nama	UTS	ART	UAS	Dist(data(i),c1)	Dist(data(i),c2)	Kelompok
Jonathan	75	40	95	$\sqrt{(75-80)^2+(40-80)^2+(95-80)^2} = 43.01$	$\sqrt{(75-50)^2+(40-50)^2+(95-50)^2} = 53.44$	c1
James	90	95	100	$\sqrt{(90-80)^2+(95-80)^2+(100-80)^2} = 26.92$	$\sqrt{(90-50)^2+(95-50)^2+(100-50)^2} = 78.26$	c1
Pedro	55	90	75	$\sqrt{(55-80)^2+(90-80)^2+(75-80)^2} = 27.38$	$\sqrt{(55-50)^2+(90-50)^2+(75-50)^2} = 47.43$	c1
Luna	85	65	85	$\sqrt{(85-80)^2+(65-80)^2+(85-80)^2} = 16.58$	$\sqrt{(85-50)^2+(65-50)^2+(85-50)^2} = 51.72$	c1
Harry	85	80	80	$\sqrt{(85-80)^2+(80-80)^2+(80-80)^2} = 5$	$\sqrt{(85-50)^2+(80-50)^2+(80-50)^2} = 55$	c1
Chloe	55	50	51	$\sqrt{(55-80)^2+(50-80)^2+(51-80)^2} = 48.64$	$\sqrt{(55-50)^2+(50-50)^2+(51-50)^2} = 5.09$	c2

Tabel 11: perhitungan k-Means iterasi ke-1

Tabel 11 menunjukkan perhitungan iterasi pertama untuk mendapatkan kelompok tiap data objek. Selanjutnya titik *centroid* diubah dengan menghitung rata-rata atribut tiap anggota kelompok pada Tabel 12 :

Nama	UTS	ART	UAS
c1	$(75+90+55+85+85)/5 = 78$	$(40+95+90+65+80)/5 = 74$	$(95+100+75+85+80)/5 = 87$
c2	55	50	51

Tabel 12: perubahan centroid iterasi 2

Setelah *centroid* sudah diatur ulang, maka perhitungan tiap data objek dengan *centroid* yang baru adalah :

Karena pada Tabel 13 tiap data objek tidak berpindah ke *cluster* lain, sehingga algoritma *k-Means* tidak perlu diulang kembali karena sudah konvergen.

5.3.3.2 Agglomerative Nesting (AGNES)

Agglomerative clustering adalah algoritma *clustering* jenis *hiearchical*. Algoritma ini merupakan metode *bottom-up* yaitu setiap data objek sebagai masing-masing *cluster*. Secara iteratif menghitung jarak tiap data objek. Dua data objek dengan jarak terkecil akan digabung menjadi sebuah *cluster*. Proses ini diulangi sampai semua data objek tergabung menjadi satu *cluster*.

Jarak tiap titik dapat dihitung menggunakan *euclidean distance*. Berikut adalah rumus untuk menghitung *euclidean distance* pada Persamaan 24 yaitu :

$$dist(X,Y) = \sqrt{\sum_{n}^{i=1}(xi-yi)^2} \tag{24}$$

Untuk mempermudah penjelasan, diberikan sebuah contoh perhitungan *Agglomerative clustering* meng-

Nama	UTS	ART	UAS	dist(c1)	dist(c2)	kelompok
Jonathan	75	40	95	35.05	49.35	c1
James	90	95	100	27.45	75.17	c1
Pedro	55	90	75	30.47	46.64	c1
Luna	85	65	85	11.57	51.39	c1
Harry	85	80	80	11.57	51.39	c1
Chloe	55	50	51	49	0	c2

Tabel 13: perhitungan k-Means iterasi ke-2

gunakan *data set* berisi nilai yaitu :

Nama	UTS	ART	UAS
Jonathan	74	40	95
James	90	95	100
Pedro	55	90	75
Luna	85	65	85

Tabel 14: data set perhitungan agglomerative

Tabel 14 berisi komponen nilai tiap siswa yaitu UTS,ART dan UAS. Berikut adalah perhitungan jarak tiap data objek menggunakan *euclidean distance* yaitu :

	Jonathan	James	Pedro	Luna
Jonathan	0			
James	$\sqrt{(75-90)^2+(40-95)^2+(95-100)^2} = 57.22$	0		
Pedro	$\sqrt{(75-55)^2+(40-90)^2+(95-75)^2} = 57.44$	43.4	0	
Luna	28.72	33.91	40.31	0

Tabel 15: perhitungan euclidean distance iterasi pertama

Tabel 15 berisi perhitungan jarak tiap data objek. Karena jarak Jonathan dan Luna paling kecil, maka akan dikelompokkan menjadi satu *cluster* yang sama yaitu c1. Satu *cluster* yang terdiri dari beberapa titik dapat dihitung berdasarkan rata-rata tiap atribut. Sehingga isi *data set* menjadi :

Tabel 16 menunjukkan perubahan titik tengah *cluster* berasal dari rata-rata atribut Jonathan dan Luna. Iterasi selanjutnya akan menghitung jarak antara tiap titik untuk mencari jarak terpendek yaitu :

Euclidean Distance	Jonathan,Luna	James	Pedro
Jonathan,Luna	$75 + 85/2 = 80$	$40 + 65/2 = 52.5$	$95 + 85/2 = 85$
James	90	95	100
Pedro	55	80	75

Tabel 16: perubahan nilai cluster iterasi 1

Euclidean Distance	Jonathan,Luna (c1)	James	Pedro
Jonathan,Luna (c1)	0		
James	44.79	0	
Pedro	40.07	45.55	0

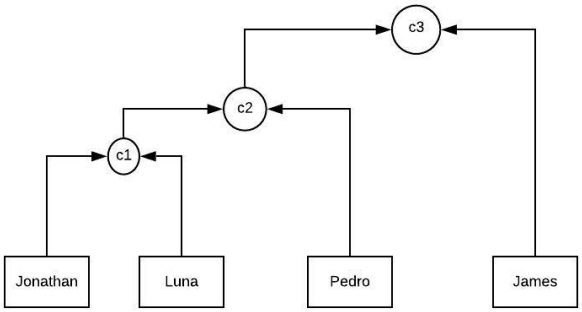
Tabel 17: perhitungan euclidean distance iterasi kedua

Tabel 17 menunjukkan bahwa jarak antara c1 (Jonathan,Luna) dan Pedro memiliki jarak terpendek sehingga dikelompokkan menjadi satu *cluster*. Setelah Pedro digabungkan dengan c1 menjadi c2, maka titik tengah dari c2 (Jonathan,Luna,Pedro) adalah :

	UTS	ART	UAS
c2 (Jonathan,Luna, Pedro)	$(80 + 55)/2 = 67.5$	$(52.55 + 80)/2 = 66.25$	$(90 + 75)/2 = 82.5$
James	90	95	100

Tabel 18: perubahan nilai cluster iterasi 2

Tabel 18 menunjukkan bahwa tinggal terdapat 2 data objek yaitu *cluster* c2 dan James. Kedua data objek dapat digabung langsung tanpa harus menghitung jarak menjadi c3 (Jonathan, James ,Pedro, Luna). Visualisasi menggunakan dendogram mengenai proses *agglomerative clustering* yaitu :



Gambar 4: Dendogram example

5. **Melakukan studi literatur mengenai teori dan implementasi visualisasi data seperti *historgram,scatter plot, box plot* untuk membantu mengetahui sifat data yang dikumpulkan menggunakan *matplotlib***
   Status: Ada sejak rencana kerja skripsi
   Hasil:

5.4 Data Visualization

Visualisasi data adalah suatu metode untuk merepresentasikan data. Visualisasi data menggunakan dengan jumlah yang banyak untuk ditampilkan secara grafis. Visualisasi data akan membantu *user* untuk membaca data lebih mudah. Dengan visualisasi, maka kita juga akan mendapatkan informasi

mengenai hubungan antara data. Visualisasi data akan membantu meringkaskan data dan memberikan pengetahuan baru.

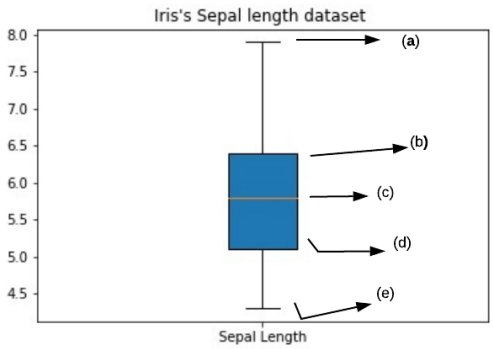
Penggunaan visualisasi data dapat digunakan pada saat *data preprocessing* maupun setelahnya. Pada tahap *data preprocessing*, visualisasi data dapat dimanfaatkan untuk membantu proses memeriksa data untuk memastikan kebersihannya. Visualisasi data tidak hanya dilakukan sekali tetapi membutuhkan beberapa kali visualisasi untuk membantu menganalisis tren dari perubahan data itu sendiri.

Visualisasi data dapat dibedakan berdasarkan tipe data yang dianalisis, jumlah atribut yang digunakan, dan bentuk data yang dibutuhkan. Terdapat beberapa teknik yang dilakukan untuk melakukan visualisasi data yaitu *Box plot*, *Histogram*, *Scatter plot* dan *Pie chart*

5.4.1 Boxplot

.

Boxplot adalah teknik *data visualization* yang dapat digunakan untuk menampilkan persebaran nilai satu atribut.Masukan data dari grafik ini adalah kumpulan atribut numerik. Teknik ini dapat membantu memberikan informasi mengenai nilai-nilai yang dapat digunakan untuk analisis data. Pada umumnya, suatu *Boxplot* akan menampilkan rata-rata(*mean*), median (Q2), nilai maksimum , nilai minimum, kuartil bawah(Q1)dan kuartil atas(Q3). Berikut adalah contoh *box plot* yang dihasilkan berdasarkan *dataset* iris pada kolom *sepal length* yaitu :



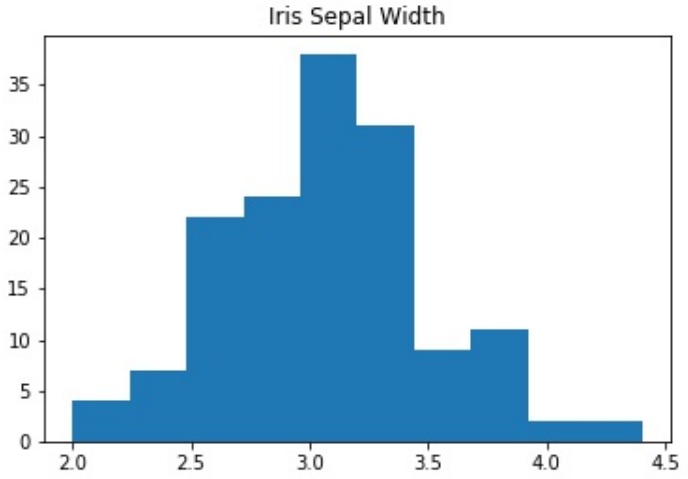
Gambar 5: *boxplot dataset iris*

Gambar 5 menunjukkan kesimpulan informasi dari persebaran *dataset* iris pada kolom *sepal length*. Berikut adalah deskripsi dari tiap huruf :

- **(a)** nilai maksimum
- **(b)** kuartil atas (Q3)
- **(c)** median (Q2)
- **(d)** kuartil bawah (Q1)
- **(e)** nilai minimum

5.4.2 Histogram

Histogram adalah teknik *data visualization* yang digunakan untuk melihat persebaran data suatu atribut. Masukan dari Data yang ditampilkan akan dibagi dalam sebuah *bin* / interval kelas. Teknik ini akan mengelompokkan jumlah *data object* berdasarkan nilai interval yang ditentukan dan sudah terurut. Berikut adalah contoh *histogram* dari *dataset iris sepal width* yaitu :



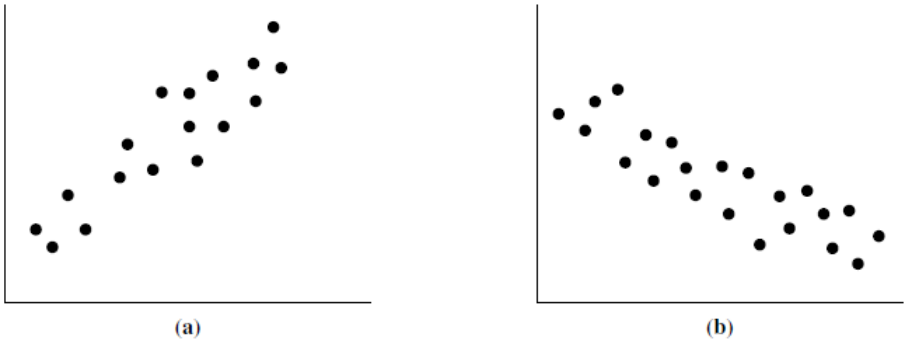
Gambar 6: *histogram dataset iris*

Histogram yang ditampilkan pada gambar diatas adalah berdasarkan *dataset iris*. Berdasarkan gambar 6, histogram satu variabel akan menunjukkan grafik dengan koordinat 2 dimensi yaitu koordinat x dan y. Nilai pada koordinat x menunjukkan tiap interval kelas. Nilai pada koordinat y menunjukkan frekuensi tiap interval data. Tiap batang pada histogram menampilkan tiap interval kelas. Tinggi batang menunjukkan frekuensi pada tiap kelas.

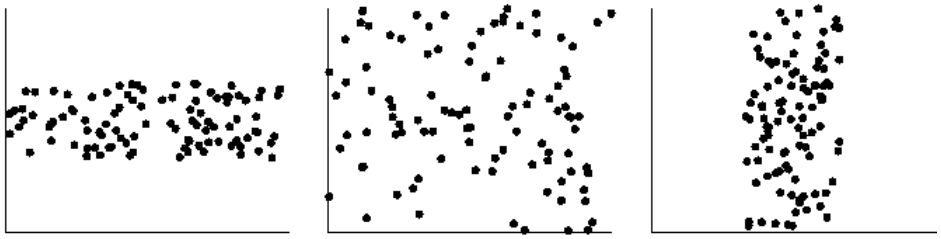
Pada contoh gambar 6, terdapat 10 *bins* yang masing-masingnya memiliki rentang sekitar nilai 0,25. Pada *bin* pertama, terdapat jarak nilai dari 2.0 sampai kurang lebih 2,25. Aturan juga diterapkan pada *bin* 2 sampai 10 yaitu menyerupai *bin* pertama.

5.4.3 Scatter Plot

Scatter plot adalah teknik *data visualization* yang dapat digunakan untuk melihat korelasi / keterhubungan antara 2 variabel data. *Scatter plot* akan menggambarkan atribut prediktor (sebab) pada koordinat X dan menggambarkan atribut respon (akibat) pada koordinat Y. Hasil *scatter plot* akan memberi penjelasan apakah atribut prediktor memiliki hubungan dengan atribut respon.Terdapat beberapa jenis *Scatter plot* berdasarkan hasil korelasinya. Terdapat *Scatter plot* yang memiliki korelasi positif, korelasi negatif dan tidak memiliki korelasi. Berikut adalah contoh gambar *scatter plot* yaitu



Gambar 7: *scatter plot dengan korelasi positif (a) dan negatif (b)*

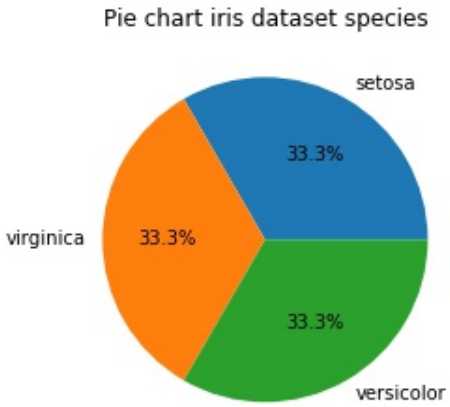


Gambar 8: *scatter plot dengan tidak ada korelasi*

Berdasarkan Gambar 7, *scatter plot* dengan korelasi terbentuk jika perubahan atribut pada koordinat X memengaruhi koordinat Y. Korelasi positif terjadi ketika semakin meningkatnya nilai pada koordinat X, maka koordinat Y akan meningkat juga. Korelasi negatif terjadi ketika perubahan koordinat X akan membuat nilai pada koordinat Y berubah berbanding terbalik. Gambar 8 menggambarkan atribut koordinat X dan koordinat Y tidak memiliki korelasi.

5.4.4 Pie Chart

Pie Chart adalah teknik *data visualization* untuk melihat frekuensi persebaran *categorial data*. *Pie chart* menggunakan bentuk lingkaran sebagai keseluruhan semua kumpulan data. Masing-masing potongan bagian pada *pie chart* merupakan presentase banyak masing-masing nilai kategori. Berikut adalah gambaran *pie chart* menggunakan *dataset iris*.



Gambar 9: *Pie Chart iris Species*

Berdasarkan Gambar 9, data yang digunakan untuk divisualisasikan adalah *a*tribut species dataset iris. Pada *dataset* terdapat atribut *species* yang memiliki kemungkinan nilai yaitu *setosa , virginica, versicolor*. Tiap warna pada *pie chart* merepresentasikan masing-masing kategori *species*. Jumlah data pada atribut *species* adalah 150, jumlah masing-masing *species* adalah 50 sehingga presentasenya sama.

6.

7. **Mempelajari bahasa pemrograman *Python* dan beberapa *library* dari *Python* seperti *Pandas*,*Sci-Kit learn* dan *matplotlib*.**
   Status : Ada sejak rencana kerja skripsi.
   Hasil :

   Analisis terhadap data *film* dilakukan untuk memahami faktor-faktor yang relevan untuk melakukan prediksi *revenue* / keuntungan *film*. Sesuai dengan referensi dari artikel *kaggle* dan latar belakang ingin menganalisis data film untuk mencari faktor-faktor yang membuat film sukses.Akan dilakukan eksperimen sederhana untuk mempelajari data *film* dan mencoba memprediksi keuntungan *film* menggunakan *linear regression*.

   Data set yang digunakan berasal dari situs *Kaggle.com*. Berikut adalah deskripsi *data set* yang digunakan. Berikut adalah deskripsi dari *data set* :

   (a) nama data set : **IMDB-Movie-Data.csv**

   (b) sumber : **kaggle**

   (c) row : **1000**

   (d) column : 12

   (e) Rank :1000 non-null int64

   (f) Title :1000 non-null object

   (g) Genre :1000 non-null object

   (h) Description : 1000 non-null object

   (i) Director :1000 non-null object

   (j) Actors :1000 non-null object

(k) Year :1000 non-null int64

(l) Runtime (Minutes) :1000 non-null int64

(m) Rating :1000 non-null float64

(n) Votes :1000 non-null int64

(o) Revenue (Millions) :872 non-null float64

(p) Metascore :936 non-null float64

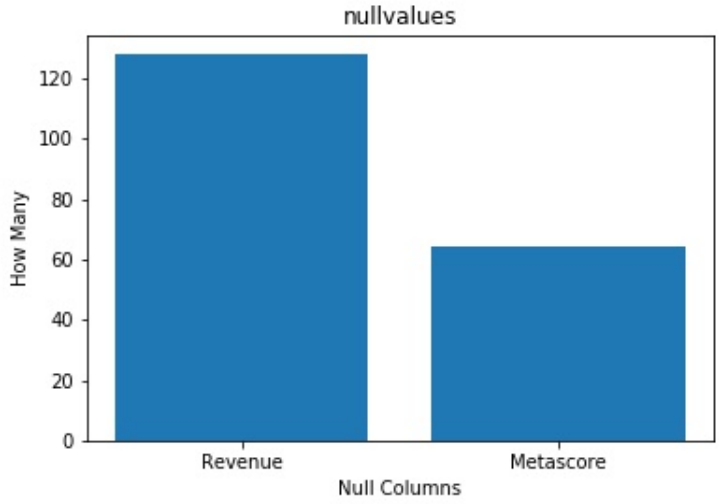
(q) dtypes: float64(3), int64(4), object(5)

Pada eksperimen kali ini saya menggunakan *data set* yang relatif kecil sebagai latihan untuk membiasakan dalam penggunaan *python* dan *dataframe*. Hal yang pertama saya lakukan adalah mengimport data menggunakan fungsi readcsv .Dataset saya peroleh dari situs Kaggle.Setelah saya berhasil melakukan readcsv,saya memanggil fungsi head() untuk melihat 5 data pertama. Berikut adalah Gambar 10 berisi pemanggilan fungsi head().

	Rank	Title	...	Revenue (Millions)	Metascore
0	1	Guardians of the Galaxy	...	333.13	76.0
1	2	Prometheus	...	126.46	65.0
2	3	Split	...	138.12	62.0
3	4	Sing	...	270.32	59.0
4	5	Suicide Squad	...	325.02	40.0

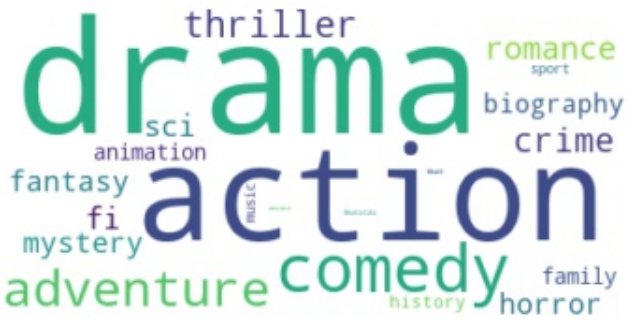
Gambar 10: *5 data pertama*

Pada tahap *pre processing* kali ini saya melakukan *data cleaning* dengan mengidentifikasi apakah ada *null* values dari dataset yang digunakan dengan menggunakan fungsi dropna(), yaitu akan membuang row jika terdapat suatu kolom yang memiliki value nan. Setelah di drop, jumlah *row* pada dataset adalah 838. Berikut adalah hasil visualisasi jumlah masing masing kolom yang memiliki *null* values menggunakan *barchart* pada Gambar 11.



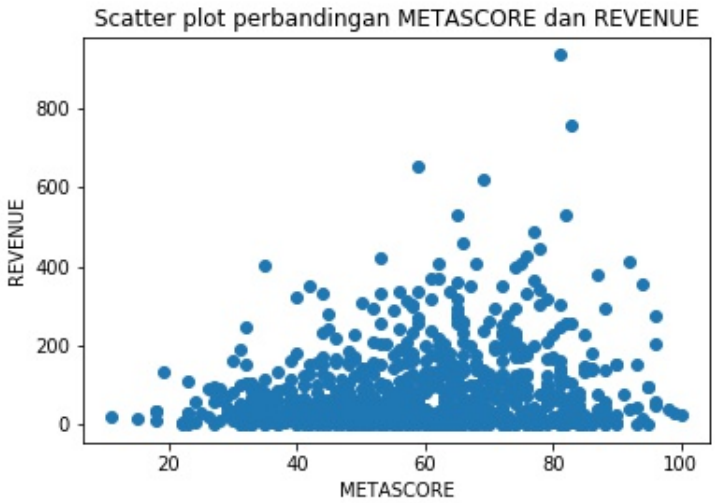
Gambar 11: *jumlah null values tiap atribut*

Kolom *genre* pada *data set* ini berisi beberapa *genre* yang di *concat* menggunakan koma, sehingga saya melakukan perubahan kolom dengan membagi menjadi 1 row 1 genre. Berikut adalah kumpulan genre yang tersedia pada dataset yang digunakan menggunakan wordcloud untuk melihat seberapa banyak masing-masing *genre*.

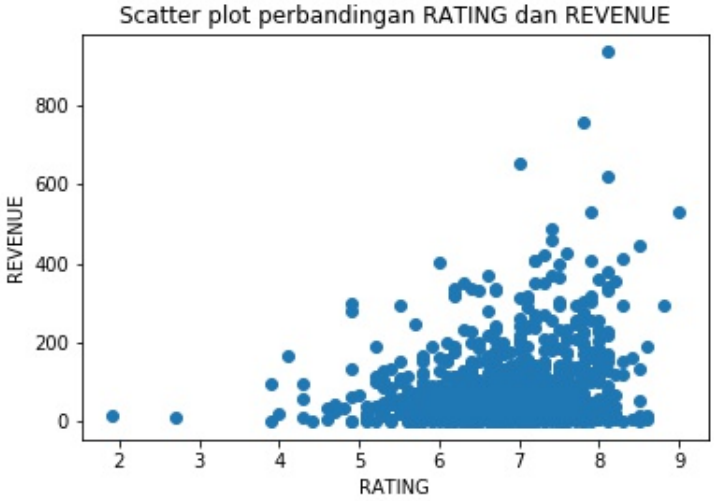


Gambar 12: *Wordcloud bobot genre*

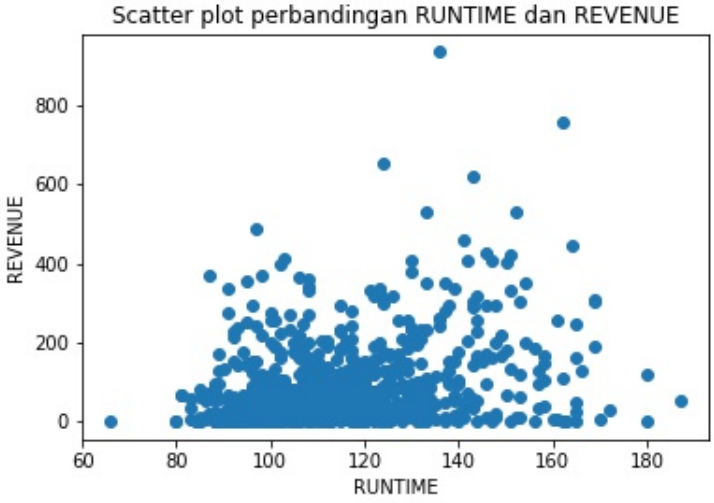
Tahap selanjutnya adalah saya melakukan visualisasi data menggunakan *scatter plot* untuk melihat relevansi numerik variabel antara prediktor(metascore , rating , runtime , votes , year) dan respons / yang ingin diprediksi(revenue). Berikut adalah Gambar 13 , 14 , 15 , 16 dan 17.



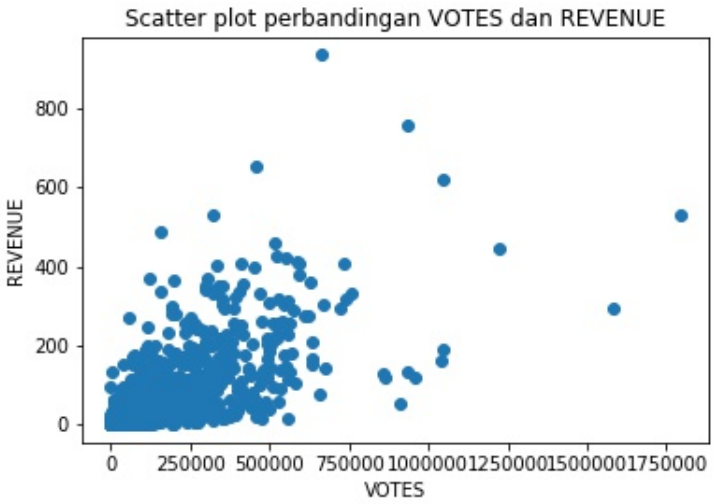
Gambar 13: *scatter plot metascore dengan revenue*



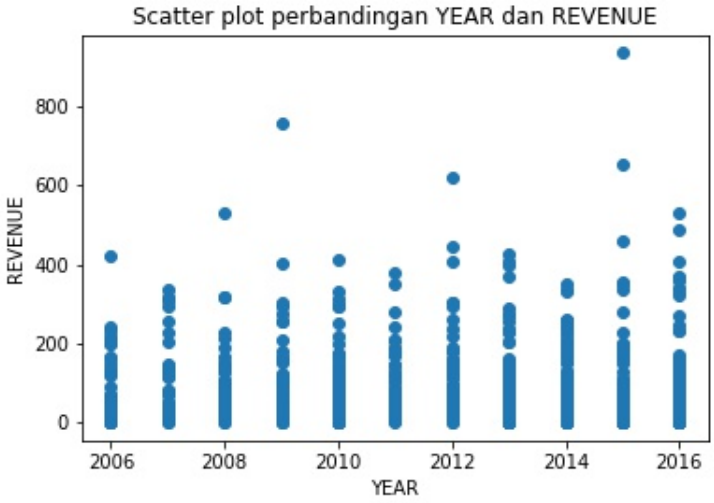
Gambar 14: *scatter plot rating dengan revenue*



Gambar 15: *scatter plot runtime dengan revenue*



Gambar 16: *scatter plot votes dengan revenue*

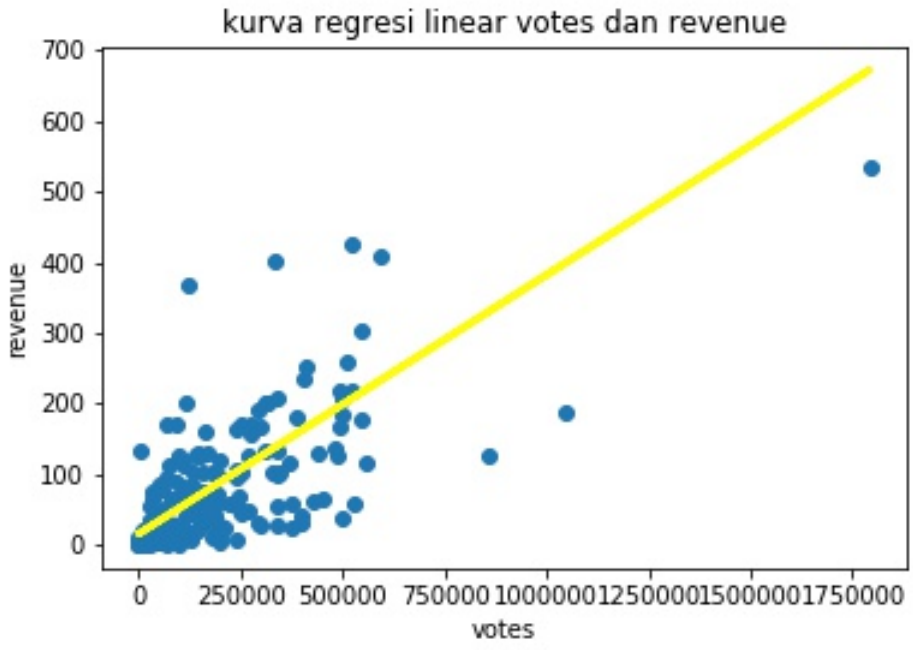


Gambar 17: *scatter plot year dengan revenue*

Untuk memilih atribut prediktor untuk memprediksi *revenue*.Atribut *votes* dipilih sebagai prediktor untuk percobaan memprediksi *revenue*. Menggunakan Persamaan *linear regression*, ditemukan koefisien dan konstanta nya menggunakan fungsi fit pada *sci kit learn* yaitu pada persamaan 25 :

$$Y = (0.00036)X + 15.09 \tag{25}$$

Berikut adalah visualisasi *linear regression* berupa kurva linear menggunakan *library matplotlib* yaitu :



Gambar 18: *Kurva linear regresi untuk memprediksi revenue*

Untuk menguji model yang dibuat, dilakukan perhitungan nilai RMSE untuk menghitung nilai *error* dari prediksi yang dibuat yaitu 70,71. R^2 dihitung untuk menguji berdasarkan presentase kebenaran prediksi yaitu sekitar 0.39 atau 39 persen. Kesimpulan yang dapat diambil adalah penggunaan fitur *votes* untuk memprediksi *revenue* memiliki akurasi yang belum terlalu tinggi. Dengan melakukan eksperimen dengan mengombinasikan atribut lain dan menggunakan algoritma *machine learning* yang lain, maka akan diuji apakah terdapat faktor-faktor yang dapat membuat film lebih sukses.

8. **Mencari sumber data yang relevan untuk melakukan pengumpulan data dari situs *review* film dan media sosial**
   Status : Ada sejak rencana kerja skripsi.
   Hasil : Untuk melakukan pengambilan data media sosial, sudah dilakukan survei untuk mencari API untuk mengakses data media sosial seperti *twitter* yang bernama python-twitter.*dataset Kaggle* juga dapat digunakan media untuk mencari data media sosial yang akan digabung dengan data *film*

6 Pencapaian Rencana Kerja

Langkah-langkah kerja yang berhasil diselesaikan dalam Skripsi 1 ini adalah sebagai berikut:

1. Melakukan studi literatur dengan mencari jurnal, *paper* mengenai penelitian sejenis dari berbagai sumber untuk membantu penulis dalam menulis
2. Melakukan studi literatur mengenai ilmu statistika dasar khususnya dalam memahami persebaran data
3. Melakukan studi literatur mengenai proses *data mining*
4. Melakukan studi literatur mengenai metode metode *machine learning* yaitu *clustering* dan *classification* yang relevan
5. Melakukan studi literatur mengenai teori dan implementasi visualisasi data seperti *histogram,scatter plot, box plot* untuk membantu mengetahui sifat data yang dikumpulkan menggunakan *matplotlib*

6. Melakukan penelitian sejenis mengenai industri perfilman untuk mengetahui relevansi antar faktor yang ada
7. Mempelajari bahasa pemrograman *Python* dan beberapa *library* dari *Python* seperti *Pandas,Sci-Kit learn* dan *matplotlib*
8. Mencari sumber data yang relevan untuk melakukan pengumpulan data dari situs *review* film dan media sosial

7 Kendala yang Dihadapi

Kendala - kendala yang dihadapi selama mengerjakan skripsi :

- Terlalu banyak melakukan prokratinasi
- Terlalu banyak mengambil mata kuliah lain yang tingkat kesulitan tinggi karena ada tugas besar
- Mengalami kesulitan pada saat melakukan eksperimen

Bandung, 25/11/2019

Teuku Hashrul

Menyetujui,

Nama: Kristopher David Harjono
Pembimbing Tunggal