# **4. Modelling**

Tahap keempat pada metodologi CRISP-DM untuk melakukan prediksi kinerja karyawan adalah *modeling*. Pada bab ini akan dijelaskan mengenai pemilihan teknik modelling, menghasilkan *test design*, membangun model, dan menilai model yang telah dibangun.

## **4.1 *Select Modeling Technique***

Teknik pemodelan yang digunakan didorong oleh tujuan penambangan data yang ingin dicapai oleh perusahaan. Secara sederhana, hal tersebut melibatkan penerapan *association rule* untuk mendapatkan keterkaitan antar variabel dalam menentukan kinerja karyawan dalam suatu perusahaan. Data karyawan dalam kurun waktu tertentu akan ditambahkan untuk mendapatkan hasil yang lebih baik. Penerapan *association rule* dengan algoritma apriori cocok digunakan untuk jenis pemodelan tersebut. Algoritma Apriori akan menggenerate *association rule* dengan menemukan *frequent item set* yang memenuhi minimum *support* (persentase kombinasi item dalam *dataset*) dan minimum *confidence* (keterkaitan antar item) yang diinisialisasi oleh penambang data untuk menentukan faktor-faktor yang berpengaruh dalam menentukan kinerja karyawan.

Dalam menggenerate *association rule*, algoritma apriori tidak memerlukan data untuk di-*split* ke dalam *test sets* dan *training sets*. Oleh karena itu, penambangan data hanya dilakukan dengan sebuah *dataset* yang memiliki data yang cukup untuk memperoleh hasil yang dapat dipercaya terhadap model yang digunakan. Hal ini berarti, *dataset* menggambarkan data secara keseluruhan dengan baik sehingga algoritma apriori nantinya akan dapat menghasilkan *rule* yang sesuai dalam menentukan kinerja karyawan.

Dalam teknik pemodelan dengan *association rule* memerlukan asumsi spesifik terhadap data, yaitu semua atribut memiliki distribusi yang sama, tidak ada *missing value*. Untuk atribut yang tidak *categorical* (*nominal*) maka akan dilakukan pembuatan bin terlebih dahulu sebelum dilakukan penerapan algoritma apriori pada data tersebut.

## **4.2 *Generate Test Design***

Sebelum melakukan pembangunan model, perlu dilakukan perancangan terhadap bagaimana model akan diuji. Terdapat 2 cara untuk menghasilkan *test design* yang komprehensif:

* Menggambarkan kriteria “*goodness*” dari sebuah model
* Mendefinisikan data dengan kriteria yang akan diuji.

“goodness” algoritma apriori dapat diukur dengan menggunakan *interestingness measure* (ukuran ketertarikan) yang didapatkan dari hasil pengolahan data dengan perhitungan tertentu. Beberapa ukuran tersebut adalah:

1. *Support*

*Support* adalah nilai penunjang atau persentase kombinasi sebuah item dalam *dataset*.

1. *Confidence*

*Confidence* adalah nilai kepastian yaitu kuatnya hubungan antar-item dalam *association rule*. *Confidence* dapat dicari setelah pola frekuensi munculnya sebuah item ditemukan.

1. *Lift*

*Lift* adalah nilai yang mengukur besarnya hubungan antara *antecedent* dan *consequent* yang tidak saling bergantung (*independent*). *Lift* memiliki range mulai dari 0 sampai dengan ∞. Nilai yang mendekati 1 mengindikasikan bahwa *antecedent* dan *consequent* tidak memiliki ketergantungan. Nilai yang jauh dari 1 mengindikasikan bahwa *antecedent* menyediakan informasi tentang *consequent*.

1. *Conviction*

*Conviction* adalah nilai yang mengukur tingkat implikasi dari suatu aturan. *Conviction* sangat memperhatikan arah dari suatu *association rule*. *Conviction* mengindikasikan bahwa *conviction*(A→B) ≠ *conviction*(B→A)

1. *Leverage*

*Leverage* adalah nilai yang mengukur banyaknya item *antecedent* dan *consequence* yang dijual secara bersamaan dalam suatu *dataset* yang lebih dari yang diharapkan. Nilai 0 menunjukkan *antecedent* dan *consequent independent*. *Leverage* memiliki jangkauan nilai dari -0,25 sampai dengan 0,25.

Kriteria model yang dinilai bergantung pada *data mining goals* pada model yang akan dibangun. Tidak ada cara objektif untuk menilai model sampai disajikan pada HRD secara langsung. Namun HRD memerlukan aturan yang menghasilkan prediksi terhadap kinerja karyawan. Berikut adalah *test design* yang dibuat untuk model yang dibangun.

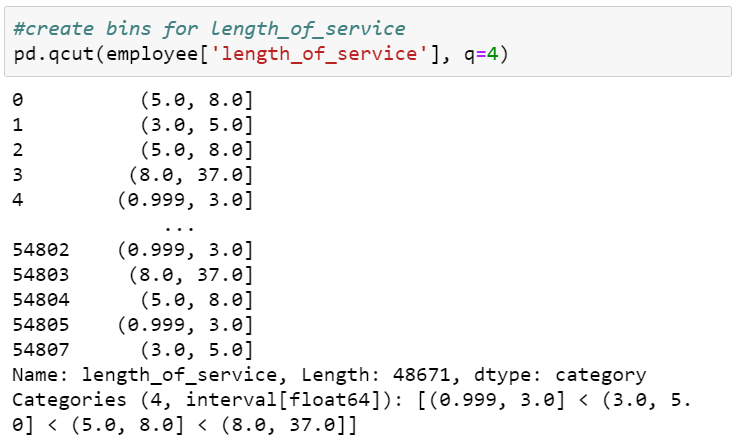
1. Melakukan pengukuran terhadap ukuran ketertarikan dari *association rule* yang telah ditemukan.
2. Menentukan keakuratan *rule* yang dihasilkan berdasarkan ukuran ketertarikan yang diperoleh dengan menyesuaikan terhadap nilai parameter yang dinyatakan berhasil. (Misalnya, *minimum confidence* = 0.5, *lift* 1 yang menandakan bahwa *antecedents* dan *consequents* berkolerasi positif, dan lain-lain).

## **4.3 *Build Model***

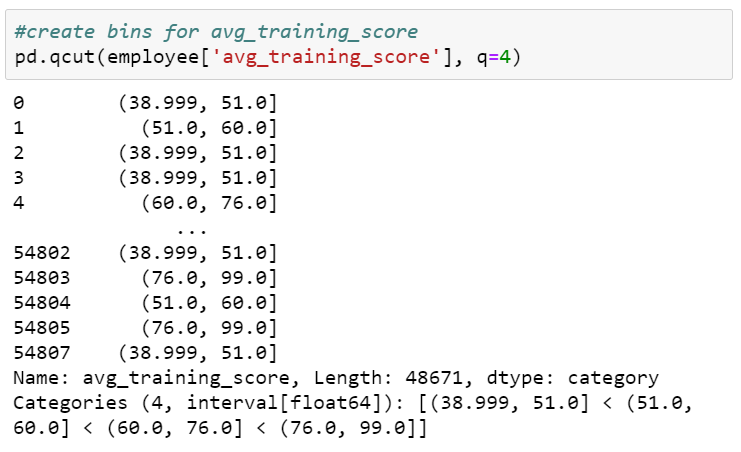
Dalam proses pembangunan model, terdapat 3 informasi yang digunakan dalam pengambilan keputusan *data mining*, yaitu:

* *Parameter settings*, merupakan parameter yang digunakan pada model yang memberikan hasil yang baik.
* Model dihasilkan.
* Deskripsi dari hasil pemodelan, termasuk *performance* dan *data issued* yang terjadi selama pengeksekusian model dan eksplorasi hasilnya.

*Association rule* dengan algoritma apriori dibangun pada bahasa pemrograman *python* dengan memanfaatkan *library python* untuk algoritma apriori, yaitu mlxtend. Mlxtend adalah *library python* yang mengimplementasikan algoritma dalam *machine learning*. Dari *library* mlxtend menggunakan *frequent patterns* untuk mengimport apriori dan *association rule* yang digunakan untuk menggenerate *association rule* dengan algoritma apriori dari *frequent itemset* yang dihasilkan. Dalam pembangunan model tersebut menggunakan *dataset* yang telah melalui tahapan *preprocessing* seperti yang dijelaskan pada asumsi di bagian 4.1 sebelumnya. Dimana atribut dengan tipe bukan kategorikal akan dibuat ke dalam beberapa bin. Seperti variabel *length\_of\_service, avg\_training\_score* akan dibagi ke dalam 4 bin.

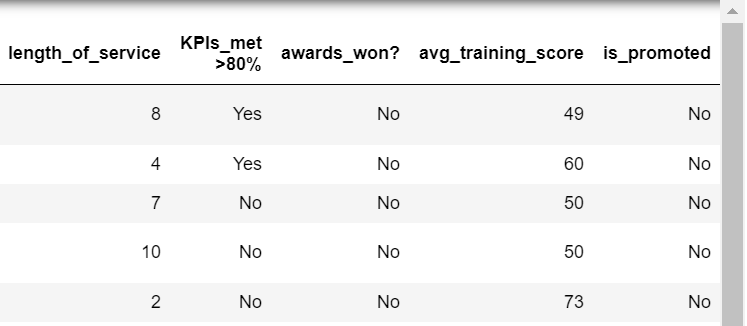


**Gambar 2. Pembuatan Bin untuk Atribut length\_of\_service**

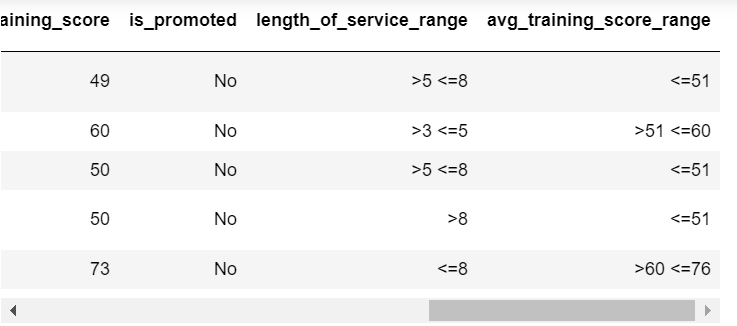


**Gambar 3. Pembuatan Bin untuk Atribut avg\_training\_score**

Setiap nilai pada kolom *length\_of\_service* dan *avg\_training\_score* akan diganti dengan nilai bin yang telah dibuat. Berikut adalah tampilannya sebelum dilakukan pembuatan *bin* dan setelah pembuatan *bin* dilakukan.



**Gambar 4. Nilai Atribut length\_of\_service dan avg\_training\_score Sebelum Dibuat ke Dalam Bin**



**Gambar 5. Nilai Atribut length\_of\_service dan avg\_training\_score Setelah Dibuat ke Dalam Bin**

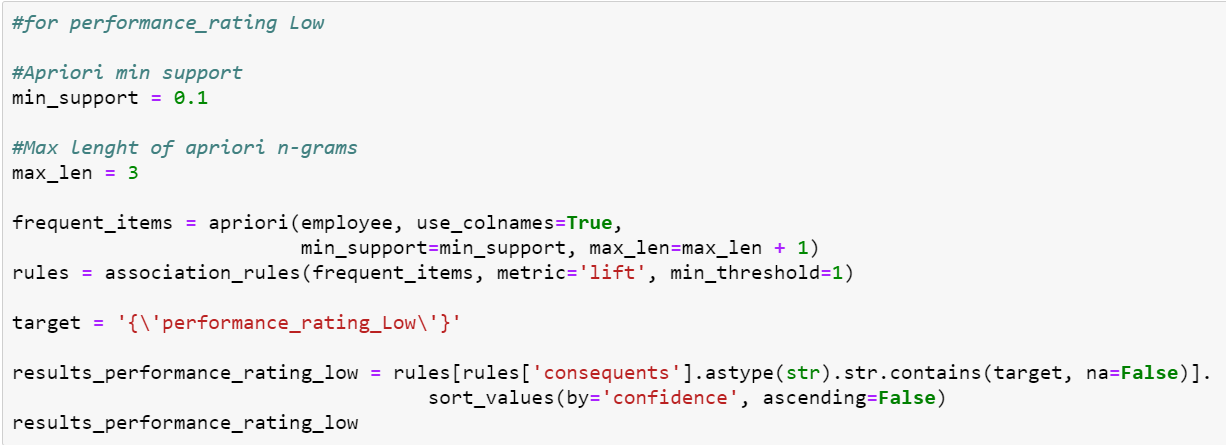
Dalam penerapan algoritma apriori terdapat beberapa parameter yang perlu ditentukan. Variabel-variabel yang akan dilihat keterkaitannya satu sama lain perlu diinisialisasi, sehingga dalam penerapan algoritma apriori hanya akan menggunakan kolom-kolom yang telah diinisialisasi untuk menggenerate *association rule*. Adapun dalam projek ini menggunakan 7 kolom dari 14 kolom yang ada di *dataset* yaitu, *performance\_rating, education, no\_of\_trainings, length\_of\_service\_range, KPIs\_met > 80%, awards\_won?,* dan *avg\_training\_score\_range*. Selain itu, terdapat beberapa parameter lainnya yang perlu ditentukan sebelum algoritma apriori dijalankan, yaitu:

* *Minimum support* = 0.1
* *Minimum confidence* = 0.6
* Maksimum *frequent itemset* yang dihasilkan = 3

Paramater-parameter tersebut diperoleh berdasarkan ukuran ketertarikan dari setiap *association rule* yang dihasilkan dengan algoritma apriori.

Setelah parameter ditentukan, maka akan dilakukan penerapan algoritma apriori dengan menentukan targetnya sebagai *performance rating* (*low, good, great, excellent, outstanding*).

1. Berikut adalah penerapan algoritma apriori dengan targetnya adalah performance\_rating\_Low.



**Gambar 6. Kode Program Penerapan Algoritma Apriori untuk Performance Rating Low**

Model tersebut menghasilkan 4 *association rule* yang memenuhi *minimum support* dan *minimum confidence*, yaitu:

**Tabel 3. Association Rule Performance Rating Low**

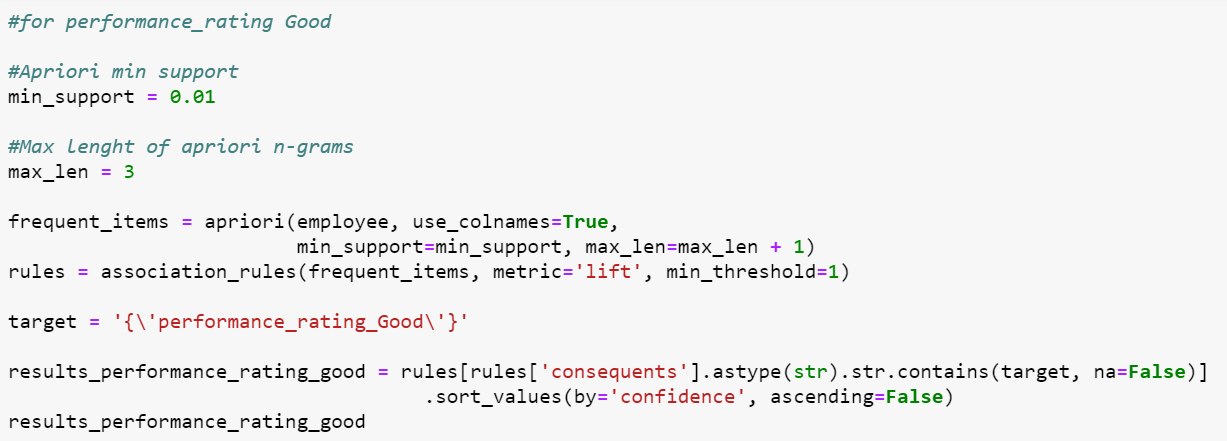
|  |  |
| --- | --- |
| ***Antecedents*** | ***Consequents*** |
| KPIs\_met > 80%\_No, awards\_won?\_No, education\_Below Secondary | Performance\_rating\_Low |
| KPIs\_met > 80%\_No, education\_Below Secondary | Performance\_rating\_Low |
| awards\_won?\_No, education\_Below Secondary | Performance\_rating\_Low |
| education\_Below Secondary | Performance\_rating\_Low |

1. Berikut adalah penerapan algoritma apriori dengan targetnya adalah performance\_rating\_Good.



**Gambar 7. Kode Program Penerapan Algoritma Apriori untuk Performance Rating Good**

Model tersebut tidak menghasilkan *rule* yang memenuhi *minimum support* dan *minimum confidence*. Oleh karena itu, berikut ditampilakn *rule* yang dihasilkan ketika *minimum support* diubah menjadi 0.01.



**Gambar 8. Kode Program Penerapan Algoritma Apriori untuk Performance Rating Good (support diubah)**

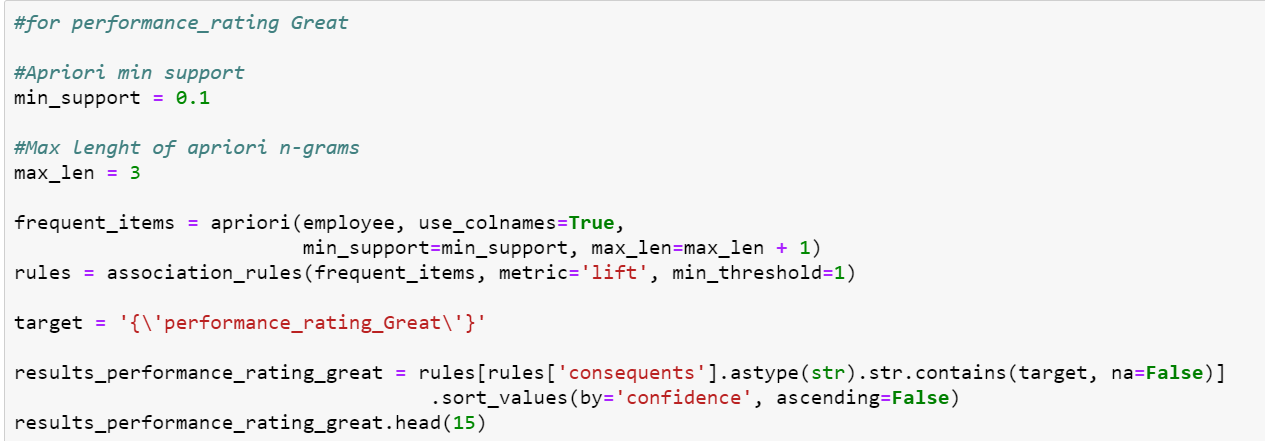
Model tersebut menghasilkan 3 *association rule*.

**Tabel 4. Association Rule Performace Rating Good**

|  |  |
| --- | --- |
| ***Antecedents*** | ***Consequents*** |
| KPIs\_met > 80%\_No, awards\_won?\_No, education\_Master’s & above | performance\_rating\_Good |
| KPIs\_met > 80%\_No, education\_Master’s & above | performance\_rating\_Good |
| KPIs\_met > 80%\_No, no\_of\_trainings\_1, education\_Master’s & Above | performance\_rating\_Good |

Namun, ketiga *rule* tersebut tidak memenuhi *minimum support* dan *minimum confidence* yang telah didefinisikan.

1. Berikut adalah penerapan algoritma apriori dengan targetnya adalah performance\_rating\_Great.



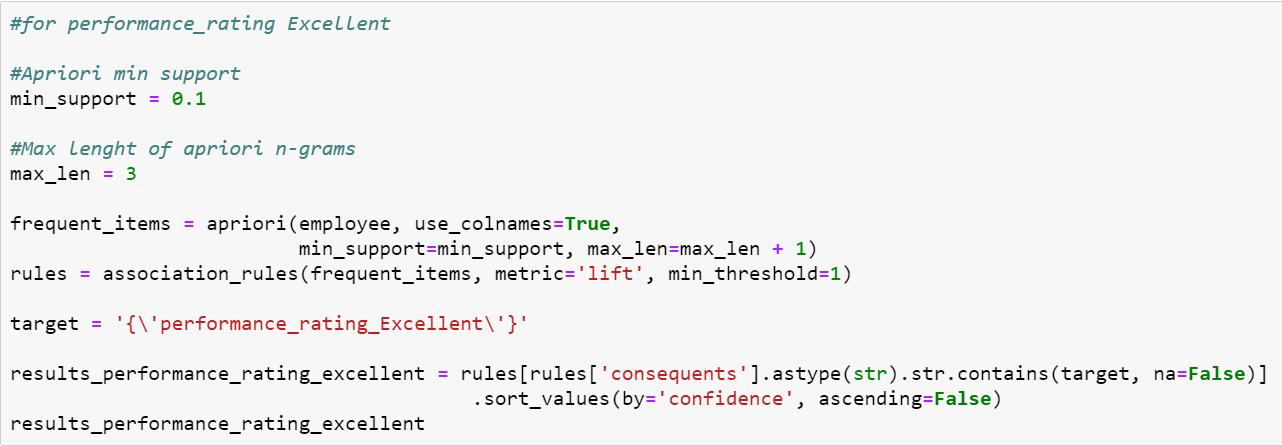
**Gambar 9. Kode Program Penerapan Algoritma Apriori untuk Performance Rating Great**

Model tersebut menghasilkan 10 *association rule* yang memenuhi *minimum support* dan *minimum confidence*, yaitu:

**Tabel 5. Association Rule Performance Rating Great**

|  |  |
| --- | --- |
| ***Antecedents*** | ***Consequents*** |
| Education\_Bachelor’s, KPIs\_met > 80%\_No, avg\_training\_score > 60 <=76 | performance\_rating\_Great |
| Awards\_won?\_No, education\_Bachelor’s, KPIs\_met > 80%\_No | performance\_rating\_Great |
| Education\_Bachelor’s, KPIs\_met > 80%\_No | performance\_rating\_Great |
| No\_of\_trainings\_1, education\_Bachelor’s, KPIs\_met > 80%\_No | performance\_rating\_Great |
| Awards\_won?\_No, KPIs\_met > 80%\_No, avg\_training\_score > 60 <=76 | performance\_rating\_Great |
| KPIs\_met > 80%\_No, avg\_training\_score > 60 <=76 | performance\_rating\_Great |
| Awards\_won?\_No, KPIs\_met > 80%\_No | performance\_rating\_Great |
| KPIs\_met > 80%\_No | performance\_rating\_Great |
| No\_of\_trainings\_1, Awards\_won?\_No, KPIs\_met > 80%\_No | performance\_rating\_Great |
| No\_of\_trainings\_1, KPIs\_met > 80%\_No | performance\_rating\_Great |

1. Berikut adalah penerapan algoritma apriori dengan targetnya adalah performance\_rating\_Excellent.



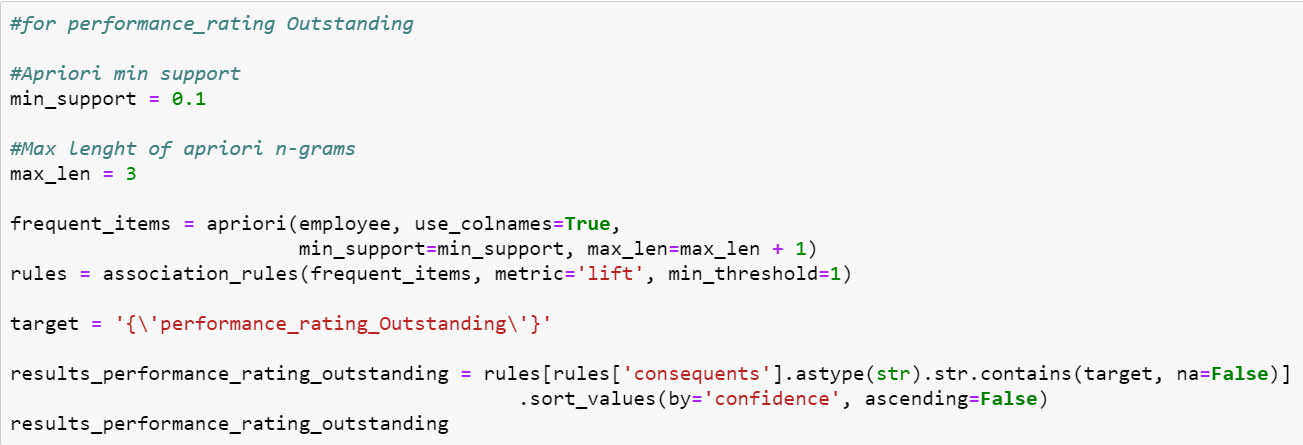
**Gambar 10. Kode Program Penerapan Algoritma Apriori untuk Performance Rating Excellent**

Model tersebut menghasilkan 7 *association rule* yang memenuhi *minimum support* dan *minimum confidence*, yaitu:

**Tabel 6. Association Rule Performance Rating Excellent**

|  |  |
| --- | --- |
| ***Antecedents*** | ***Consequents*** |
| Awards\_won?\_Yes, KPIs\_met>80%\_Yes, no\_of\_trainings\_1 | performance\_rating\_Excellent |
| Education\_Bachelor’s, awards\_won?\_Yes, KPIs\_met > 80%\_Yes | performance\_rating\_Excellent |
| Awards\_won?\_Yes, KPIs\_met > 80%\_Yes | performance\_rating\_Excellent |
| Education\_Bachelor’s, awards\_won?\_Yes, no\_of\_trainings\_1 | performance\_rating\_Excellent |
| awards\_won?\_Yes, no\_of\_trainings\_1 | performance\_rating\_Excellent |
| Education\_Bachelor’s, awards\_won?\_Yes | performance\_rating\_Excellent |
| Awards\_won?\_Yes | performance\_rating\_Excellent |

1. Berikut adalah penerapan algoritma apriori dengan targetnya adalah performance\_rating\_Outstanding.



**Gambar 11. Kode Program Penerapan Algoritma Apriori untuk Performance Rating Outstanding**

Model tersebut menghasilkan 3 *association rule* yang memenuhi *minimum support* dan *minimum confidence*, yaitu:

**Tabel 7. Association Rule Performance Rating Outstanding**

|  |  |
| --- | --- |
| ***Antecedents*** | ***Consequents*** |
| KPIs\_met>80%\_Yes, awards\_won?\_No, education Bachelor’s | performance\_rating\_outstanding |
| No\_of\_trainigs\_1, KPIs\_met>80%\_Yes, awards\_won?\_No | performance\_rating\_outstanding |
| KPIs\_met>80%\_Yes, awards\_won?\_No | performance\_rating\_outstanding |

## **4.4 *Assess Model***

*Assess model* merupakan tahapan yang dilakukan untuk menilai kesesuaian model yang telah dibangun dengan kriteria sukses yang telah didefinisikan. Secara umum, hasil yang diperoleh dari pembangunan model dengan menggunakan algoritma apriori telah menghasilkan *rule* yang baik yang dinilai berdasarkan kesesuaiannya dengan minimum support dan minimum confidence. Diperoleh beberapa *rule* yang berkaitan dalam menentukan kinerja karyawan (*performance rating*) *low*, *great*, *excellent*, dan *outstanding*. Namun untuk *performance rating* dengan kategori good tidak diperoleh *rule* yang memenuhi *minimum support* dan *minimum confidence*.

Berikut adalah rangkuman jumlah *rule* yang diperoleh dari setiap kategori *performance rating,* serta rentang *confidence* dari *rule* yang dihasilkan tersebut.

|  |  |  |
| --- | --- | --- |
| ***Performance rating*** | **Jumlah *rule*** | ***Range confidence*** |
| Low | 4 | 0.95 – 0.96 |
| Good | 3 | 0.61 – 0.64 |
| Great | 10 | 0.62 – 0.86 |
| Excellent | 7 | 0.88 – 0.95 |
| Outstanding | 3 | 0.70 – 0.74 |

Berdasarkan Tabel dapat dilihat bahwa model yang dibangun telah cukup baik dalam menggenerate *association rule* dalam menentukan kinerja karyawan berdasarkan *performance rating*.