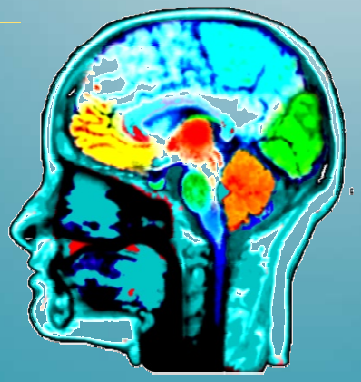
Introduction To Data Mining

Isfahan University of Technology (IUT)



Classification-ensemble

Dr. Hamidreza Hakim
hamid.hakim.u@gmail.com

ENSEMBLE TECHNIQUES

Ensemble Methods

- Construct a set of base classifiers learned from the training data

- Predict class label of test records by combining the predictions made by multiple classifiers (e.g., by taking majority vote)

-
توی خیلی از مسائل به یه شرایطی می رسیم که یک عامل برای تصمیم گیری برامون کفایت نمی کنه و می خوایم نظر چندین عامل رو بپرسیم --> خطای کمتر
تکنیک های Ensemble: میخواد کلاسیفرهای مختلف رو با هم ترکیب بکنه

Ensemble learning

- Motivations:
  - Ensemble model improves accuracy and robustness over single model methods.
  - A complex problem can be decomposed into multiple sub-problems that are easier to understand and solve (divide-and-conquer approach).

هدف:
به امید کنار هم گذاشتن یک سری تکنیک های دسته بندی کننده بتونیم دقت رو بهبود بدیم
مثل اجاره قیمت یک شهر می گیم این افراد خبره های قیمت گذاری هان و برای هر کدوم باید بریم سراغ یک خبره

Popular methods:

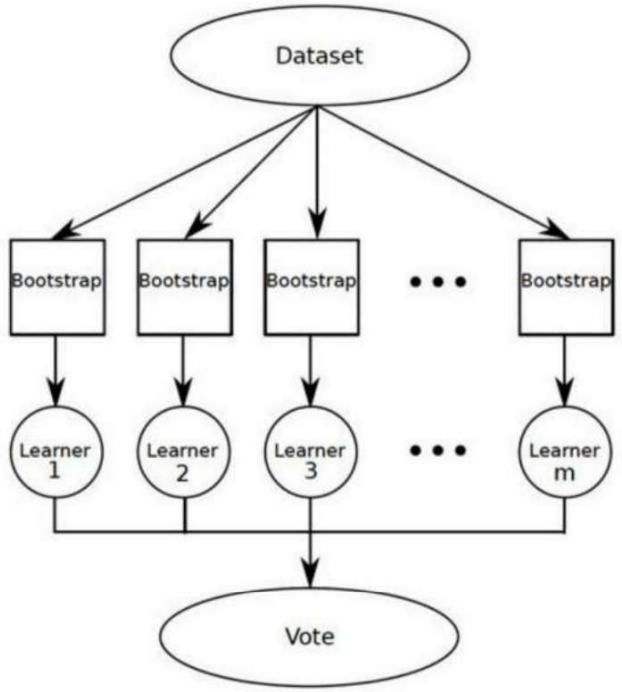
– Bagging

– Boosting

– Stacking

Bagging: Bootstrap Aggregation

- Training
  - Given a set D of d tuples, at each iteration i, a training set Di of d tuples is sampled with replacement from D (i.e., bootstrap)
  - A classifiermodel Mi is learned for each trainingset Di
- Classification: to classify an unknownsample X Each classifierMi returns its class prediction • The bagged classifier M* counts the votes and assigns the class with the most votes to X
- Regression: take the average value instead of voting
- Bagging produces a combined model that often performs significantly better than the single model built from the original training data, and is never substantially worse.
- Example
  - Random forest



Source: KDnuggets.com

Bagging:
جعبه جعبه کردن
مثلا یک درخت تصمیم ساختی این درخت تصمیم ممکنه اورفیت شده باشه
ایده اش : به جای یک درخت تصمیم چندتا درخت تولید بکن --> و هر کدوم از این درخت رو روی یک دیتای جداگانه اموزش بده و نتیجه نهایی درخت ها رو جمع بکن و رای گیری بکن روی یک دیتای تست
دیتا تیکه تیکه بکن --> هر تیکه از این دیتا می ده به دست یک مدل یا کلاسیفایر یا درخت اینجا و اموزش می بینه روی اون دیتا و می ره سراغ کلاسیفایر بعدی و n تا کلاسیفایر اموزش میدیم و تهش می خوایم نظرسنجی بکنیم و دیتای تست اینجا به همه این مدل ها داده میشه و همشون یک نظری می دن و اینجا میایم رای گیری میکنیم و هرچی اکثریت گفت میشه کلاس نهایی
برای تقسیم دیتا: test trancit ؟ --> هردفعه به دیتا شماره بده و یه تعدادی از این دیتا رو به صورت تصادفی انتخاب بکن و بده به یادگیرنده برای راند اول و به همین ترتیب
اینجا ممکنه بین دیتاها اشتراک وجود داشته باشه چون داریم داده رو تصادفی رو تقسیم میکنیم ممکنه چندتاش مشترک باشه بین چندتا درخت تصمیم

مثال: رندوم فارست مبتنی بر Bagging است

Bagging Algorithm

Algorithm 4.5 Bagging algorithm.

1: Let k be the number of bootstrap samples.
2: **for** $i = 1$ to k **do**
3: Create a bootstrap sample of size N, D_i.
4: Train a base classifier C_i on the bootstrap sample D_i.
5: **end for**
6: $C^*(x) = \underset{y}{\text{argmax}} \sum_i \delta\big(C_i(x) = y\big)$.
$\{\delta(\cdot) = 1$ if its argument is true and 0 otherwise.$\}$

Bagging (Bootstrap AGGregatING)

- Bootstrap sampling: sampling with replacement

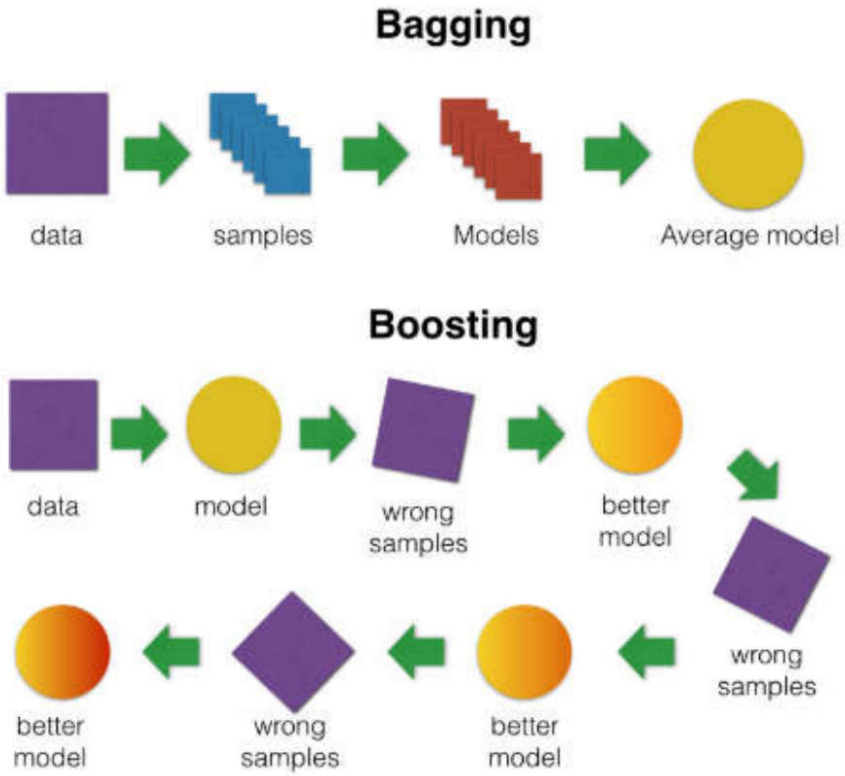
Original Data	1	2	3	4	5	6	7	8	9	10
Bagging (Round 1)	7	8	10	8	2	5	10	10	5	9
Bagging (Round 2)	1	4	9	1	2	3	2	7	3	2
Bagging (Round 3)	1	8	5	10	5	5	9	6	3	7

- Build classifier on each bootstrap sample

تکراری بودن مشکلی نداره اینجا

2. Boosting

- Comparing with bagging:
- The same base classifiers are used in both
- Boosting uses weighted voting/averaging
- Bagging is parallel while boosting is sequential
- Boosting tends to have greater accuracy, but it also risks overfitting the model to misclassified data.



Source: bradzzz.gitbooks.io

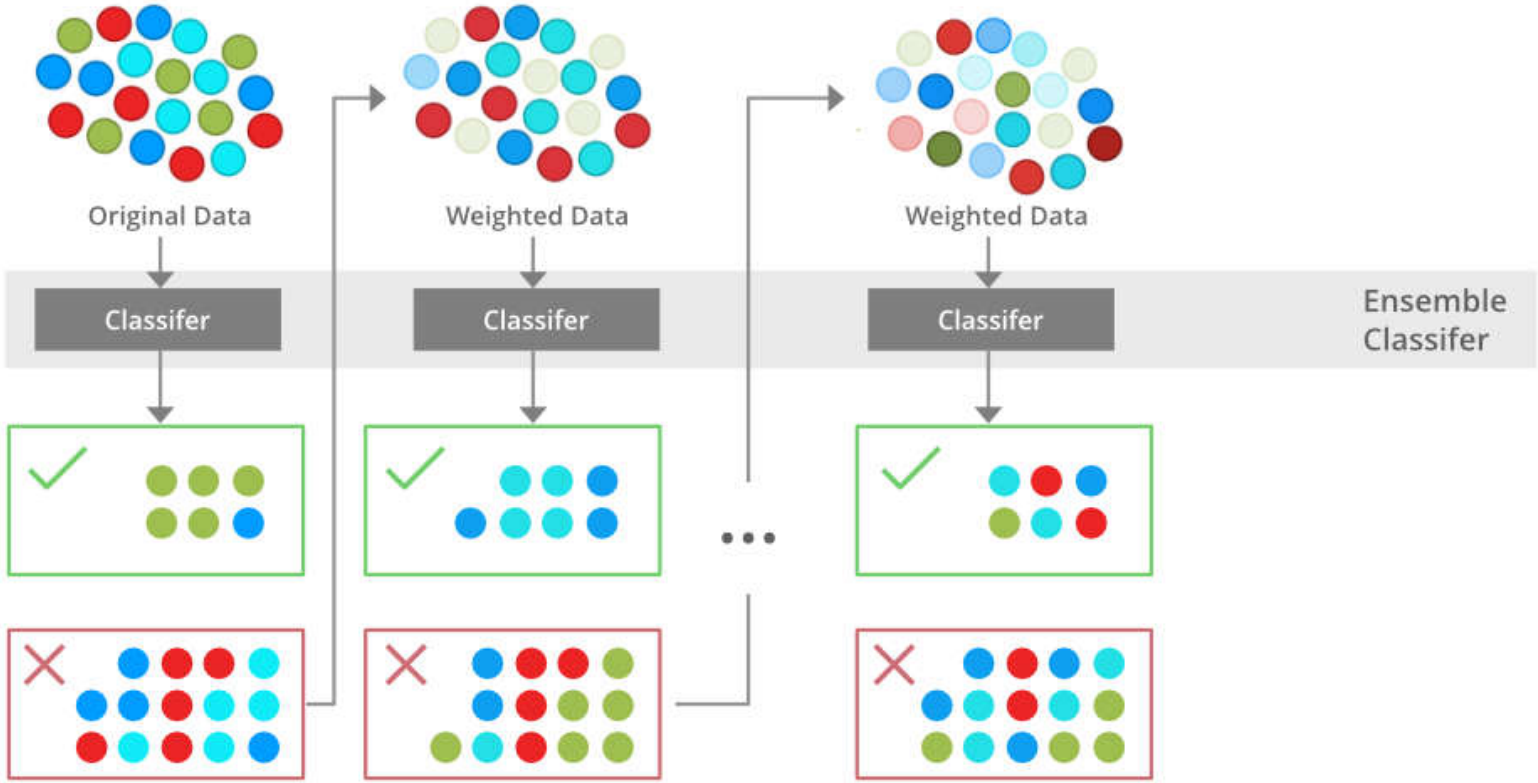
- Example
  - adaboost
  - XGBoost

Boosting:
منطقش سریالی است --> یک مدل رو انتخاب کردیم و می خوایم با چندتا مدل یک دقت خوبی به دست بیاریم --> یک مدل رو روی این دیتا تست کن و اون دیتایی که نتونسته خوب مدل یاد بگیره و گفته غلط بیا این دیتا رو بده به یک مدل دیگه (چون اونجایی که درست گفته که اوکیه ولی اونجایی که غلط گفته مشکل داره) و اینو چند گام انجام میدیم تا به یک فضایی برسیم که بدونیم خطا قابل قبول است
مدل بعدی که انتخاب میکنیم باید از همون جنس باشه --> ینی جنس همشون یکسان باید باشن (مدل ها)
ترین ست اینجا فرق داره قطعا
چندتا مدل داریم اینجا
قسمت تستش: اینجا وزن دار نگاه میکنیم --> به هر کدوم از این مدل ها یک وزنی می دیم براساس اون خطایی که میدن اینا --> وزن بیشتری رو به کسی میدیم که خطای کمتری داره ینی کمتر داده رو غلط تشخیص دادن
فرایند اموزش: این Boosting رو به صورت موازی انجام دادن چطوری؟ با وزن دهی --> وزن داده های خوب رو کم میکنه ینی اون داده هایی که مدل قبلی درست تشخیص داده؟؟؟

اینجا رکوردهای تکراری دیگه نداریم

اینجا هر مدلی نمی تونه استفاده بشه چون روش Boosting به شدت می تونه اورفیت بکنه بخاطر همین میان از مدل هایی استفاده می کنن که به شدت ساده است ینی سراغ مدل های پیچیده نباید بریم

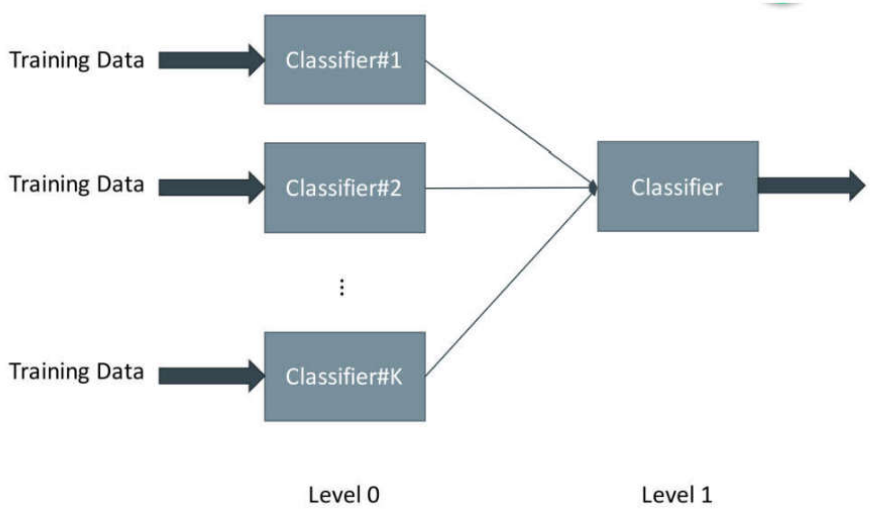
2. Boosting


Original Data
Weighted Data
Weighted Data
Classifer
Classifer
Classifer
Ensemble
Classifer

---

3. Stacking

- It introduces the concept of a metalearner, which replaces the voting procedure.
- Normally is used to combine models of different types.
- Because most of the work is already done by the level-0 learners, it makes sense to choose a rather simple algorithm for the level-1 classifier.
- Use out-of-fold predictions (OOF) as the training data for the level 1 classifier.



Stacking:
بهشون metalearner میگن
اینجا کلاسیفایر ها با هم متفاوت است --> هر کلاسیفایر رو روی کل دیتای اموزشی ترین بکن و یک خروجی میده و به همین صورت
خروجی اینا چجوری با هم ترکیب بشه؟ برای ح این موضوع یک کلاسیفایر دیگه در نظر بگیر به عنوان کلاسیفایر رهبر --> و این کلاسیفایر رو بیا روی خروجی اینا ترین بکن و این کلاسیفایر مخصوص این میشه که روی اظهار نظر بقیه بیاد نظر بده

اینجا رای گیری دیگه نداریم

چطوری اموزش میدن؟ داده های اموزش رو تقسیم بندی می کنن --> یک قسمت رو میذارن که اون کلاسیفایر نهایی اموزش ببینه و مطمئن بشن که اون کلاسیفایر نهایی هم تونسته به خوبی از پس ماجرا بربیاد

کمتر می ریم سراغ این روش ها چون خیلی دیتا میخواد که بخوایم همه اینارو اموزش بدیم