Prediksi Data Kartu Kredit Nasabah Bank Dengan Model Decision Tree

Rimba Erlangga
Program Studi Ilmu Komputer
Dept. Ilmu Komputer dan Elektronika
Universitas Gadjah Mada

M.Fawwaz Mayda
Program Studi Ilmu Komputer
Dept. Ilmu Komputer dan Elektronika
Universitas Gadjah Mada

Abstract

Makalah ini adalah makalah yang kami buat dalam rangka memenuhi penilaian *C-Compiler* bidang *Data Mining*. Makalah ini berisi hasil prediksi kami terhadap data yang diberikan yang kami implementasi kan dalam Algoritma *Decision Tree* untuk keperluan klasifikasi.

I. Pendahuluan

Permasalahan yang diberikan disini adalah untuk melakukan klasifikasi terhadap status default(apakah kartunya aktif atau tidak) dari kartu kredit. Disini diberikan sebanyak 25 Kolom yang terdiri dari:

A : 'ID',
kategorik, yaitu ID unik untuk setiap nasabah. Kolom ini hanya berguna untuk identifikasi dan tidak akan kami gunakan dalam klasifikasi nantinya.

B : 'LIMIT_BAL',
numerik, yaitu jumlah kredit yang tersisa.

C : 'SEX',
kategorik, jenis kelamin nasabah, yaitu 0 untuk laki-laki dan 1 untuk perempuan .

D : 'EDUCATION',
kategorik, tingkat pendidikan terakhir nasabah,

E : 'MARRIAGE',
kategorik, status pernikahan nasabah

F : 'AGE',
numerik, umur nasabah

G-L : 'PAY_1', 'PAY_2', 'PAY_3', 'PAY_4', 'PAY_5', 'PAY_6',
numerik, yaitu ketepatan si nasabah untuk membayar tagihan. ‘PAY_1’ adalah seberapa tepat nasabah membayar tagihan di bulan pertama dan seterusnya untuk ‘PAY_6’.

M-R : 'BILL_AMT1', 'BILL_AMT2', 'BILL_AMT3', 'BILL_AMT4', 'BILL_AMT5', 'BILL_AMT6',
Merupakan data numerik, yang merupakan tagihan dibulan ke 1 (‘BILL_AMT1’) hingga bulan ke-6 (‘BILL_AMT6’)

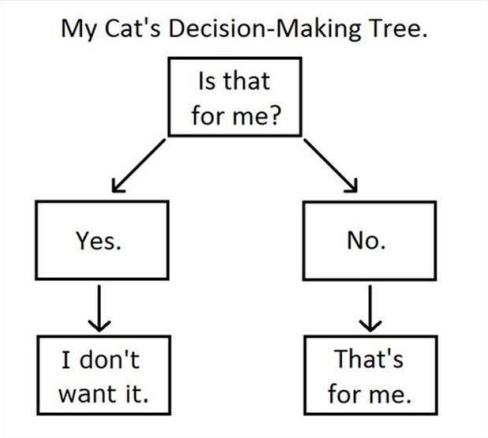
S-X : 'PAY_AMT1', 'PAY_AMT2', 'PAY_AMT3', 'PAY_AMT4', 'PAY_AMT5', 'PAY_AMT6',
Merupakan data numerik, yang menyatakan jumlah yang dibayar si nasabah pada bulan ke 1(‘PAY_AMT1’) hingga bulan ke-6 (‘PAY_AMT6’).

Y : 'default'
status yang diberlakukan pada kartu kredit nasabah.

Diantara 25 kolom tersebut terdapat 24 kolom yang berguna sebagai predictor dan kolom terakhir yang berguna sebagai target variabel. Disini yang menjadi target variabel nya adalah kolom ‘*default*’. Ke-24 kolom yang ada disini akan berguna sebagai penentu hasil pada kolom ‘*default*’ (sesuai namanya sebagai *predictor* kolom). Tetapi perlu kita lakukan analisa data sehingga mendapatkan kolom yang mana yang bisa

berguna dalam prediksi. Metode ini dikenal dengan sebutan *feature selection*, hal ini didasari dari pemikiran bahwa akan ada beberapa *parameter* yang dapat memberikan kontribusi besar terhadap hasil prediksi dan ini sebaiknya kita coba temukan agar mempermudah algoritma bekerja nantinya.

Lalu untuk bagian algoritmanya kami akan menggunakan algoritma *Decision Tree.* Hal ini dikarenakan dengan adanya percampuran dari data yang terdiri dari data *categorical* dan *numeric* selain itu algoritma *Decision Tree* dapat meniru pola pikir manusia yang biasanya memutuskan 'ya' atau 'tidak' melalui banyak scenario jika-maka.

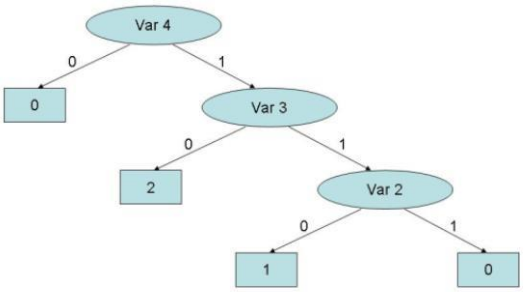


Proses *Data Mining* seperti diatas akan sangat berguna dalam berbagai hal di masa Industri 4.0 sekarang ini. Dikarenakan sekarang sedang maraknya penggunaan *Machine Learning* untuk membantu menyelesaikan berbagai permasalahan kehidupan manusia, dan untuk mencapai hal tersebut diperlukan banyak sekali data untuk digunakan dalam rangka *training model*. Jumlah data yang besar tersebut juga dapat merujuk pada penggunaan *Big Data* dalam berbagai proses bisnis. Disini dikarenakan banyak nya data juga akan digunakan metode *Feature Selection* untuk mempermudah pekerjaan.

II.Metode

A. Feature Selection

Feature Selection atau Feature Reduction adalah suatu kegiatan yang umumnya bisa dilakukan secara preprocessing dan bertujuan untuk memilih feature yang berpengaruh dan mengesampingkan feature yang tidak berpengaruh dalam suatu kegiatan pemodelan atau penganalisaan data



B. Decision Tree

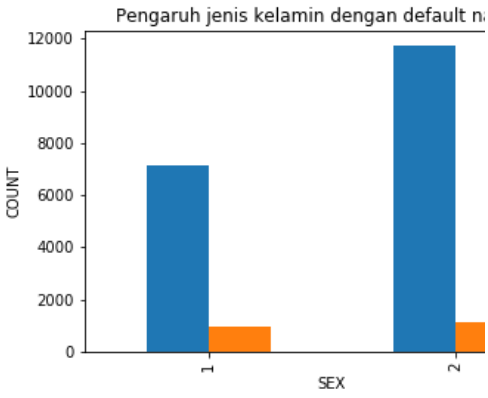
Decision tree adalah sebuah diagram alir yang berbentuk seperti struktur pohon yang mana setiap internal node menyatakan pengujian terhadap suatu atribut, setiap cabang menyatakan output dari pegujian tersebut dan leaf node menyatakan kelas–kelas atau distribusi kelas. Node yang paling atas disebut sebagai root node atau node akar. Sebuah root node akan memiliki beberapa edge keluar tetapi tidak memiliki edge masuk, internal node akan memiliki satu edge masuk dan beberapa edge keluar, sedangkan leaf node hanya akan memiliki satu edge masuk tanpa memiliki edge keluar.

III.Analisis dan Implementasi.

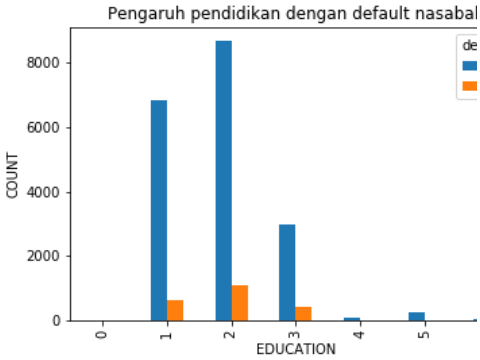
A. Feature selection

Untuk melakukan *feature selection* kita pertama akan melakukan *plotting* terhadap berbagai kolom sebagai sumbu-x dan kolom *default* sebagai kolom-y. Disini kita akan menggunakan beberapa library Pyhton seperti *Pandas, Matplotlib,* dan *Sklearn.*

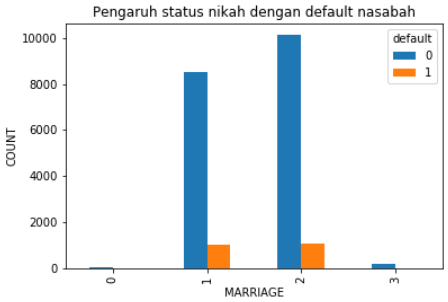
```
pd.crosstab(df['SEX'],df.
plt.title('Pengaruh jenis
plt.ylabel('COUNT')
plt.show()
```



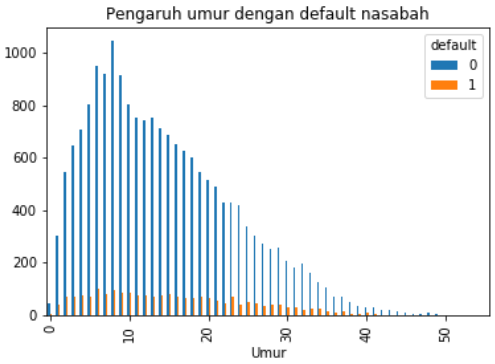

```
pd.crosstab(df['EDUCATION'],df.default).plot
plt.title('Pengaruh pendidikan dengan defaul
plt.ylabel('COUNT')
plt.show()
```




```
pd.crosstab(df['MARRIAGE'],df.default).plot(kind='bar')
plt.title('Pengaruh status nikah dengan default nasabah')
plt.ylabel('COUNT')
plt.show()
```




```
pd.crosstab(df.AGE,df.default).plot(kind='bar')
plt.title('Pengaruh umur dengan default nasabah')
plt.xlabel('Umur')
plt.xscale('linear')
plt.show()
```

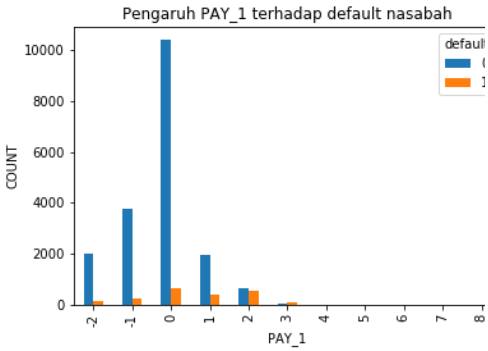


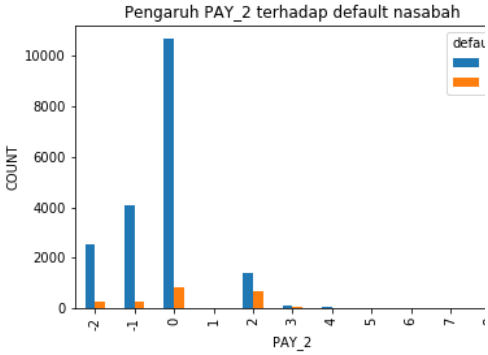
Dilihat dari sini kami mendapat bahwa hasil dari grafik pengaruh kolom-kolom diatas terhadap *default* dari nasabah harus ikut dipertimbangkan dikarenakan pada beberapa *value* dapat memberikan hasil prediksi yang berbeda.

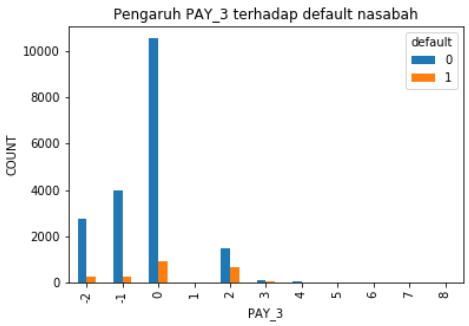
Untuk kolom ‘PAY_*’ kami menggunakan *Cross Tabulation* sama seperti kolom sebelumnya untuk mendapat grafik dari

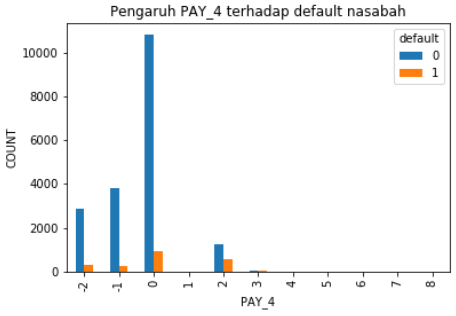
pengaruh kolom tersebut terhadap *default* nasabah.

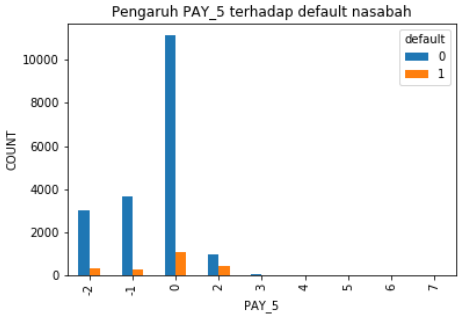
```
for i in range(1,7):
    st='PAY_' + str(i)
    pd.crosstab(df[st],df.default).plot(kind='bar')
    plt.title('Pengaruh '+st+' terhadap default nasaba
    plt.ylabel('COUNT')
    plt.show()
```

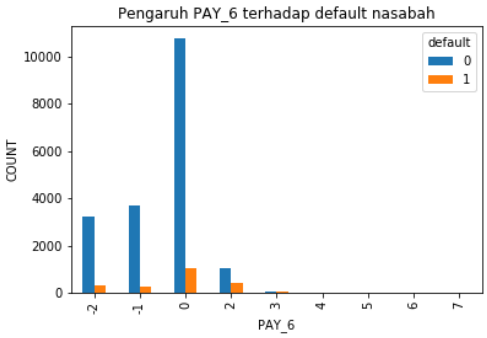










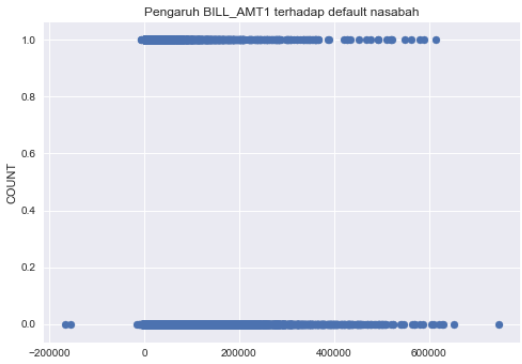


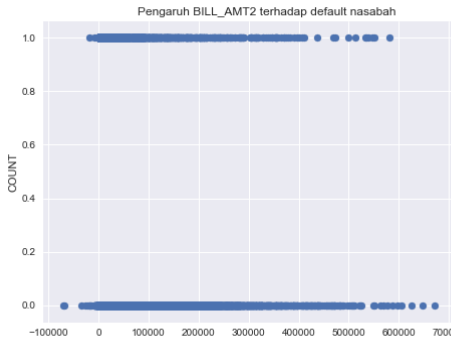
Dari kolom diatas kami menyimpulkan bahwa tidak semua kolom memberikan gambaran yang bagus untuk prediksi kali ini. Disini kami mengamati bahwa kolom ‘PAY_1’ memberikan dampak yang cukup besar dalam hasil prediksi diikuti oleh kolom ‘PAY_2’ yang sebenarnya agak kurang lebih dengan kolom ‘PAY_3’ dan ‘PAY_4’. Lalu kami juga memilih kolom ‘PAY_6’ untuk digunakan dalam prediksi dibandingkan dengan kolom ‘PAY_5’ dikarenakan adanya perbedaan hasil dimana kami melihat kolom ‘PAY_6’ menghasilkan data yang lebih akurat.

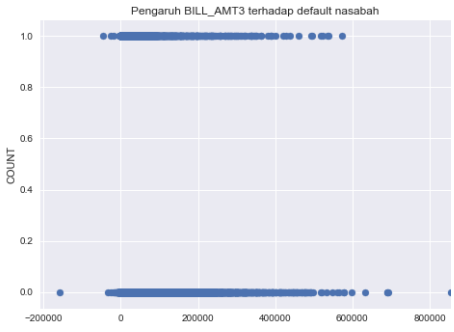
Untuk kolom ‘BILL_AMT*’ kami menggunakan *Scatter Plot*

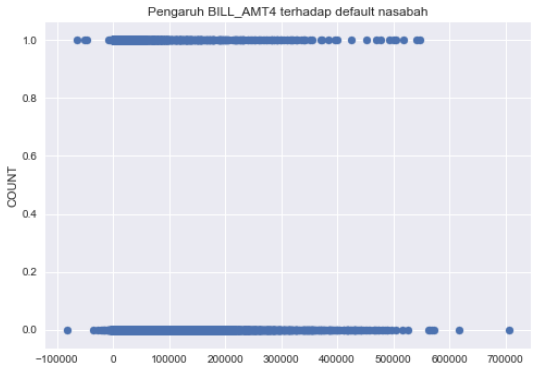
untuk mendapatkan grafik pengaruh terhadap default nasabah.

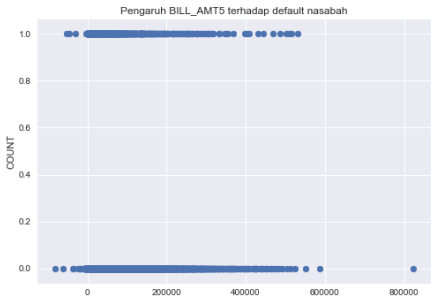
```
for i in range(1,7):
    st='BILL_AMT' + str(i)
    plt.scatter(x=df[st],y=df.default)
    plt.title('Pengaruh '+st+' terhadap default
    plt.ylabel('COUNT')
    plt.show()
```

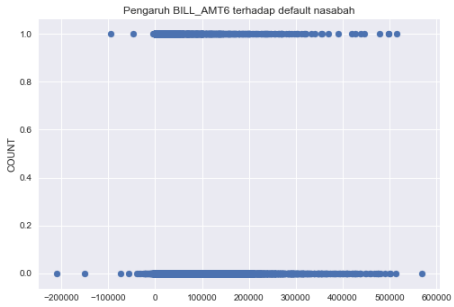










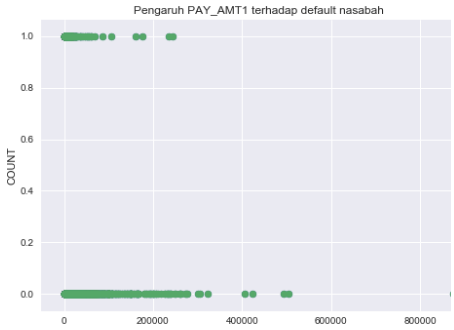


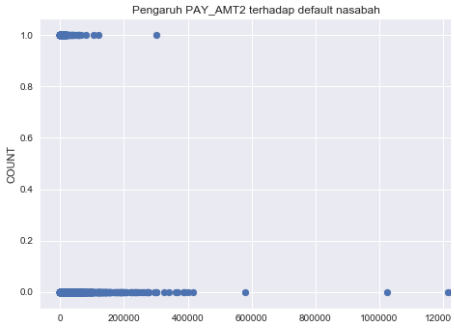
Diantara kolom yang ada kami memilih kolom ‘BILL_AMT1’, dikarenakan cukup untuk menggambarkan distrubusi data diantara semua kolom ‘BILL_AMT*’ kami hanya memilih kolom ini dikarenakan kolom lainnya terlihat hampir sama.dalam hal kontribusi terhadapa nilai *default.*

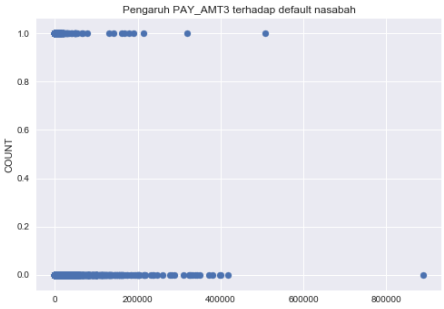
Lalu kami juga menggunakan *Scatter Plot* untuk kolom

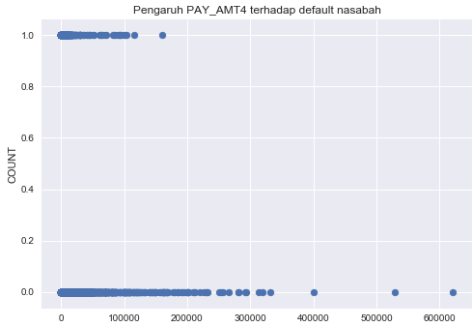
‘PAY_AMT*’ berikut grafik nya:

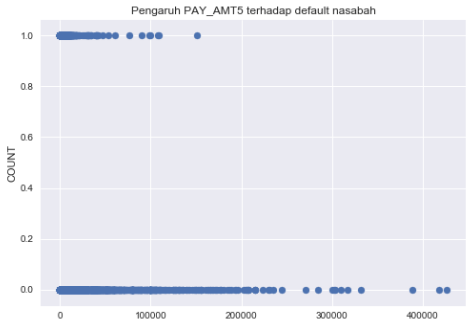
```
for i in range(1,7):
    st='PAY_AMT' + str(i)
    plt.scatter(x=df[st],y=df.default)
    plt.title('Pengaruh '+st+' terhadap d
    plt.ylabel('COUNT')
    plt.show()
```

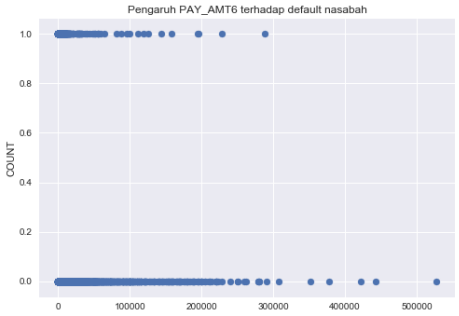








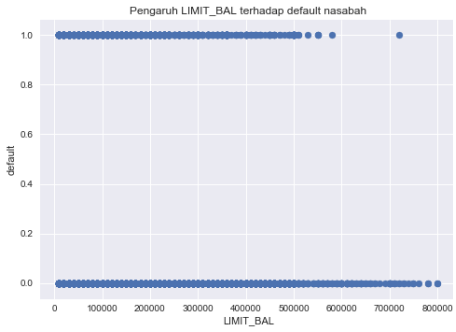




Kami memutuskan untuk tidak menggunakan kolom ini dikarenakan banyaknya *outliers* yang kami rasa dapat menganggu kinerja algoritma.

Dan kolom terakhir yang akan kami gunakan adalah kolom ‘LIMIT_BAL’ dikarenakan secara logika jumlah kredit yang dimiliki nasabah akan berkontribusi apakah status kredit nasabah akan masih aktif atau tidak.

```
plt.scatter(x=df['LIMIT_BAL'],y=df.default)
plt.title('Pengaruh LIMIT_BAL terhadap default nasabah')
plt.xlabel('LIMIT_BAL')
plt.ylabel('default')
plt.show()
```



Dapat dilihat bahwa, faktor-faktor yang paling berpengaruh yaitu: ['LIMIT_BAL','SEX','EDUCATION','MARRIAGE','AGE','PAY_1','PAY_2','PAY_6','BILL_AMT1'] karena alasan yang telah dijelaskan diatas.

IV.Pengujian dan Pembahasan.

A. Skenario Pengujian

Pengujian yang dilakukan dengan cara men-tuning parameter Algoritma *Decision Tree* dengan cara yang *Brute Force.* Disini sebelumnya kami akan men-split data menjadi data *train* dan *test* dengan masing-masing x dan y, dimana x berisi kolom predictor dan y berisi *target value.*

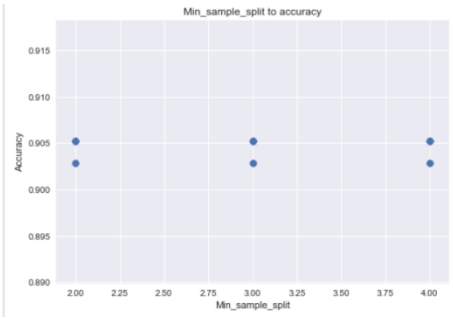
```
from sklearn.model_selection import train_test_split
to_keep=['LIMIT_BAL','SEX','EDUCATION','MARRIAGE','AGE','PAY_1','PAY_2','PAY_6','BILL_AMT1']
X=df.loc[:,to_keep]
Y=df.loc[:,'default']
x_train,x_test,y_train,y_test=train_test_split(X,Y,test_size=0.2,random_state=21)
```

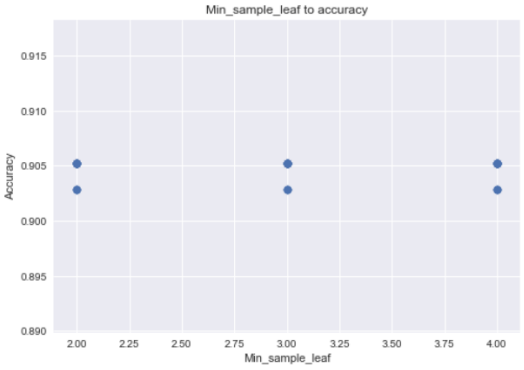
Disini kami akan men-tuning 4 parameter Decision Tree yaitu *min_sample_split, min_sample_leaf, max_depth,max_leaf_nodes.*

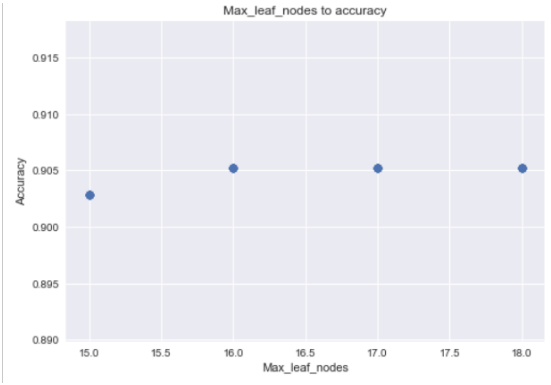
```
from sklearn.tree import DecisionTreeClassifier
mss=[]
msl=[]
md=[]
mln=[]
scores=[]
for minsamplesplit in range(2,5):
    for minsampleleaf in range(2,5):
        for maxdepth in range(15,19):
            for maxleafnodes in range(15,19):
                dt=DecisionTreeClassifier(max_depth =maxdepth ,
                                          min_samples_split = minsam
                                          min_samples_leaf=minsample
                                          , max_leaf_nodes = maxleaf
                dt.fit(x_train,y_train)
                score=dt.score(x_test,y_test)
                mss.append(minsamplesplit)
                msl.append(minsampleleaf)
                md.append(maxdepth)
                mln.append(maxleafnodes)
                scores.append(score)
```

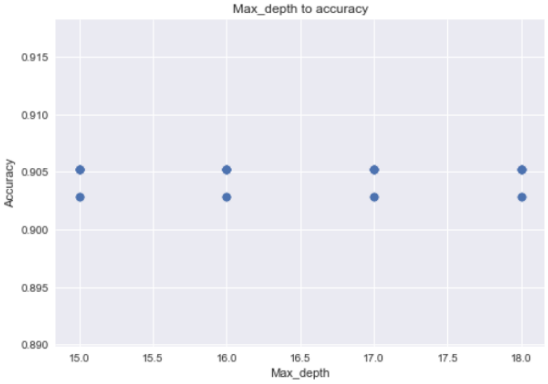
Hasil dari uji coba tadi akan kami muat dalam sebuah *DataFrame* untuk memudahkan pengamatan (dikarenakan formatnya dalam bentuk tabel). Sesudahnya kami menggambarkannya dalam bentuk grafik untuk memudahkan pengamatan dan melihat bagaimana nilainya berubah.

Berikut adalah hasilnya dalam bentuk grafik:









Dari hasil diatas kami mendapatkan parameter yang bisa memberikan hasil yang bagus yaitu dengan *Max_depth=15, max_leaf_nodes=18, min_sample_leaf=2, dan min_sample_split=2.*

V.Kesimpulan

Kesimpulan disini adalah bahwa dalam bidang Data Mining tidak hanya diperlukan algoritma tetapi lebih penting adalah analisis data yang bisa memberikan pengaruh terhadap akurasi yang lebih baik. Ini berkaca dari pengalaman kami pada saat menggunakan algoritma Decision Tree tanpa feature selection kami hanya mendapat akurasi sekitar 57% lalu setelah menggunakan feature selection kami dapati akurasi nya meningkat menjadi 60%.