

# گزارش تسک هفته 4 - طبقه‌بندی سگ و گربه

## 1. گزارش مختصر

### (الف) رویکرد (Approach)

برای انجام این تسک، مراحل زیر را به ترتیب اجرا کردم:

1. ابتدا، تمام تصاویر از دایرکتوری بارگذاری شدند.
2. برچسب هر تصویر (0 برای گربه و 1 برای سگ) بر اساس ابتدای نام فایل آن (dog یا cat استخراج شد
3. طبق دستورالعمل تسک، تمام تصاویر به حالت خاکستری (grayscale) تبدیل و ابعاد آنها به 128 در 128 پیکسل تغییر داده شد
4. سپس، ویژگی‌های HOG (هیستوگرام گرادیان‌های جهت‌دار) از تمام تصاویر پردازش شده استخراج گردید این ویژگی‌ها، الگوهای شکلی و لبه‌ها را در تصویر توصیف می‌کنند.
5. کل مجموعه داده (شامل ویژگی‌های HOG و برچسب‌ها) با نسبت 80% آموزشی و 20% آزمایشی تقسیم‌بندی شد
6. هنگام تقسیم‌بندی، از پارامتر stratify استفاده کردم تا مطمئن شوم که نسبت تعداد سگ‌ها و گربه‌ها در هر دو مجموعه آموزشی و تست یکسان باقی می‌ماند
7. در نهایت، مدل‌های مختلف بر روی داده‌های آموزشی، آموزش داده شدند و دقت آنها با استفاده از داده‌های تست، ارزیابی گردید

### ب) جزئیات تقسیم داده و دقت تست

من در مجموع 355 تصویر داشتم. پس از تقسیم 20/80، توزیع داده‌ها به این صورت شد:

- تعداد داده‌های آموزشی: 284 تصویر
- تعداد داده‌های تست: 71 تصویر

من چندین مدل و تنظیمات مختلف را آزمایش کدم. نتایج دقت (Accuracy) به دست آمده بر روی 71 تصویر تست، به شرح زیر است:

فایل نوتبوک (آزمایش انجام شده)	دقت (Accuracy)
(درخت تصمیم پایه) <code>cat_dog_classification-simple 78%.ipynb</code>	78%
HOG (تغییر <code>cat_dog_classification-bigger cell size 80%.ipynb</code> ( $16 \times 16$ به <code>pixels_per_cell</code> )	80%
هرس درخت با <code>cat_dog_classification-with max depth 80%.ipynb</code> ( <code>max_depth=10</code> )	80%
هرس درخت با <code>cat_dog_classification-with min samples leaf 83%.ipynb</code> ( <code>min_samples_leaf=5</code> )	83%
استفاده از مدل <code>cat_dog_classification-with random forest 90%.ipynb</code> (Random Forest)	90%

### ج) بهبودها (Improvements)

همانطور که در جدول نتایج مشخص است، مدل اولیه درخت تصمیم (78%) نتیجه خیلی خوبی نداشت. این موضوع احتمالاً به دلیل Overfitting مدل بوده است.

برای بهبود نتیجه، چند روش پیشنهادی در تسك را امتحان کردم:

1. هرس کردن درخت (Pruning): با محدود کردن پارامترهای درخت، مثل `max_depth` (حداکثر عمق) یا `min_samples_leaf` (حداقل نمونه در برگ)، تلاش کردم مدل را ساده‌تر کنم تا بهتر تعمیم دهد. این کار دقต را به 80% و 83% رساند.
2. تنظیم پارامتر HOG: پارامتر `pixels_per_cell` را از (8, 16) به (8, 16) تغییر دادم تا ویژگی‌های عمومی‌تری از تصویر استخراج شود، که این کار نیز دقت را به 80% بهبود داد.
3. استفاده از طبقه‌بند جایگزین (Random Forest): بهترین نتیجه زمانی حاصل شد که طبق دستور تسك، به جای یک درخت تصمیم تنها، از مدل Random Forest (جنگل تصادفی) استفاده کردم.

این مدل که از تعداد زیادی درخت تشکیل شده، بسیار قوی‌تر عمل کرده و توانست به دقت عالی 90% دست پیدا کند.

## 2. خلاصه آموخته‌ها

### الف) چیزهایی که از HOG یاد گرفتم

- HOG یک مدل یادگیری ماشین نیست، بلکه یک روش استخراج ویژگی بسیار قدرتمند است.
- کار HOG این است که یک تصویر (مجموعه‌ای از پیکسل‌ها) را به یک بردار عددی (Vector) تبدیل می‌کند که برای مدل‌های یادگیری ماشین قابل فهم است.
- این بردار، اطلاعات مربوط به جهت لبه‌ها و شکل‌ها (Gradients) در نواحی کوچک تصویر را در خود ذخیره می‌کند. به همین دلیل برای تشخیص اشیائی مانند سگ (که گوش‌ها و پوزه مشخصی دارد) بسیار مفید است.

### ب) چیزهایی که از Decision Tree یاد گرفتم

- درخت تصمیم (DT) خود مدل طبقه‌بند (Classifier) است که بردار ویژگی HOG را به عنوان ورودی می‌گیرد و پیش‌بینی (سگ یا گربه) را تحويل می‌دهد.
- این مدل با ساختن مجموعه‌ای از قوانین ساده if-then کار می‌کند.
- فهمیدم که بزرگترین مزیت درخت تصمیم، سادگی و قابل فهم بودن آن است، اما بزرگترین نقطه ضعف آن، تمایل شدید به Overfitting است.
- دلیل اینکه دقت مدل پایه 78% بود، همین Overfitting بود. با هرس کردن درخت یا استفاده از مدل‌های گروهی (Random Forest Ensemble) که از صدها درخت استفاده می‌کنند، می‌توان این مشکل را به خوبی حل کرد و به نتایج بسیار بهتری (مانند 90%) رسید.