

Entrega final: detección de objetos en imágenes

Modelos Estadísticos para Regresión y Clasificación

2023

1. Introducción

El objetivo del trabajo es aplicar las técnicas de clasificación vistas en el curso al problema de *detección de objetos* en imágenes. Concretamente nos enfocaremos en la *detección de rostros*, ver Figura 1 (Derecha).

	0	5	10	15															
0	154	224	240	244	245	247	246	247	249	247	247	243	221	187	171	156	122	81	58
	136	213	176	201	229	232	218	209	213	215	210	199	188	115	67	67	67	58	53
	120	159	204	107	137	131	84	45	75	81	73	92	141	107	58	47	73	80	64
	124	126	140	153	69	130	84	54	182	233	154	108	91	51	88	74	105	68	62
	120	85	47	59	43	51	41	85	114	232	220	138	46	46	90	86	46	63	
5	93	82	75	65	56	52	55	65	188	254	253	212	62	51	55	60	60	56	62
	134	211	234	209	168	161	180	197	231	253	252	225	147	130	129	127	148	107	102
	206	240	243	249	248	247	247	196	243	253	253	225	116	226	240	241	225	219	209
	216	236	246	246	249	243	211	137	246	251	252	233	86	132	187	211	203	180	187
	177	233	245	239	226	186	104	181	227	235	239	208	112	79	144	165	172	178	145
10	130	183	200	166	149	119	107	144	85	71	68	50	55	61	67	73	85	88	88
	84	103	130	157	154	141	185	75	84	37	37	58	48	67	61	62	61	62	63
	71	75	91	147	151	183	233	172	72	40	37	43	63	98	72	60	56	56	58
	70	76	93	156	171	194	212	184	96	65	57	59	68	115	97	68	58	58	58
	73	89	104	174	126	78	92	145	137	98	72	58	63	57	53	67	63	60	59
15	70	88	107	100	49	42	57	57	57	57	57	57	57	57	57	57	57	57	57
	68	84	88	73	37	31	49	66	61	41	31	38	44	49	52	59	61	57	58
	63	68	69	67	88	163	161	117	117	98	69	59	62	76	76	60	55	52	57
	62	64	64	73	137	205	158	79	32	44	42	46	59	79	83	60	52	51	58

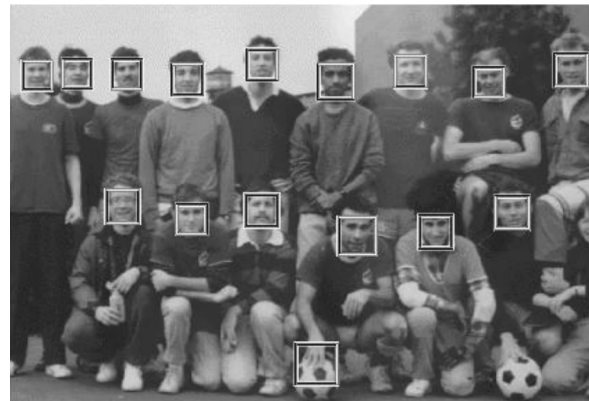


Figura 1: A la izquierda, una imagen monocolor es una matriz 2D. Las entradas de la matriz corresponden a los valores de los píxeles que van de 0 a 255. A la derecha, ejemplo de detección de rostros en imágenes. Extraído del artículo de Viola y Jones [1].

2. Imágenes y features

Nosotros trabajaremos con imágenes monocolor que suelen representarse en una escala de grises como se muestra en la Figura 1 (Izquierda). De este modo una imagen es una matriz $\mathbf{x} = [x_{ij}]_{i=0, j=0}^{i=H-1, j=W-1}$ con H (height) filas y W (width) columnas. Las entradas x_{ij} corresponden a los valores de los píxeles. En el caso monocolor es usual que el rango de valores consista en enteros que van desde 0 (negro) hasta el 255 (blanco). Pero es común también que dichos valores estén normalizados para que su rango sea entre 0 (negro) y 1 (blanco).

No trabajaremos con los píxeles directamente, sino que nuestro enfoque se basará en la extracción de *features*. Para nosotros una feature es una *función* $f : \mathbb{R}^{H,W} \rightarrow \mathbb{R}$. Por ejemplo, una feature

f puede estar relacionada con la detección de bordes en una determinada región de la imagen, en cuyo caso valores extremos de $f(\mathbf{x})$ indican la presencia de dicha característica para \mathbf{x} .

3. Detectores de rostros basados en clasificadores

Construiremos detectores de rostros a partir de clasificadores. Si disponemos de un clasificador $h : \mathbb{R}^{H,W} \rightarrow \{0,1\}$ para predecir si una imagen es un rostro o no, es decir $h(\mathbf{x}) = 1$ cuando \mathbf{x} es un rostro, podemos construir un detector de rostros escaneando todas las sub-imágenes de la imagen original. Por esta razón nuestro *principal objetivo* será elaborar clasificadores que permitan detectar si una imagen es un rostro.

Para esto dispondremos de un conjunto de datos de entrenamiento $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ en donde $\mathbf{x}_i \in \mathbb{R}^{W,H}$ es una imagen e $y_i \in \{0,1\}$ es la etiqueta que indica con 1 si \mathbf{x}_i es un rostro. De las N imágenes, p de ellas son rostros (ejemplos positivos) y n son background (ejemplos negativos).



Figura 2: Muestra del conjunto de rostros disponibles Labeled Faces in the Wild.

Como mencionamos antes, los clasificadores deberán trabajar con features extraídas de las imágenes, y no con los datos brutos representados por la matriz de píxeles. Para esto construiremos

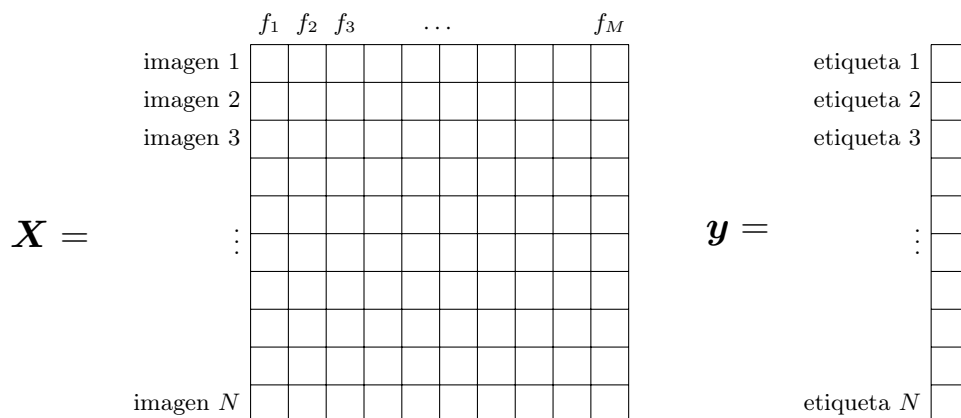
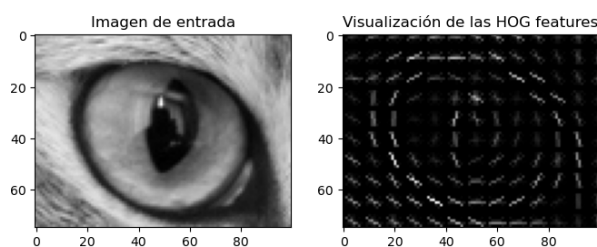


Figura 3: El par (\mathbf{X}, \mathbf{y}) en donde \mathbf{X} es la matriz de features e \mathbf{y} es el vector de etiquetas.

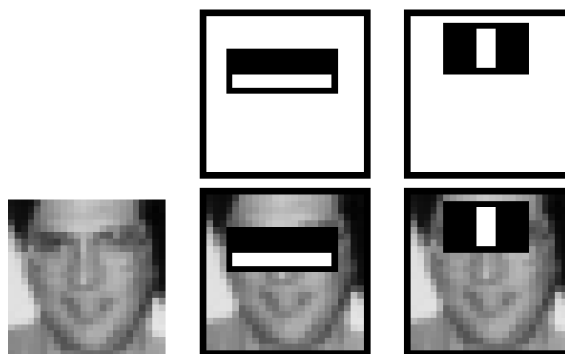
una *matriz de features* \mathbf{X} cuyas filas representan imágenes y columnas features. Debemos elegir una cantidad M de features $\{f_j\}_{j=1}^M$ de forma tal que la entrada i, j de la matriz \mathbf{X} es $f_j(\mathbf{x}_i)$. El par (\mathbf{X}, \mathbf{y}) , en donde \mathbf{y} es el vector de etiquetas, servirá de input a los clasificadores, ver la Figura 3.

Utilizaremos principalmente:

- **Histogram of Oriented Gradients (HOG)**



- **Bases de Haar**



4. Entrega y tareas a desarrollar

Fecha de entrega: Lunes 11/12/2023

Se deberá entregar un documento que contenga un **informe/reporte** de los resultados obtenidos detallando las diferentes experimentaciones realizadas. El **código** implementado deberá ser entregado también, en una notebook de Python.

Se deberán desarrollar las siguientes tareas que deberán estar incuídas en el informe:

- **Preprocesamiento:** definir y explicar detalladamente el tratamiento inicial de las imágenes, como pueden ser la normalización, el rescalado y la aplicación de transformaciones en general. Dado que las imágenes de fondo deben ser creadas, indicar y justificar la proporción de imágenes de fondo en relación a la de rostros con la cual se decide trabajar.
- **Extracción de features:** definir y explicar detalladamente los procesos de extracción de features utilizados y la construcción de la matriz de features.
- **Clasificadores:** implementar los algoritmos de clasificación vistos en el curso. A modo de ejemplo:
 - K vecinos más cercanos
 - Árboles de decisión

- Regresión logística
 - Random Forest
 - Boosting
- **Evaluación de modelos:** se deberá detallar la evaluación de los diversos clasificadores utilizando las técnicas de validación y evaluación de modelos vistas en el curso. Para ello utilizar las diferentes métricas relevantes al problema de clasificación (accuracy, precision, recall, TPR, FPR, curvas ROC, AUC, balanced accuracy, etc). Se deberá resumir los resultados de los experimentos en tablas que permitan recopilar la información de cada modelo evaluado y sus respectivas métricas.
 - **Modelo final:** se deberá elegir un modelo final. El informe deberá contener la justificación de la elección de dicho modelo. El mismo deberá adjuntarse a la entrega.

Referencias

- [1] Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*. Vol. 1. Ieee, 2001.
- [2] Shalev-Shwartz, Shai, and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014. Chapter 10.4.
- [3] Schapire, Robert E., and Yoav Freund. "Boosting: Foundations and algorithms." *Kybernetes 42.1 (2013)*: 164-166. Chapter 3.4.3.
- [4] Baumann, Florian, et al. "Cascaded random forest for fast object detection." *Image Analysis: 18th Scandinavian Conference, SCIA 2013, Espoo, Finland, June 17-20, 2013. Proceedings 18*. Springer Berlin Heidelberg, 2013.
- [5] [Detecting Faces \(Viola Jones Algorithm\) - Computerphile](#)