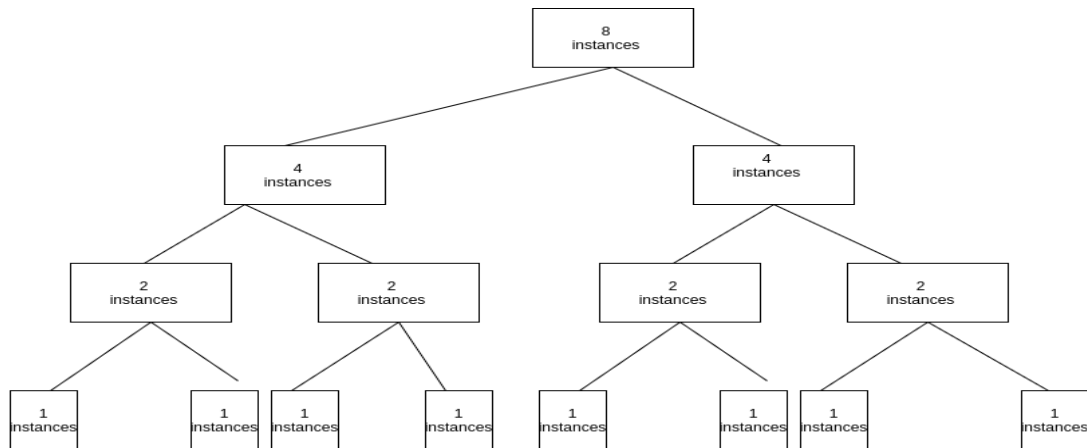


# HyperParameter Decision Tree

كل ال min and max hyper parameters بتقدر من خلالها تعمل regularization سواء كان ال model بتاعك بي over fit ولا بي under fit و الاتنين عكس بعض تقليل ال Max بيقلل ال over fitting وتزويد ال Min بيقلل برضه ال Over Fitting. طب لو كان فيه under fitting أما ازود ال max ، او انى اقلل ال Min. كمان ال None ده بيخلي ال tree بالنسبة لل feature ده non parametric بمعنى انه هيوصل لاقصى حد يقدر يوصله مثال:

What is the approximate depth of a Decision Tree trained (without restrictions) on a training set with 1 million instances?

قبل ما نتكلم فى الإجابة نفسها محتاجين نعرف انى sklearn كل ال trees فيها عبارة عن Binary tree ، وبناء على ده لو ناقشنا السؤال ده هنلاقى انى من غير اى تحكم فى الموديل ممكن فى النهاية يكون عندى ال tree depth بتاعها حوالى 20 ممكن اقل لكن مش ازيد لان ال  $m = \log_2(m)$  number of instances معنى كده لو جربت مثلا فى 8 instance هنلاقى ال depth بتاعك بقا اخرة 3 وفى الآخر هيكون عندك 8 من ال leaf nodes كل واحدة منها بتحتوى على instance واحد لانها فى الآخر binary tree ، يعنى كأنك معملتش اى training اصلا.



**criterion{"gini", "entropy"}, default="gini"**

انتا عايز ال Decision Tree تستخدم انهى Loss error function وهى بتعمل Training .

**splitter{"best", "random"}, default="best"**

هتعمل split لكل Node بناء على random feature ولا هتبحث عن ال best feature الى تعمل من عنده split .

**max\_depth:int, default=None**

عايز ال depth بتاع ال Tree يعتمد على ال log instance of data ولا تحدده انتا ، لان ، وده لو كان ال model بيعمل over fitting وقللته تقدر تعمل بيه regularization وتقلل ال over fit بتاع الموديل.

### **min\_samples\_split:int or float, default=2**

اقل عدد من ال samples يكون موجود فى ال Node عشان يحصلها Split ، وده كل ما زاد كل ما قلل ال Over Fitting لو كان فيه مشكلة over fitting ، والعكس لو كانت مشكلة under fitting .

### **min\_samples\_leaf:int or float, default=1**

اقل عدد من ال samples ينفع ال leaf node تحتويه ، والا مش هعمل اصلا split لل node الى المفروض يحصل عندها split ، لو كانت هتؤدى انى عدد ال samples ال ال Node هتحتوية ال leaf node اصغر من ال min .

### **Max\_leaf\_nodes: int, default=None**

اكبر عدد من ال leaf nodes ممكن ال Tree تحتوية .

### **max\_features:int, float or {"sqrt", "log2"}, default=None**

كمان تقدر تتحكم فى عدد ال features المستخدمة لما تيجى تعمل split من عند اى node .  
ال float من 0 ل 1 بيضربه فى عدد ال total number of features .  
الباقى بييجيب log or sqrt من عدد ال total number of features وييتدى ياخذ العدد الجديد ده بيبحث فيه لما يجى يعمل split .

### **min\_impurity\_decrease:float, default=0.0**

هيحصل split لل node لو كان هتنتج nodes تؤدي لتقليل impurity بدرجة أكبر من او مساوى ال value اللي انا محددها .

تَمَّ بِحَمْدِ اللَّهِ .