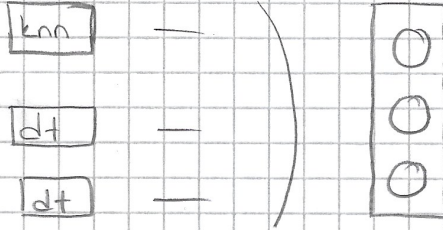
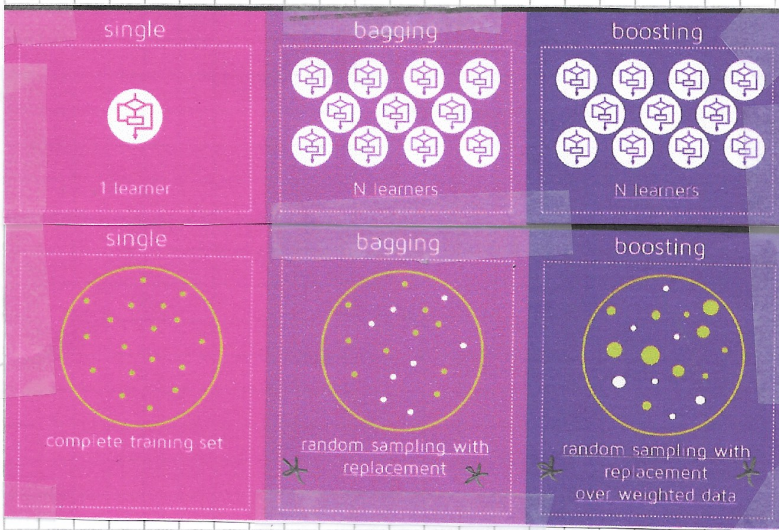


## ENSEMBLE METHOD



Bazen aynı datayı alıp farklı learnerlarda çalıştırıyoruz.  
Bazende farklı learnerlardan alıp birleştiriyoruz.



DT'leri: n tane learner ile yapınca = Random Forest (RF)

ör

\*Eğer 1000 tane diabetes training dataset var ise;

Single

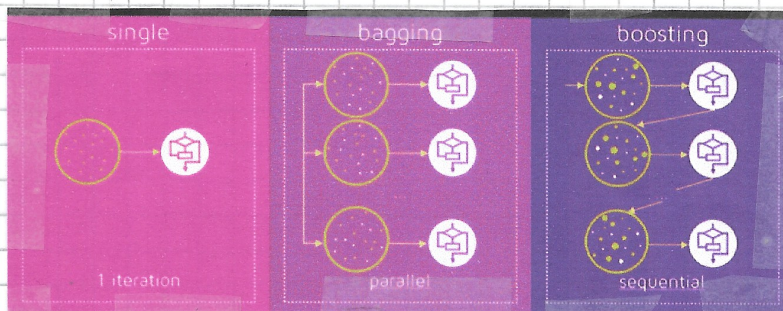
- 1000 data ile training yapar

Bagging

-  $N=10$  sample dataset  
1000 içinden 10 tane dataset oluşturuyorum  
ör 200'lük olsun  
200 tane seçip bir dataset oluşturuyorum.

Boosting

[N-learners: N tane dataset oluşturup modelleme yapmış]





BAGGING: N-learners

- Random Forest Classifier bir bagging yöntemidir. Birden fazla dt oluşturacak bizim için.
- Sample oluştururken 1. grupta seçilen data ( $x_n$ ) 2. grupta da seçilebiliyor.

Ör: Yaş  $\rightarrow$  18      Bunu DT ile yapınca çok sağlamalayabiliyor.  
 1000 tane {      Bu yüzden 200'lük datasetler seçip ayırıyorum.  
                      70      (Bu 200'lük gruplarda her zaman her çeşit (18, 19, 20...))  
                                      veri olmayabilir.

- Bu kadar kalabalık bir grubu modellemeye çalışmaktansa daha küçük grup (sample) oluşturup modellemek daha kolay.
- Yeni data gelince soruyorum - Bu kim?  $N=10$  learners diyor ki  $\rightarrow$  3'ü siyah 7'si beyaz diyor. Ben de diyorum ki o zaman bu beyaz.

Bagging: Tüm train data'ından  $N$  tane dataset oluşturur, random seçer. İlk datasette olan 2. datasette olabilir. Yeni gelen data ne diye sorduğunda oluşturulan  $N$  datasetin ne dediklerine bakar. Çoğunluğun kararına uyur.

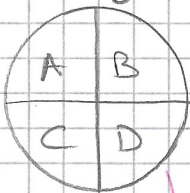
BOOSTING: N-learners

1000 tane datayı verip DT oluşturuyorum.

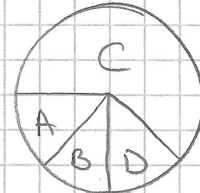
FP	-	-	-
FN	-	-	-
-	-	-	-
-	-	-	-

$\rightarrow$  Nerede eksik bilgim var ise onların daha fazla olduğu bir dataset (sample) oluşturup onu daha iyi öğrenmeye çalışıyorum. Bunu nasıl yapıyoruz - Rulet tekerleği örneği ile

Rulet Tekerleği:



$\xrightarrow{\text{C öğrencilerinin daha yaşun olduğu dataset elde etmiş oluyorum. Resample yaparak}}$



Bu genel DT iken bu daha iyi C'lere özgü DT oluyor

\* Böylece gitgide daha iyi öğreniyorum. \*

Boosting: Tüm train datayı train edip DT oluşturur. Daha sonra kendini eksik gördüğü (A-B-D sayısal olarak C'den az çıkmış) dataların (A-B-D) içerisinde daha çok olduğu yeni trainset (sample) oluşturup A-B-D özelinde bir öğrenme gerçekleştirebilir ve yeni DT oluşturur. Bu olay  $N$  kadar tekrarlar