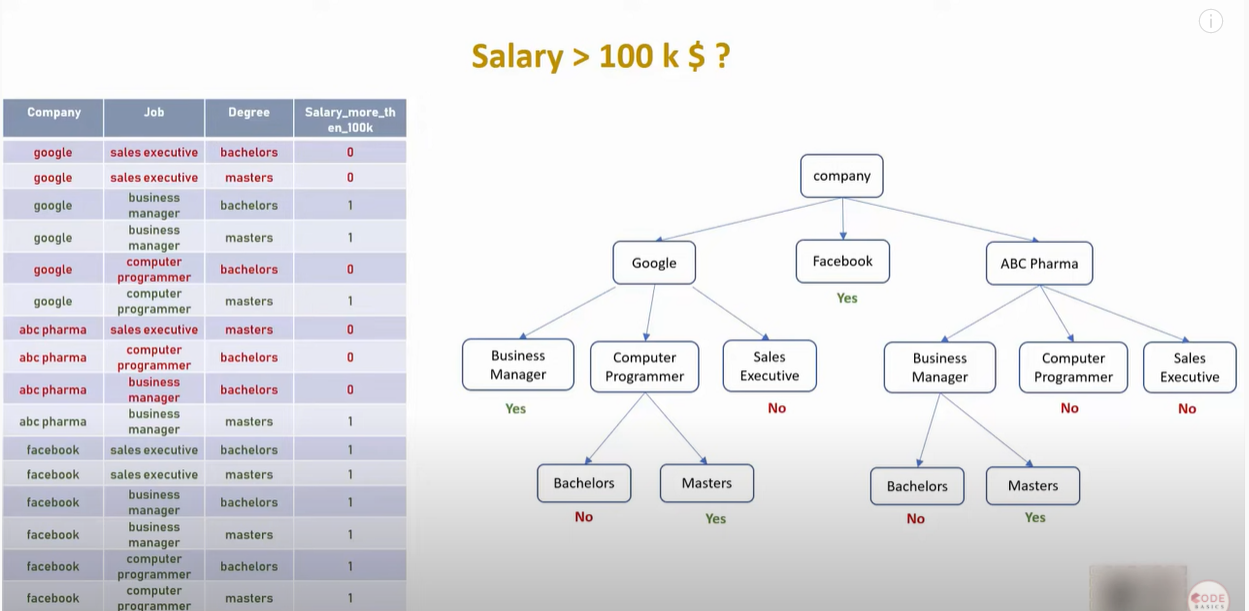
Chart, scatter chart

Description automatically generated

For this graph use SVM, KNN

But when you cannot separate in groups -> decision tres



**import** pandas **as** pd

df **=** pd**.**read\_csv("salaries.csv")

df**.**head()

|  | **company** | **job** | **degree** | **salary\_more\_then\_100k** |
| --- | --- | --- | --- | --- |
| **0** | google | sales executive | bachelors | 0 |
| **1** | google | sales executive | masters | 0 |
| **2** | google | business manager | bachelors | 1 |
| **3** | google | business manager | masters | 1 |
| **4** | google | computer programmer | bachelors | 0 |

inputs **=** df**.**drop('salary\_more\_then\_100k',axis**=**'columns')

target **=** df['salary\_more\_then\_100k']

**from** sklearn.preprocessing **import** LabelEncoder

le\_company **=** LabelEncoder()

le\_job **=** LabelEncoder()

le\_degree **=** LabelEncoder()

inputs['company\_n'] **=** le\_company**.**fit\_transform(inputs['company'])

inputs['job\_n'] **=** le\_job**.**fit\_transform(inputs['job'])

inputs['degree\_n'] **=** le\_degree**.**fit\_transform(inputs['degree'])

inputs\_n **=** inputs**.**drop(['company','job','degree'],axis**=**'columns')

inputs\_n

|  | **company\_n** | **job\_n** | **degree\_n** |
| --- | --- | --- | --- |
| **0** | 2 | 2 | 0 |
| **1** | 2 | 2 | 1 |
| **2** | 2 | 0 | 0 |
| **3** | 2 | 0 | 1 |
| **4** | 2 | 1 | 0 |
| **5** | 2 | 1 | 1 |
| **6** | 0 | 2 | 1 |

target

0 0

1 0

2 1

3 1

4 0

5 1

6 0

Name: salary\_more\_then\_100k, dtype: int64

**from** sklearn **import** tree

model **=** tree**.**DecisionTreeClassifi

model**.**fit(inputs\_n, target)

model**.**score(inputs\_n,target)

1.0 <- 100% accurate

Google = 2

Computer Engineer = 1

Bachelors degree = 0

Master degree = 1

**Is salary of Google, Computer Engineer, Bachelors degree > 100 k ?**

model**.**predict([[2,1,0]])

array([0], dtype=int64)

**Is salary of Google, Computer Engineer, Masters degree > 100 k ?**

model**.**predict([[2,1,1]])

array([1], dtype=int64)