MINI PROJECT
SAINS DATA GENOM

Ditulis Oleh:

Tulus Setiawan 2006568802

Disusun sebagai pemenuhan nilai UAS mata kuliah Sains Data Genom
Dosen Pengampu: Prof. Setia Pramana, Ph.D.

DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS INDONESIA
2023

BAB I
PENDAHULUAN

Dalam sejumlah riset, informasi dipresentasikan oleh sekelompok entitas tanpa label. Langkah awalnya adalah menganalisis keterhubungan antara entitas-entitas tersebut dan mengelompokkannya menjadi cluster-cluster homogen, di mana entitas dalam satu cluster memiliki kesamaan sementara entitas dari cluster yang berbeda menunjukkan perbedaan. Dengan kata lain, perlu minimalisasi jarak intra-cluster dan maksimalisasi jarak antar-cluster. Pada umumnya, analisis klaster dapat dianggap sebagai bagian dari *unsupervised learning* yang menangani data dengan struktur tersembunyi, yang perlu diungkap dari kumpulan data tanpa label. Untuk mengungkapkan informasi tersebut dan mengelompokkan variabel, analisis klaster melibatkan berbagai metode dan teknik. Dalam panduan singkat ini, penulis akan menjelaskan teknik-teknik utama dari analisis klaster beserta kelebihan dan kekurangannya.

Sebaliknya, penerapan metode lasso dalam model klasifikasi pada dataset genomik terbukti sangat bermanfaat karena dapat mengatasi masalah regresi *pitfalls*. Metode LASSO (Least Absolute Shrinkage and Selection Operator), yang diperkenalkan oleh Robert Tibshirani pada tahun 1996, menjadi alat yang efektif untuk regularisasi dan seleksi fitur. Dengan membatasi nilai absolut parameter model, LASSO melakukan penyusutan dengan mengenakan penalti pada beberapa koefisien regresi hingga menjadi nol. Variabel yang masih mempertahankan koefisien tidak nol setelah proses ini dipilih untuk dimasukkan dalam model, memberikan kontribusi pada pengurangan kesalahan prediksi. Keuntungan dari penerapan LASSO melibatkan peningkatan akurasi prediksi, terutama bermanfaat dalam konteks dataset dengan jumlah observasi yang terbatas dan sejumlah fitur yang besar. Selain itu, LASSO meningkatkan interpretasi model dengan menghilangkan variabel yang tidak relevan, sehingga mengurangi risiko *overfitting*.

Penelitian ini juga melibatkan analisis data dari dataset "*Brain cancer gene expression - CuMiDa*". Dataset terdiri dari 130 sampel, masing-masing sampel memiliki 54676 gen, dan masing-masing sampel terdiri atas 5 kelas tipe sampel. Kemudian, dataset tersebut akan dilakukan regresi logistik LASSO dengan tujuan untuk *feature selection*, selanjutnya akan dilakukan klasifikasi berdasarkan fitur-fitur yang telah dipilih sebelumnya menggunakan metode *logistic regression* dan XGBoost sebagai perbandingan..

BAB II
METODE ANALISIS

2.1 *Gene Filtering*

Metode *gene filtering* adalah suatu teknik yang digunakan untuk menyederhanakan dan membersihkan data ekspresi gen dengan cara menghilangkan gen-gen yang tidak memberikan informasi atau tidak relevan untuk analisis. Tujuan utama dari metode ini adalah mengurangi gangguan dan kerumitan data ekspresi gen, sehingga memungkinkan fokus pada gen-gen yang paling penting. Dengan mengimplementasikan metode *gene filtering*, kita dapat meningkatkan kekuatan dan ketepatan dalam mengidentifikasi gen yang diekspresikan secara berbeda, mengelompokkan jenis sel, serta mengidentifikasi biomarker dan target terapeutik yang potensial. Terdapat berbagai pendekatan dalam metode *gene filtering*, termasuk penggunaan ambang batas tetap atau adaptif berdasarkan rata-rata atau varians ekspresi gen, memanfaatkan analisis komponen utama, atau menggunakan algoritma pengoptimalan. Metode ini dapat diterapkan pada berbagai jenis data ekspresi gen, seperti microarray, RT-PCR, atau RNA-seq, sehingga memberikan fleksibilitas dalam analisis genetik.

2.2 *Differential Gene Expression*

Ekspresi gen diferensial adalah suatu teknik yang bertujuan untuk mengidentifikasi gen-gen yang menunjukkan perubahan tingkat ekspresi antara kondisi atau kelompok yang berbeda, seperti sehat versus sakit, diobati versus tidak diobati, atau tahap perkembangan yang berbeda. Metode ini sangat berguna dalam mengungkapkan perbedaan dalam respons genetik terhadap berbagai kondisi atau perlakuan. Dengan menganalisis ekspresi diferensial, kita dapat menemukan gen-gen yang mungkin terlibat dalam mekanisme biologis khusus atau jalur yang berperan dalam variasi fenotipik atau respons terhadap rangsangan tertentu. Selain itu, metode ekspresi diferensial memainkan peran penting dalam mengidentifikasi biomarker potensial atau target terapeutik untuk penyakit tertentu.

Dalam menemukan perbedaan ekspresi gen, dapat digunakan metode t-test dan analisis Limma (Linear Models for Microarray Data). T-test adalah metode statistik yang membandingkan rata-rata dua kelompok sampel yang independen, memberikan nilai p sebagai indikator signifikansi statistik perbedaan tersebut. Sementara itu,

metode Limma menggunakan model linear untuk memperkirakan perbedaan ekspresi gen antara kelompok perlakuan. Metode ini memiliki keunggulan, seperti kemampuannya menangani variasi heterogen dalam data, uji hipotesis kompleks, dan kontrol tingkat kesalahan tipe I melalui koreksi *multiple testing*. Dengan demikian, ekspresi gen diferensial dengan Limma dapat memberikan informasi yang lebih akurat dan komprehensif dalam mengidentifikasi perubahan ekspresi gen yang signifikan.

2.3 *Clustering*

Clustering merupakan suatu metode yang membantu mengelompokkan titik data yang serupa. Salah satu tipe *clustering* yang sering digunakan adalah *hierarchical clustering* dan/atau metode partisis. Ide dari metode pengelompokan ini adalah bahwa metode ini dapat membantu untuk menafsirkan data berdimensi tinggi.

a. *Hierarchical Clustering*

Pengelompokkan hirarki digunakan untuk mencoba dan menemukan beberapa struktur dalam tren ekspresi gen, lalu mempartisi gen ke dalam kelompok yang berbeda. Terdapat dua langkah dalam prosedur pengelompokan ini:

1) Hitung metrik "jarak" antara setiap pasangan gen
2) Pengelompokkan gen secara hirarki menggunakan metode aglomerasi tertentu

Terdapat banyak cara yang dapat dilakukan untuk menjalani kedua langkah diatas, seperti *simple euclidean distance metric* untuk menghitung jarak dan *single linkage* / *complete linkage* untuk mengelompokkan gen.

Hasil dari proses pengelompokkan hirarki dapat dinyatakan dalam berbagai grafik atau plot. Penggunaan dendrogram dapat berfungsi untuk menentukan secara visual berapa banyak kelompok yang layak untuk dijadikan fokus. Akhirnya, tren dari ekspresi gen dapat divisualisasikan setelah proses pengelompokkan gen ke masing-masing *cluster* yang sesuai menggunakan *heatmap*.

b. Partitioning Methods

Selain pengelompokkan hirarki, terdapat metode clustering lainnya, yaitu metode partisi. Dalam metode partisi, sebelum proses pengelompokan, ditentukan terlebih dahulu suatu angka tetap k yang mewakili jumlah dari *cluster*. Sampel kemudian ditugaskan ke *cluster* di mana mereka paling cocok. Hal ini lebih fleksibel daripada pengelompokan hirarki, tetapi juga memiliki masalah dalam pemilihan nilai k.

Selanjutnya, untuk mengukur seberapa baik suatu objek ditempatkan dalam *cluster*, dinilai menggunakan *silhouette score*. *Silhouette score* merupakan metrik evaluasi yang mengukur keakuratan pengelompokkan berdasarkan kedekatan suatu objek dalam *cluster* yang sama dan seberapa jauh objek tersebut dari *cluster* lainnya. Nilai silhouette score berkisar antara -1 hingga 1, di mana nilai dekat dengan 1 menunjukkan bahwa objek tersebut ditempatkan dengan baik dalam *cluster* dan memiliki jarak yang baik dari *cluster* lain. Sedangkan, nilai dekat dengan -1 menunjukkan bahwa objek tersebut mungkin ditempatkan dengan buruk dalam *cluster* dan memiliki kedekatan yang lebih baik dengan *cluster* lain. Untuk nilai sekitar 0, hal ini menunjukkan bahwa objek berada di dekat batas antara dua *cluster*.

2.4 *LASSO Regression*

Feature selection adalah proses penting dalam pemodelan yang melibatkan pemilihan subset variabel penjelas untuk menjelaskan variabel respons. Tujuan utamanya termasuk meningkatkan interpretabilitas model dengan menghilangkan variabel yang tidak diperlukan dan tidak memberikan informasi, mengurangi ukuran masalah untuk pemrosesan algoritma yang lebih cepat, dan mengurangi overfitting. Pada dataset dengan dimensi tinggi di mana jumlah fitur melebihi jumlah observasi, pemilihan variabel menjadi lebih penting.

Metode LASSO (Least Absolute Shrinkage and Selection Operator), diperkenalkan oleh Robert Tibshirani pada tahun 1996, menjadi alat yang kuat untuk regularisasi dan *feature selection*. Dengan memberlakukan batasan pada jumlah nilai absolut parameter model, LASSO melakukan proses penyusutan yang melibatkan penalisasi beberapa koefisien regresi menjadi nol. Variabel yang masih memiliki koefisien yang tidak nol setelah proses ini dipilih untuk model, berkontribusi pada

pengurangan kesalahan prediksi. LASSO memberikan keuntungan seperti peningkatan akurasi prediksi, terutama bermanfaat dengan jumlah observasi yang sedikit dan sejumlah fitur besar. Selain itu, LASSO meningkatkan interpretabilitas model dengan menghilangkan variabel yang tidak relevan, sehingga mengurangi *overfitting*.

2.5 *Logistic Regression*

Regresi logistik adalah model klasifikasi statistik yang mengukur hubungan antara variabel dependen kategorikal (hanya memiliki dua kategori) dan satu atau lebih variabel independen, yang biasanya bersifat kontinu, dengan menggunakan skor probabilitas sebagai nilai yang diprediksi dari variabel tergantung. Regresi logistik tidak mengasumsikan hubungan linear antara variabel tergantung dan independen. Variabel independen tidak perlu berdistribusi normal, atau berkaitan linear, atau memiliki varians yang sama di setiap kelompok. Regresi logistik dikembangkan untuk mengklasifikasikan hasil biner berdasarkan variabel independen yang bersifat kategorikal atau kontinu.

Pada regresi logistik, probabilitas berkisar antara 0 dan 1, dengan 0 diberikan untuk kejadian yang tidak terjadi dan 1 diberikan untuk kejadian yang terjadi. Berdasarkan probabilitas yang diestimasi untuk setiap sampel, yang akan berada antara 0 dan 1, dan batas yang ditentukan untuk model, regresi logistik mengklasifikasikan sampel ke dalam kelompok yang berbeda. Batas probabilitas untuk klasifikasi bergantung pada tujuan. Sebagai contoh, akan dikembangkan model regresi logistik untuk menentukan apakah seseorang memiliki kanker atau tidak. Untuk tujuan ini, maka ditentukan batas bawah probabilitas, katakanlah 0,4, untuk model. Dengan cara ini, model kemungkinan kecil melewatkan kasus kanker dengan mengklasifikasikan setiap pasien dengan probabilitas menjadi bagian dari kelompok kanker, yaitu mengkategorikan probabilitas lebih besar dari 0,4 sebagai memiliki kanker.

2.6 *XGBoost*

XGBoost, atau eXtreme Gradient Boosting, merupakan sebuah algoritma ensemble learning yang diperkenalkan pada tahun 2014 oleh Tianqi Chen dan telah menjadi sangat populer dalam penyelesaian masalah klasifikasi dan regresi. XGBoost

didasarkan pada kerangka kerja gradient tree boosting yang bersifat scalable, portable, dan dapat didistribusikan. Algoritma ini mengadopsi pendekatan boosting, di mana setiap model yang dibangun diperkuat secara iteratif dengan memperbaiki kesalahan prediksi model sebelumnya. Pada setiap iterasi, XGBoost memfokuskan pembelajarannya pada residu atau selisih antara kelas aktual dan prediksi model saat ini, menggunakan pohon keputusan sebagai weak learner.

Chen & Guestrin (2016) menyoroti salah satu kelebihan XGBoost, yaitu kemampuannya untuk mengoptimalkan fungsi kerugian (loss function) dan regularisasi. Fungsi kerugian umum yang digunakan dalam klasifikasi adalah log-loss atau cross-entropy loss, yang membantu mengukur kesalahan prediksi probabilitas kelas. Sementara itu, regularisasi digunakan untuk mencegah overfitting dan meningkatkan generalisasi model.

Chen & Guestrin (2016) juga menekankan performa unggul XGBoost dalam menangani masalah klasifikasi, didukung oleh berbagai studi empiris dan partisipasi dalam kompetisi data terbuka. Kelebihan algoritma ini mencakup skalabilitasnya yang tinggi, memungkinkan pemrosesan data berukuran besar dengan efisiensi yang tinggi. XGBoost juga menunjukkan ketahanan yang baik terhadap outliers, serta kemampuan yang baik dalam menangani fitur campuran, baik yang bersifat numerik maupun kategorik, serta menangani missing values.

BAB III
HASIL DAN PEMBAHASAN

3.1 Dataset

Dataset yang akan digunakan berjudul "*Brain cancer gene expression* - CuMiDa", GSE50161 *microarray experiment*. Data tersebut merupakan data publik yang tersedia di Kaggle. Data terdiri dari 130 sampel, 54676 gen (features), dan 5 kelas. Dari total keseluruhan data, data yang akan digunakan hanya sebanyak 50% sampel dengan cara diambil secara acak.

3.2 Normalisasi

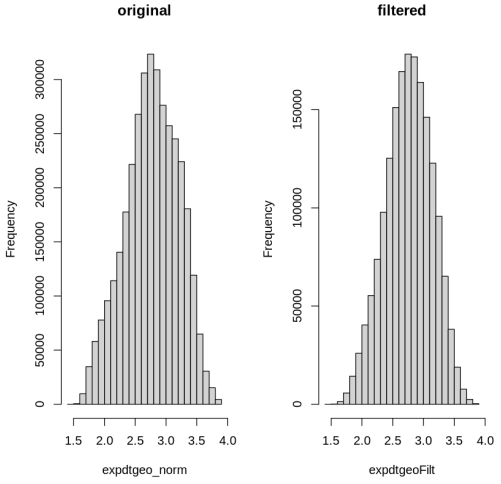
Setelah data di-*input*, kemudian dilakukan proses normalisasi data menggunakan penyetaraan log2. Tujuan dilakukan normalisasi data adalah untuk komponen tahap awal proses data untuk membuat data dapat dibandingkan dalam rentang yang sama. Menggunakan normalisasi data, didapat data sesudah dan sebelum normalisasi memiliki nilai minimum dan maksimum terlampir dalam output berikut.

```
Nilai minimal-maksimal data sebelum normalisasi
[1]  2.825938 14.849308
```

```
Nilai minimal-maksimal data setelah normalisasi
[1] 1.498730 3.892324
```

3.3 *Gene Filtering*

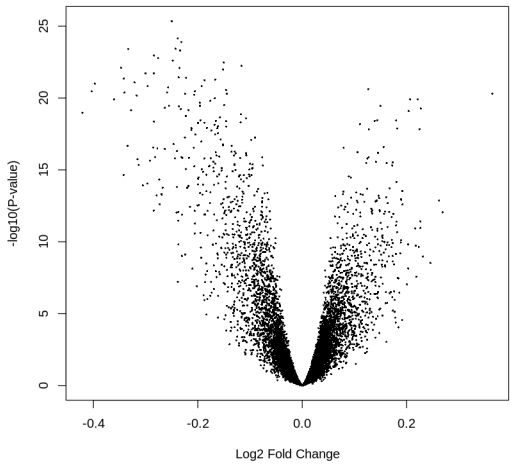
Pada tahap ini, dilakukan proses *filtering* terhadap gen atau *feature* yang dengan cara menghilangkan data-data yang duplikat dan juga dengan menggunakan metode nilai *interquartile range* (IQR) untuk proses *filtering*. Hasil dari proses sebelum dan sesudah *gene filtering* dapat dilihat pada gambar di bawah ini.



Hasil data yang telah mengalami filtering menunjukkan bahwa distribusinya yang semula tidak normal, kemudian menjadi berdistribusi normal. Selain itu, jumlah data yang terfilter menjadi lebih sedikit, yang merupakan akibat dari penghapusan duplikat gen.

3.5 *Differential Expression Gene*

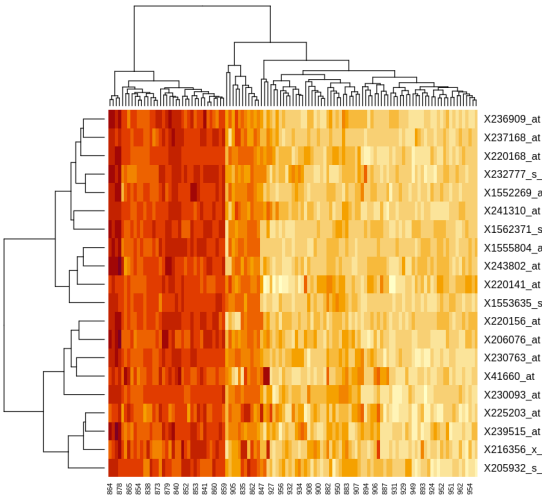
Pada tahap ini, akan dilakukan analisis data genetik dengan fokus pada identifikasi gen yang menunjukkan perbedaan ekspresi antara empat kelompok sub tipe kanker, yaitu kelompok ependymoma, glioblastoma, medulloblastoma, dan pilocytic_astrocytoma. Proses analisis ini menggunakan teknik library Limma. Limma adalah suatu modifikasi model linear yang digunakan khususnya untuk menganalisis ekspresi gen atau protein. Library Limma digunakan untuk memperoleh informasi statistik seperti F-statistic, p-value, dan koefisien yang terkait dengan masing-masing gen. Proses dimulai dengan pembuatan model desain linear yang melibatkan variabel kelompok yang ingin dibandingkan, yaitu yang bersubtipe kanker saja. Hasilnya, didapatkan koefisien untuk setiap gen, yang mencerminkan perbedaan ekspresi antarsubtipe. Informasi tambahan, seperti p-value dan F-statistic, memberikan dasar statistik untuk menentukan signifikansi perbedaan tersebut. Hasil dari analisis Limma dapat divisualisasikan dalam bentuk volcano plot berikut.



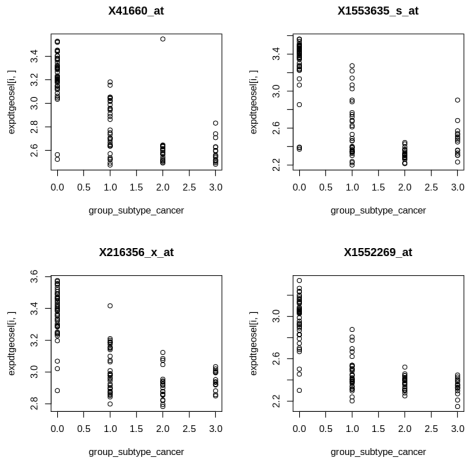
Dengan menggunakan koefisien 2, volcano plot di atas merepresentasikan bahwa kebanyakan dari selisih ekspresi gen dari keempat kelompok sub tipe kanker cenderung tidak terlalu besar dibuktikan dengan sebaran titik dari volcano plotnya berada di interval antara -0.2 hingga 0.4 meskipun p-value yang dihasilkan sudah

signifikan. Hasil tersebut dapat kita abaikan karena bukan merupakan bagian yang penting.

Selain itu, berikut ini merupakan interpretasi dari 20 gen teratas.



Penggunaan heatmap membantu memvisualisasikan pola ekspresi dari 20 gen teratas. Heatmap berikut menggambarkan expresi dari setiap gen (baris) dan setiap sampel (kolom). Warna merah menandakan gen tersebut memiliki ekspresi lebih tinggi (over expression). Terlihat pengelompokkan yang jelas antara sampel dari keempat grup, yaitu ependymoma, glioblastoma, medulloblastoma, dan pilocytic_astrocytoma.

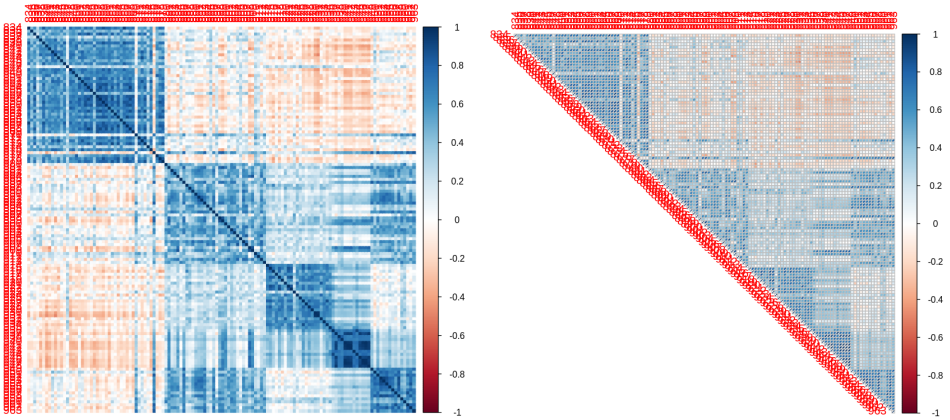


Boxplot untuk empat gen teratas lebih lanjut memperjelas perbedaan ekspresi di antara kelompok-kelompok sub tipe kanker. Terlihat bahwa gen-gen tersebut memiliki ekspresi yang sangat berbeda diantara keempat grup sub tipe kanker tersebut.

3.6 *Clustering*

Pada tahap ini, sebelum dilakukan proses *clustering* maka dilakukan pemilihan data terlebih dahulu karena jumlah gen dari hasil filtering masih terlalu banyak, sehingga akan dipilih 100 gen dengan variabilitas tinggi untuk mempermudah proses clustering.

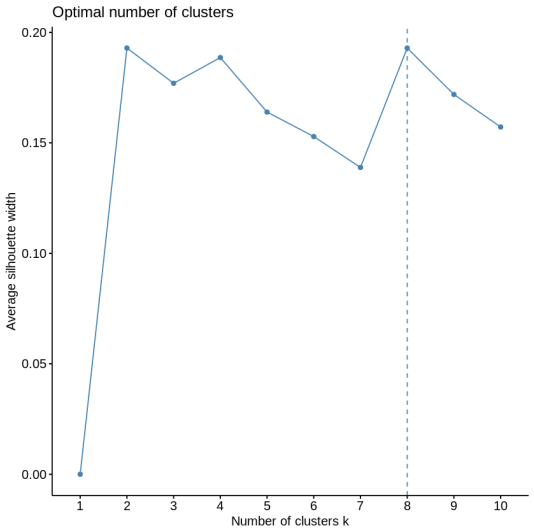
Berikut merupakan visualisasi *corelation plot* dari 100 gen yang terpilih dengan metode .



Gambar dapat diinterpretasikan bahwa semakin biru nilainya atau semakin menuju angka 1, maka korelasi antar gen semakin kuat positif. Jika warnanya semakin putih atau menuju 0 maka korelasi antar gen semakin lemah atau tidak berkorelasi. Kemudian, jika warnanya semakin merah atau menuju -1 maka korelasi antar gen semakin kuat negatif.

3.6.1 *K-means Clustering*

Selanjutnya, data dari gen gen yang sudah terpilih akan dilakukan *rescalling*. Setelah itu, akan ditentukan jumlah *k cluster* dengan metode *silhouette score* untuk mendapatkan jumlah *k cluster* yang optimal. Didapatkan jumlah *k cluster* yang optimal untuk *k-means clustering* pada data ini adalah sebanyak 8, seperti tertera pada gambar berikut.



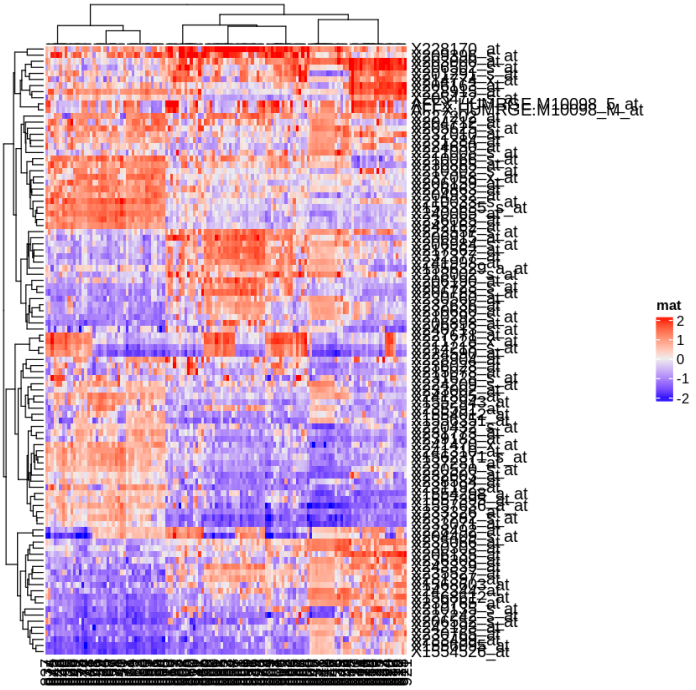
Dengan menggunakan *k cluster* sebanyak 8, didapatkan hasil *k-means clustering* seperti berikut ini

```
X204409_s_at:        8 X224590_at:        3 X1557636_a_at:     1 AFFX.HUMRGE.M10098_5_at:   2 X206552_s_at:      6 X214218_s_at:      3 X1568612_at:       6 X237282_s_at:      1
X240065_at:          5 X206915_at:        4 X1556351_at:       1 X242344_at:        8 X231192_at:        1 AFFX.HUMRGE.M10098_M_at:   2 X206785_s_at:      4 X243339_at:        4 X227242_s_at:
      6 X241470_x_at:      1 X221728_x_at:      3 X204712_at:        5 X210066_s_at:      5 X227202_at:        7 X235591_at:        4 X236085_at:        5 X236308_at:        8 X232377_at:        4
X1554298_a_at:       1 X210135_s_at:      6 X206135_at:        2 X1557395_at:       1 X224209_s_at:      5 X221008_s_at:      8 X227671_at:        3 X210033_s_at:      5 X231397_at:        4
X206502_s_at:        2 X231628_s_at:      8 X230865_at:        5 X230560_at:        4 X241805_at:        5 X241998_at:        4 X1554526_at:       6 X1552943_at:       1 X228492_at:        8
X224650_at:          8 X237281_at:        1 X235527_at:        4 X239765_at:        6 X241310_at:        1 X206190_at:        4 X1553635_s_at:     5 X240713_s_at:      4 X206898_at:        4
X1568603_at:         8 X205625_s_at:      5 X1556095_at:       6 X242162_at:        5 X236028_at:        1 X231384_at:        5 X230765_at:        6 X238584_at:        1 X233499_at:        6 X1554012_at:
      1 X232010_at:        5 X204932_at:        5 X206163_at:        2 X203000_at:        2 X228170_at:        7 X232636_at:        4 X235066_at:        8 X232603_at:        1 X231773_at:        1
X206984_s_at:        4 X209396_s_at:      7 X1562371_s_at:     1 X220520_s_at:      1 X214774_x_at:      2 X233002_at:        5 X233071_at:        1 X217562_at:        4 X218002_s_at:      8
X233326_at:          1 X220432_s_at:      1 X239183_at:        1 X237058_x_at:      5 X230303_at:        8 X228915_at:        2 X226863_at:        5 X1556329_a_at:     4 X210302_s_at:      5
X228904_at:          5 X205472_s_at:      2 X242193_at:        6 X207723_s_at:      4 X230680_at:        4 X215078_at:        1 X223557_s_at:      4 X201291_s_at:      2 X206159_at:        5
X210292_s_at:        4
```

Dengan *size cluster* sebagai berikut,

```
22 · 10 · 4 · 20 · 20 · 10 · 3 · 11
```

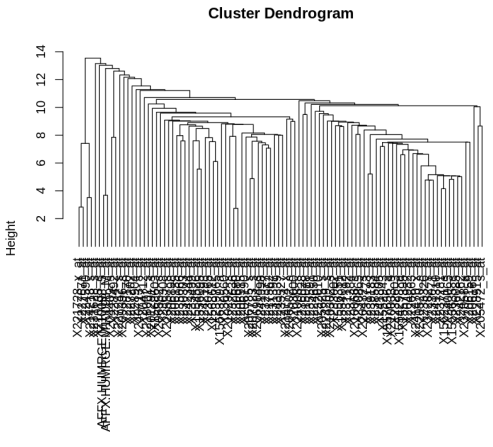
Dengan menggunakan *library ComplexHeatmap,* didapatkan *output* sebagai berikut,



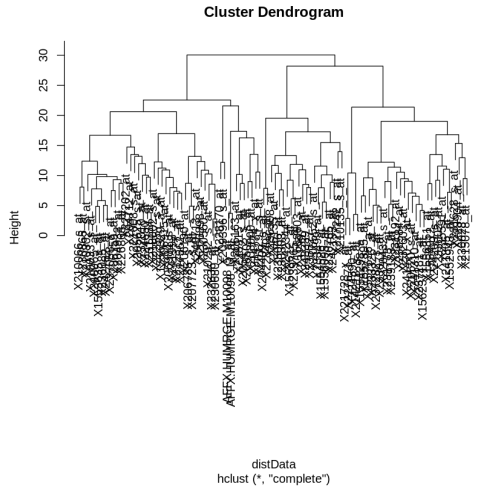
3.6.2 *Hierarchical Clustering*

Berikut ini merupakan hasil *hierarchical clustering* dengan berbagai metode untuk $k = 8$:

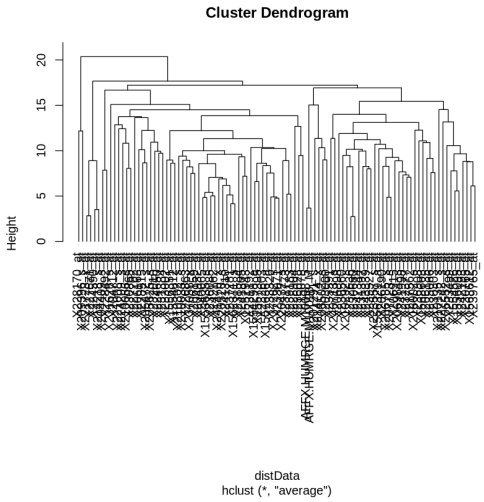
- *Dendogram Clustering* dengan metode *single*:



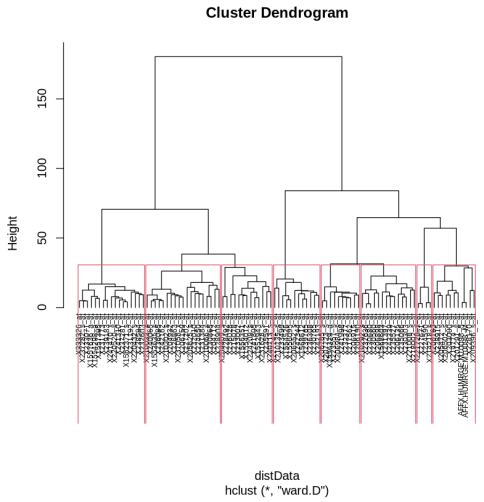
- *Dendogram Clustering* dengan metode *complete*:



- *Dendogram Clustering* dengan metode *average*:



- *Dendogram Clustering* dengan metode *ward.D*:



3.7 *Classification*

3.7.1 *Dataset*

Dataset bersumber dari website Kaggle, dengan judul "Brain cancer gene expression - CuMiDa". Dataset terdiri dari 130 sampel, masing-masing sampel memiliki 54676 gen, dan masing-masing sampel terdiri atas 5 kelas tipe sampel. Dalam proses klasifikasi, akan digunakan sebanyak 50% sampel dari total keseluruhan data dan hanya 2 kelas tipe sampel yang digunakan, yaitu "ependymoma" dan "glioblastoma". Kemudian data di normalisasi dan filtering seperti yang sudah dijelaskan pada Subbab 3.2 dan Subbab 3.3

3.7.2 *Feature Selection* menggunakan *Lasso Regression*

Menggunakan cross-validation sebanyak 5, parameter lambda yang akan dicari nilai optimalnya terdiri atas 1000 parameter kandidat, yang dibangun dari persamaan berikut.

$$\lambda_i = \sum_{n=0}^{1000} 10^{-2+\frac{n}{1000}}$$

Hasil dari parameter tuning dengan metode grid search adalah parameter lambda yang teroptimal, yaitu sebagai berikut.

A data.frame: 1 × 2

	alpha	lambda
	<dbl>	<dbl>
205	1	0.06558686

Kemudian, dengan melakukan *fitting data training* ke model yang menggunakan parameter lambda teroptimal, didapat fitur-fitur dengan dugaan sebagai fitur yang berpengaruh terhadap proses klasifikasi adalah sebagai berikut.

A data.frame: 13 × 3

Row	Column	Value
<chr>	<chr>	<dbl>
(Intercept)	s1	11.1138727355
X239500_at	s1	-0.4879627649
X208212_s_at	s1	1.7141671230
X205161_s_at	s1	-0.1634057811
X213497_at	s1	-0.0004986593
X202409_at	s1	-1.1416992651
X230130_at	s1	-0.1983158703
X204081_at	s1	0.6580184276
X241310_at	s1	-0.8110366199
X228850_s_at	s1	-0.8576696368
X213150_at	s1	0.0396541599
X204933_s_at	s1	-2.4494683153
X213540_at	s1	-0.0565528653

3.7.3 Model klasifikasi dengan *Logistic Regression*

Kemudian, setelah data training telah melalui proses *feature selection*, maka dilakukan *fitting* model *logistic regression*, sehingga didapat parameter-parameter berikut.

```
13 x 1 sparse Matrix of class "dgCMatrix"
                      s0
(Intercept)   33.428417
X239500_at     2.838177
X208212_s_at  33.601641
X205161_s_at  -9.381943
X213497_at    -6.602530
X202409_at    -4.880916
X230130_at    -8.396394
X204081_at    12.068227
X241310_at    -1.430402
X228850_s_at  -5.782826
X213150_at     1.377602
X204933_s_at -12.563109
X213540_at   -10.826164
```

Kemudian,model diuji pada data test sehingga didapat prediksi dari data test sebagai berikut.

A matrix: 1 × 16 of type dbl

	835	838	840	850	866	882	888	889	892	897	899	900	904	910	912	913
s0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1

Dengan nilai aktual dari data test sebagai berikut.

```
0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1
▶ Levels:
```

Berdasarkan hasil prediksi di atas, model tepat memprediksi data test dengan akurasi 0.9375 atau 93.75%.

3.7.4 Model klasifikasi dengan XGBoost

Selanjutnya, akan dicoba metode klasifikasi lain untuk dibandingkan dengan metode *logistic regression* berdasarkan akurasinya. Model klasifikasi yang digunakan untuk pembanding, yaitu *XGBoost Classifier*. Menggunakan data yang sudah dilakukan *feature selection* untuk dipisah dan dijadikan data *train* dan data *test.*

Setelah itu akan didefinisikan terlebih dahulu parameter-parameter untuk model XGBoost, seperti berikut ini.

```
params <- list(
  objective = "binary:logistic", # binaryclass classification
  max_depth = 3, # maximum depth of each tree
  subsample = 0.8, # subsample ratio of the training data
  colsample_bytree = 0.8 # subsample ratio of the features
)
```

Barulah kemudian dilakukan *fitting model* untuk melakukan training model pada data train dengan *nrounds* sebanyak 100. Setelah itu, model diuji pada data test sehingga didapat prediksi dari data test sebagai berikut.

Dengan nilai aktual dari data test sebagai berikut.

Berdasarkan hasil prediksi di atas, model tepat memprediksi data test dengan akurasi 0.9375 atau 93.75%.

BAB IV
KESIMPULAN

- Dari data sample yang digunakan, data yang akan diproses untuk *clustering*, yaitu 100 gen yang memiliki variabilitas tertinggi
- Untuk analisis *clustering,* terdapat 8 *cluster* yang merepresentasikan gen gen tersebut setelah dilakukan *silhouette score* untuk menentukan jumlah *cluster* yang optimal.
- Terdapat beberapa output *clustering* yang merepresentasikan kelompok dari gen-gen tersebut dengan *k-means clustering* dan *hierarchical clustering*.
- Data yang telah dilakukan proses normalisasi dan filtering akan dilakukan proses *feature selection* dan *hyperparameter tuning* menggunakan *grid search* untuk menemukan fitur terbaik dan parameter yang optimal
- Dilakukan klasifikasi dengan metode *logistic regression* dan *xgboost*, dari kedua metode didapatkan akurasi yang sama, yaitu 0.9375.
- Sehingga kedua metode tersebut bisa sama sama digunakan karena memiliki kinerja yang sama

DAFTAR PUSTAKA

Fonti, V., & Belitser, E. (2017). Feature selection using lasso. *VU Amsterdam research paper in business analytics*, 30, 1-25.

Rahnenfuhrer, J., & Markowetz, F. (2004). Exploratory Data Analysis: Clustering gene expression data. http://compdiag.molgen.mpg.de/ngfn/pma2004nov.shtml

Rezaei, N., & Jabbari, P. (2022). *Immunoinformatics of Cancers: Practical Machine Learning Approaches Using R*. Academic Press.

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In Proceedings of the 655 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785- 656 794).

LAMPIRAN

Berikut adalah lampiran kode R yang digunakan, lengkap dengan penjelasannya.

- Project UAS SD Genom_Tulus Setiawan_2006568802