پیش پردازش:

ابتدا داده های تست و آموزش را باید ایجاد کنیم. به این منظور داده ها را از فایل های pairsDevTest و pairsDevTrainخط به خط میخوانیم. هر خط نشان دهنده دو وکتوری است که باید مقدار آن را از فایلی با همین نام بخوانیم. برخی سطرها متعلق به یک نفر و برخی دیگر متعلق به دونفر هستند. همچنین بر اساس این که دو تصویر متعلق به یک نفر هستند یا نه مقادیر 0 و 1 را به عنوان لیبل ذخیره می کنیم.

برای ترکیب دو وکتور سه راه وجود دارد:

- کانکت
- جمع
- تفریق

Train:

بعد از ایجاد دیتاست های تست و آموزش نیاز است تا داده را train کنیم.

برای این منظور از دو الگوریتم مختلف در ensemble استفاده شده است:

- AdaBoost
- RandomForest

در هر دو روش از کتابخانه sklearn استفاده شده است. با تغییر پارامترهای این الگوریتم ها و نوع کانکت نتایج زیر حاصل می‌شود:

AdaBoost

type	params	Score
concat	-	**0.502**
sum	-	**0.522**
subtract	-	**0.731**
	n_estimators=300	**0.788**
	n_estimators=300,base_estimator=dt_stump	**0.734**

RandomForest

type	params	Score
concat	-	**0.632**
	n_estimators=300,max_depth=50,max_features=0.4	**0.688**
sum	-	**0.491**
subtract	-	**0.818**
	n_estimators=300,max_depth=50,max_features=0.4	**0.817**
	n_estimators=500,max_depth=80,max_features=0.5	**0.803**
	n_estimators=300,max_depth=50,max_features=0.4, warm_start=True	**0.824**

باتوجه به این که کم کردن وکتور ها از هم در واقع فاصله شان را به دست می آورد منطقی است که حالت های subtract نتیجه بهتری می دهند.

همچنین به طور کلی الگوریتم randomForest عملکرد بهتری از الگوریتم Adaboost در این دیتاست دارد.

توضیح برخی از پارامترها:

n_estimators: حداکثر تعداد estimatorهایی که بوستینگ در آنها خاتمه یافته است. در صورت perfect fit، روند یادگیری زودتر متوقف می شود.

warm_start: هنگامی که روی True تنظیم شود، از راه حل کال قبلی برای fit کردن و اضافه کردن estimatorهای بیشتری به مجموعه استفاده می کند، در غیر این صورت، فقط یک forest کاملاً جدید را در fit میکند.