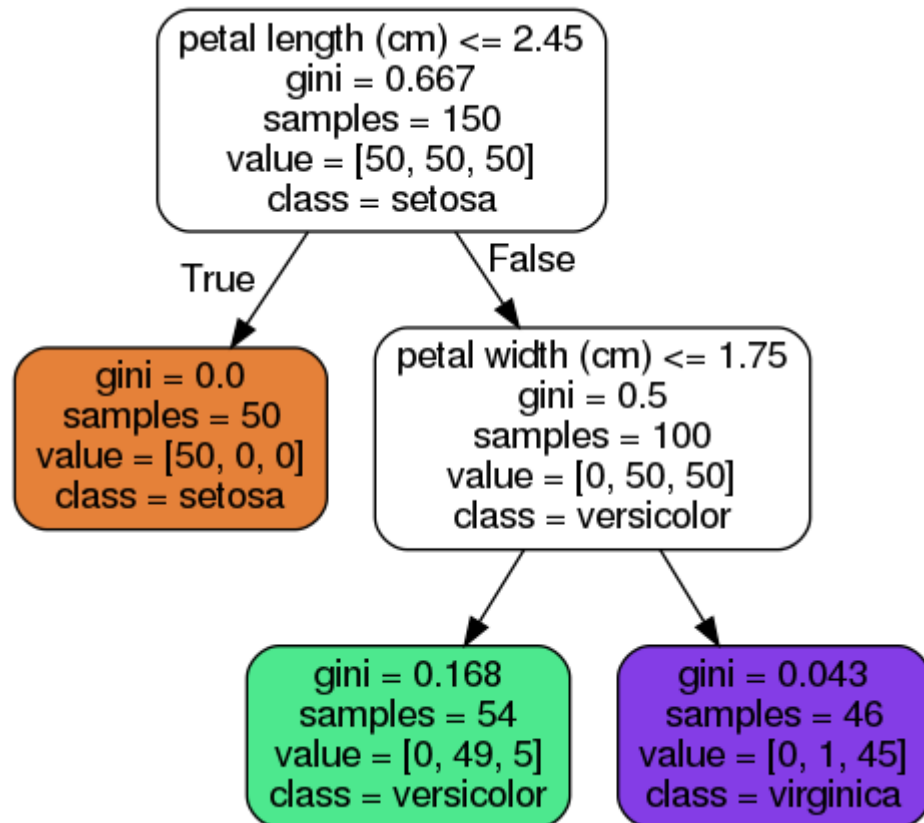


## Post 1

ال Decision Tree هي واحد من ال models القوية في ال ml ومن خلالها نقدر اننا نعمل classification and regression وكمان multi output ، وهي في الاخر عبارة عن Binary tree ، بتشتغل كأنك بتعمل if condition لو صح ووصلت ل leaf node خلاص كده ولو غلط بتفضل مكمل لحد ما توصل ل max-depth ال انتا محددة وهنا في كل مرة انتا بتسال سؤال زي ما هو واضح من ال graph ، وبالنسبة لل attributes ال جوة كل node بتقول انتا عندك اد ايه instances في ال node ده ال هو samples attribute وكمان انتا عندك اد ايه instances per class ده بالنسبة لل values attribute اما بالنسبة لل gini ده بيحدد اد ايه ال node ديه pure لو 0 تبقا pure لان مفيش غير class واحد في ال node زي left node in depth 1 في الصورة ، لكن لو كان فيه اكثر من class بيتدى اشوف ال impurity بتاعت ال node عن طريق اني ققول 1 - عدد ال instances for class k over number of samples in this node to ال power of 2 ونفس الكلام بالنسبة لل classes فلو خدنا مثال من ال في الصورة مثلا لما كان  $gini = 1 - (0/54)^2 - (49/54)^2 - (5/54)^2$  مش بس كده لما اجي اعمل prediction ل probability هلاقى كل ال instance for class k بتاخذ نفس النسبة وليكن مثلا ال class 1 هنلاقى ال 49 من ال instances دول واخدين 0.907 ولو كان من class 2 هيقا 0.09 وهنا لو جيت عملت predict لل class مش ال probability هلاقى في الاخر انو بيعمل predict لل class 1 بدم راح لل node ديه حتى لو كان ال instance ده اقرب لل class 2 لانه في الاخر بياخد ال highest probability .



## Post 2

ال Decision Tree شغال عن طريق Algorithm اسمه CART هو اختصار ل Classification And Regression Tree ببساطة ال Algorithm ده شغال عن طريق انه يعمل split لل data الى فى ال node ل subset-2 كل مرة بناء على  $k$  and  $t_k$  single feature ودول بيختارة عن طريق انه يعمل search ل احسن pair من ال  $(k, t_k)$  عن طريق انه يعمل مقارنة لكل ال features مع كل instances فى كل node لذلك ده بياخد وقت كبير جدا حتى بالنسبة ل data صغيرة لانه عبارة عن

$$O(n*m*\log(m))$$

على الرغم من انى ال prediction يكون سريع جدا لانه بياخد  $\log m$  لانه بيحتاج انه يمشى فى ال tree ويعمل check فقط على ال feature الخاص بال node الى فيه .

### POST 3

ال Descisoin Tree رغم انها ليها Params كتيرة الا انها بتعرف ب non parametric models وده بسبب انى ال params بتاعتها مش بتكون محددة سابقا فمثلا ال `max_depth` ب يكون ب None معنى كده انى هيفضل يحصل split لحد ما يحصل over fitting لو انا معملتش control فى حرية الموديل بتاعى ونلاقى هنا فى عدد من ال hyperparams بيتحكم فى ال Tree معظمهم بيبدء ب `max` or `min` وانك تقلل ال `max` وتزود ال `min` ده بيعالج مشكلة ال over fitting الى بيتحصل زى انك تقلل ال `max_depth` بتاع ال tree لو لقيت الموديل بي overfit الداتا وهكذا ال params الثانية في منها الى بيتحكم فى اقل عدد من ال samples ال node ممكن تحتوية وده بدورة بيتحكم فى انى هل هيحصل split تانى من ال node ديه ولا لا مش فقط الى بيتحكم هو ال `max_depth` وال param ده هو `min_samples_split` كمان انا ممكن اتحكم فى عدد ال leaf node بحيث معديش عدد معين زى `max_leaf_nodes` وكمان اقل عدد من ال instances ولكن بالنسبة لل leaf node نفسها وغيره من ال params الكتاب بيناقشها فى الجزئية الخاصة ب chapter 6 section `regularized hyper parameters` وفى الصورة بيوضح الفرق بين انك تسبب الموديل من غير ما تتحكم فى ال params زى الى على الشمال وبين انك تتحكم فيها زى على اليمين .

model on the right will probably generalize better.

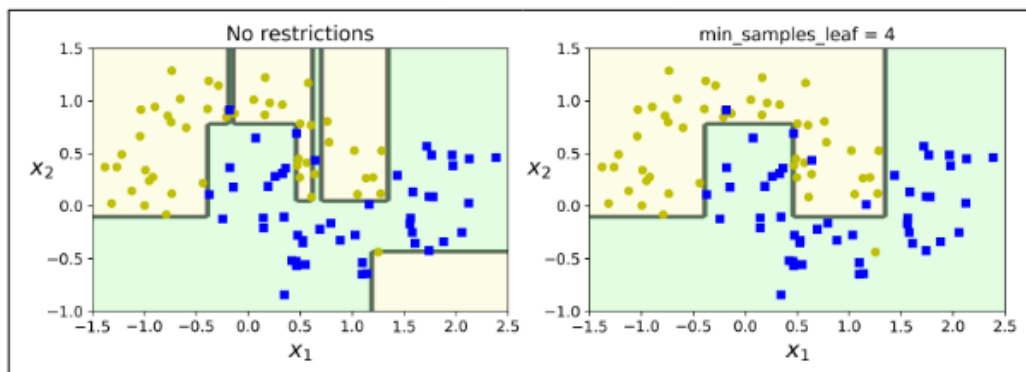


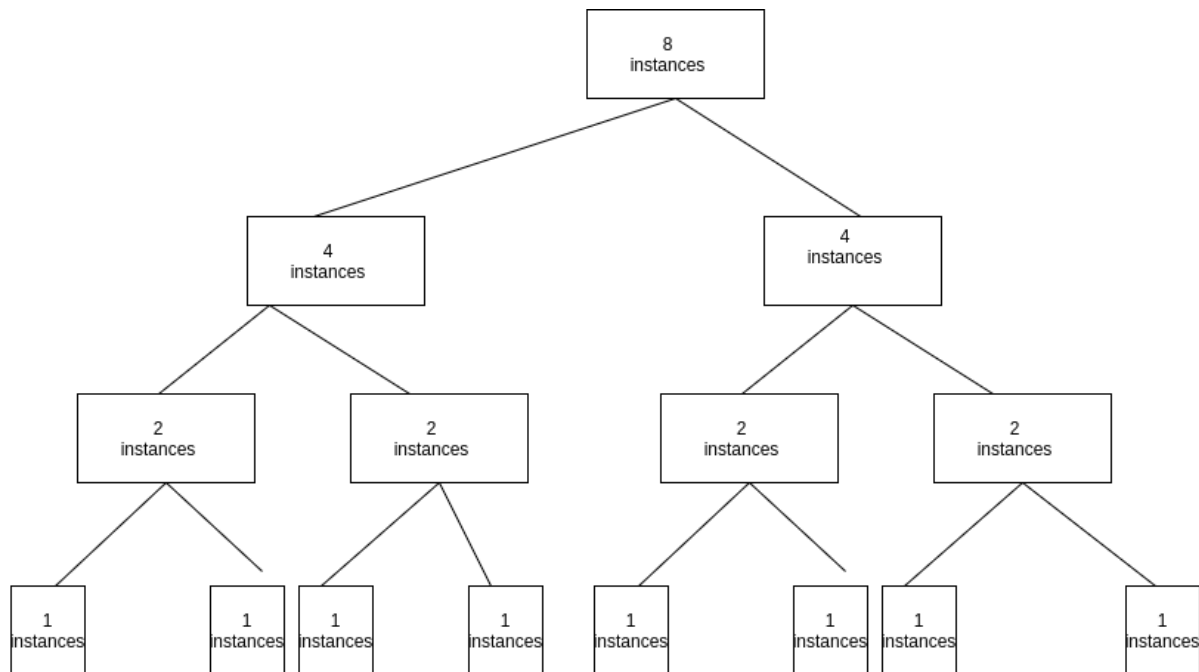
Figure 6-3. Regularization using `min_samples_leaf`

### Exercises posts

#### Post of q1

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  $\text{depth} = \log_2(m)$  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.



## Post of q2

الطبيعى انى ال child's node ال impurity بتاعها بيكون اقل من ال parent node لان ال error ده بيتوزع على two node لكن مش ده دوما بيحصل لان ساعات بيكون ال child's فيها error اكبر من ال parent لكن ده بيكون مجرد فقط تعويض ال child's الى جاية بعد كده بمعنى لو عندى داتا فى مثلا 5 امثلة

AAAAB

First impurity =  $1 - (\frac{1}{5})^2 - (\frac{4}{5})^2 = .32$

Then it will split to two nodes maybe AAA and AB

AB

$$\text{Second impurity} = 1 - \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2 = .5$$

AAA

$$\text{Third impurity} = 1 - \left(\frac{3}{3}\right)^2 = 0$$

Over all error =  $\frac{2}{3} * .5 + \frac{1}{3} * 0 = .2$  which less than  
the parent ^^