

Aprendizaje Automático
Práctica 1 - curso 2023-2024:
El proceso del aprendizaje automático



Universitat
de les Illes Balears

Joan Alcover Lladó
Antonio Pujol Villegas

Introducción.....	3
Tratamiento de datos.....	3
Secuencia de experimentación.....	3
Experimento 1.....	3
Experimento 2.....	4
Experimento 3.....	4
Experimento 4.....	4
Experimento 5.....	4
Experimento 6.....	5
Experimento 7.....	5
Experimento 8.....	6
Experimento 9.....	6
Validación.....	7
Conclusiones.....	7
Trabajo en pareja.....	8

Introducción

Este documento consiste en presentar el ejercicio 2 de la primera práctica de la asignatura de Aprendizaje Automático perteneciente a los estudios del grado de Ingeniería Informática del curso 2023-2024.

En dicho ejercicio se pretende hacer una serie de consideraciones, experimentos, y evaluaciones continuas con el fin de obtener un modelo de aprendizaje que nos permita clasificar, de entre 14 clases posibles asociadas a diferentes paisajes, imágenes que representen alguna de las siguientes 14 ubicaciones: habitación, costa, bosque, carretera, terreno industrial, ciudad, cocina, sala de casa, montaña, oficina, zona rural, tienda, calle, suburbio, rascacielos.

La métrica que se empleará para evaluar el nivel de correctez de la predicción de los modelos bajo los que vayamos iterando será, en principio, el Accuracy (precisión) devuelto por los mismos. Sin embargo, a lo largo del proceso de aprendizaje, se concluirá si esta es la métrica en la que deberíamos enfocarnos, o si se trataría de otra.

La técnica de aprendizaje automático empleada será, en este caso, la correspondiente al SVM (Support Vector Machine), que ya nos viene indicada en el enunciado de la práctica. El Kernel empleado en primer lugar será el Lineal, pero se harán pruebas para evaluar la idoneidad del mismo.

A continuación, se pasará a comentar los pasos que hemos ido siguiendo, y los experimentos realizados, a la hora de llevar a cabo este ejercicio.

Tratamiento de datos

- A lo largo de la secuencia de experimentación se hace una serie de redimensionamiento de las imágenes tratadas, según lo adecuado que resulten para los modelos empleados y los resultados que provean.
- Se llegará a usar la clase Standard Scaler para la normalización de los datos obtenidos, previo al entrenamiento de los modelos de clasificación.

Secuencia de experimentación

En primer lugar, se tienen dos sets de datos, imágenes de testeo e imágenes de entrenamiento; ambas muestras sin ningún tipo de tratamiento o preprocesamiento que permita tener en cuenta características concretas más allá de las que se puedan obtener directamente de las imágenes mediante métodos de obtención de características orientados a las mismas. Esto es algo que se indica directamente en el enunciado del ejercicio y de donde también se recomienda hacer uso del HOG (Histogram of Oriented Gradients) para hacer una primera extracción de características, que es lo que se pasará a hacer en el primer experimento.

Experimento 1

Según la recomendación del enunciado, se pasa a implementar el HOG (Histogram of Oriented Gradients) para la obtención de características, como se ha comentado, y se vió que habían problemas de errores de ejecución por el hecho de que las imágenes tenían diferentes dimensiones, ya que estas

no habían sido preprocesadas para su tratamiento. Por esta razón, se pasa a realizar un redimensionamiento de todas ellas a las dimensiones 100x100 píxeles.

Esto se hizo aún teniendo en cuenta que podría producirse una pérdida de información, pero con la intención de aplicar el HOG con el fin de tener una primera impresión de la posible precisión que se podría tener a la hora de predecir resultados. Para ello se utilizó la métrica del Accuracy y las muestras de entrenamiento con una separación de las mismas para entrenamiento y testeo. Esta primera iteración, con unos parámetros del HOG con 8 orientaciones, 4x4 píxeles por celda, y 1x1 celdas por bloque, proporcionó un Accuracy del 61,33 %.

Experimento 2

El del Experimento 1 pareció un resultado razonablemente positivo, pero no se quiso descartar otros posibles métodos de obtención de características, por lo que pasamos a utilizar el método LBP (Local Binary Patterns), de igual manera, tras un resize del 100x100. La razón era que se quería probar otro método de obtención de características para imágenes, y evaluar su rendimiento en comparación con el uso del HOG. Ello dio un Accuracy del 23,55%.

Experimento 3

Teniendo estos dos experimentos en cuenta, se consideró hacer uso de un tercer método de obtención de características conocido por sus siglas GLCM (Grey Level Co-Occurrence Matrix), de nuevo, con un resize de las imágenes a 100x100 píxeles. Este nuevo método de obtención de características se empleó por la misma razón que se hizo uso del LBP. Dicho método proporcionó un Accuracy del 29,33%.

Experimento 4

Una vez se completaron estas pruebas con el uso del resize y los tres métodos de obtención de características antes comentados, se consideró realizar pruebas con un modelo que emplease una combinación de los mismos, manteniendo los parámetros del HOG, a la vez que el uso del resize de 100x100. Por un lado se realizó una iteración únicamente con estas consideraciones, que devolvió un Accuracy del 35,33%, y posteriormente se hizo una segunda iteración con el uso de la clase Standard Scaler para la normalización de las muestras, que resultó en un Accuracy del 53,33%.

Experimento 5

Tras todas estas pruebas se llegó a la conclusión de que, el margen de mejora del Accuracy residía en el tipo de Kernel, y sus parámetros, que se empleaba en el SVM, que hasta ahora era el Lineal. Sin embargo, se intentó probar un tratamiento de datos alternativo al del resize de 100x100. Este consistiría en aplicar un Padding de píxeles a todas las imágenes que tuvieran unas dimensiones inferiores a la imagen de muestreo con las dimensiones más grandes. Con este proceso agregado, y volviendo a realizar la prueba exclusivamente con el HOG y los mismos parámetros que antes, se llegó a obtener un Accuracy del 69,66%.

Si bien el resultado fue el mejor obtenido, el uso del Padding suponía asumir un riesgo considerable de overfitting, ya que el número de píxeles a tener en cuenta varía grandemente entre imágenes, y

podría afectar a cómo el modelo aprende a determinar las clases. Por consiguiente, se decidió hacer pruebas en base a los resultados obtenidos mediante el K-Fold, como método de separación de datos, y el Grid Search, como método de obtención de los parámetros óptimos para los mejores resultados de la métrica elegida, que en este punto es el Accuracy.

Experimento 6

Si bien se entendía que el HOG era la mejor manera de la que se disponía de extraer características, no se tenía en claro qué otros parámetros podían acercarse a una mayor precisión de clasificación para luego emplearse en el Grid Search, por lo que directamente se pasó a realizar una batida de 6 ejecuciones para discernir qué `resize` podría ser más útil, y si el uso del `Standard Scaler` era propicio.

El experimento consistió en el uso del Kernel Lineal sin extracción de características, empleado con tres posibles `resize` (100, 250, 400) y cada ejecución hará uso y desuso de la clase `Standard Scaler`. Los resultados fueron:

Resize 100:

Normal: Accuracy del 24%

StandardScaler: Accuracy del 24%

Resize 250:

Normal: Accuracy del 23,33%

StandardScaler: Accuracy del 24,33%

Resize del 400:

Normal: Accuracy del 23,33%

StandardScaler: Accuracy del 24,33%

Con estos datos, se consideró que a la hora de realizar un `resize`, lo más óptimo era alrededor de los 250 píxeles, y en combinación con el uso del `Standard Scaler`, ya que no hay una mejora destacable si incrementamos o reducimos las dimensiones desde este punto, y debido a que los mejores resultados se obtuvieron con la estandarización de datos.

Experimento 7

Teniendo en cuenta los resultados y conclusiones anteriores, se pasó a realizar una optimización de los parámetros del HOG. Más en concreto, los parámetros que se han querido optimizar del HOG son los siguientes:

- Orientations: 8, 9, 10
- Pixels_per_cell: (4, 4), (6, 6), (8, 8)

Además se ha querido optimizar el `resize` de la imagen (50x50, 100x100, 200x200, 250x250)

El kernel utilizado para realizar dicha optimización es el kernel “rbf”.

De entre los mejores resultados se obtuvo:

Resize 200:

- Accuracy del 64% con 10 orientaciones y 8x8 píxeles por celda.

- Accuracy del 63,66% con 8 orientaciones y 8x8 píxeles por celda.

Resize 250:

- Accuracy del 63% con 8 orientaciones y 4x4 píxeles por celda.
- Accuracy del 63% con 10 orientaciones y 4x4 píxeles por celda.

Con estos datos se consideró que para las próximas pruebas se debía seguir en la línea de mantener el resize en 200x200 píxeles, y que se pasaría a emplear al HOG con 10 orientaciones y 8x8 píxeles por celda.

Experimento 8

Una vez se han obtenido los parámetros óptimos del HOG, así como el resize de las imágenes se va a proceder a encontrar el modelo SVM más óptimo para la clasificación de los datos.

Esto se va a realizar mediante el Grid Search. El cual mediante una serie de parámetros se va a poder hacer una serie de pruebas secuenciales en el que se va a obtener los mejores hiperparámetros del modelo.

Además del Grid Search se va a seguir la técnica de K-Fold, más en concreto, se va a usar un k-fold de 5 iteraciones.

Para la elección de las diferentes iteraciones que realiza el Grid Search se ha tenido en cuenta que hay una serie de hiperparámetros que en algunos kernels no aportan ninguna mejora. Eso es debido a que el kernel no lo utiliza. Por ejemplo, el kernel “linear” no utiliza el hiperparámetro “gamma”. Es por eso que nos queda lo siguiente:

```

parametros_grid_linear = {
    'kernel': ['linear'],
    'C': [0.1, 1, 10, 100],
}

parametros_grid_rbf = {
    'kernel': ['rbf'],
    'C': [0.1, 1, 10, 100],
    'gamma': ['scale', 'auto', 1, 0.1],
}

parametros_grid_poly = {
    'kernel': ['poly'],
    'C': [0.1, 1, 10, 100],
    'degree': [2, 3, 4],
    'coef0': [0.0, 1.0, 2.0],
}

parametros_grid_sigmoid = {
    'kernel': ['sigmoid'],
    'C': [0.1, 1, 10, 100],
    'coef0': [0.0, 1.0, 2.0],
    'gamma': ['scale', 'auto', 1, 0.1],
}

```

Los mejores hiperparámetros obtenidos mediante esta técnica son los siguientes:

- kernel: rbf
- C: 10
- gamma: scale

Con un accuracy del 64.33%.

Experimento 9

Gracias a los datos obtenidos del Experimento 9, se tenía una dirección clara de qué parámetros emplear y con qué valores en el paso anterior a la validación final. Además, viendo que no había cambios significativos en el valor del Accuracy, se decidió considerar el uso de F1 como la métrica principal a tener en cuenta debido a que es la que nos permite discernir, con mayor claridad, si se hace una clasificación acertada o no en cuanto a la detección de clases, ya que el Accuracy se refiere más a la exactitud del modelo en sí.

Se pasó, por consiguiente, a realizar la ejecución de obtención de características con HOG, esta vez junto con LBP y GLCM, empleando un K-Fold de $k=5$ junto con Grid Search empleando la combinación de Kernels antes indicado. El mejor resultado obtenido de este experimento fue el Kernel Lineal, con un valor de $C=0.1$, y con un valor de $F1 = 65.33\%$.

Classification Report			
Coast	0.60	0.80	0.69
Forest	0.78	1.00	0.88
Highway	0.94	0.80	0.86
Insidecity	0.52	0.76	0.62
Mountain	0.60	0.60	0.60
Office	0.83	0.75	0.79
OpenCountry	0.57	0.55	0.56
Street	0.86	0.79	0.83
Suburb	0.90	0.86	0.88
TallBuilding	0.55	0.63	0.59
bedroom	0.53	0.40	0.46
industrial	0.50	0.32	0.39
kitchen	0.50	0.73	0.59
livingroom	0.48	0.53	0.50
store	0.65	0.46	0.54
accuracy	0.65	0.65	0.65
macro avg	0.65	0.67	0.65
	precision	recall	f1-score

Validación

Tras obtener los mejores parámetros contemplados para la métrica F1, se concluyó que se había realizado un curso de experimentación adecuado y que los resultados de los mismos eran satisfactorios, por lo que finalmente se procedió a validar el modelo con las muestras de testeo disponibles.

Para esta ejecución se ha hecho un resize de las imágenes a 200x200 píxeles, se ha usado HOG, con LBP y GLCM para la obtención de características, uso de SVM con Kernel Lineal, con un valor de $C=0.1$.

Como resultado, se obtuvo un valor de $F1=67.83\%$

Classification Report			
0	0.51	0.54	0.53
1	0.70	0.80	0.75
2	0.75	0.91	0.82
3	0.75	0.74	0.74
4	0.50	0.45	0.48
5	0.62	0.65	0.63
6	0.47	0.52	0.49
7	0.61	0.56	0.58
8	0.73	0.67	0.70
9	0.81	0.80	0.80
10	0.64	0.65	0.64
11	0.61	0.56	0.58
12	0.86	0.74	0.79
13	0.90	0.86	0.88
14	0.71	0.68	0.69
accuracy	0.68	0.68	0.68
macro avg	0.68	0.68	0.67
	precision	recall	f1-score

Conclusiones

A lo largo del desarrollo de este trabajo se ha percibido que para aportar una solución a un problema mediante el uso del aprendizaje automático, se ha de poder contemplar variedad de escenarios y posibilidades, haciendo numerosas pruebas y combinaciones de patrones y conceptos que se alineen o puedan adaptarse al problema a resolver. En concreto para este caso, el mero hecho de encontrar una manera de tratar las imágenes para su correcto procesamiento ya es un paso a tener en cuenta y que requiere de ensayo y error para no perder información y, a la vez, trabajar con la mínima indispensable posible.

Por otro lado, ha sido interesante ver los cambios que se veían reflejados en los resultados según los métodos de características que se usaban, a la vez de si se realizaban combinaciones o procesamiento de los datos como el uso de la clase Standard Scaler.

Otro aspecto a tener en cuenta es que el tiempo de dedicación orientado únicamente a la búsqueda de parámetros óptimos puede llegar a ser altamente costoso y requiere de una planificación, estudio, y baraje de parámetros previo para aprovecharse mínimamente.

Finalmente, en cuanto al resultado obtenido de la validación con datos de testeo se refiere, se considera que se trata de una puntuación satisfactoria, tanto a nivel global para, como a nivel de los resultados de clasificación concretos de cada clase. Además de ello, al poder ver las demás métricas aparte de F1, se aprecia una alta similitud entre los valores de las métricas a nivel global, y se considera que parte de ello se debe a que a lo largo de todo el proceso de afinación del modelo y de los datos, se tuvo el enfoque primeramente en el Accuracy, y posteriormente en el F1.

Trabajo en pareja

Este trabajo ha sido realizado en pareja por los autores del documento coordinando tareas trabajadas de manera individual y conjunta. Concretamente:

Joan Alcover:

- Programación y ejecución de secuencia de experimentos: HOG, LBP, GLCM, K-Fold, Grid Search, entrenamiento final y validación final.

Antonio Pujol:

- Programación y ejecución de secuencia de experimentos: Sin métodos de extracción de características, HOG.

Conjuntamente:

- Programación del HOG.
- Diseño de pruebas y secuencias de experimentos.
- Recopilación de resultados.
- Documentación.