


## Bloc 2

### Aprenentatge Automàtic

## Pràctica 2:

### Sessió 3

## Aplicació dels algoritmes de Perceptró i Regressió Logística a diversos conjunts de dades

**DOCENCIA VIRTUAL**

**Responsable del Tratamiento:** Universitat Politècnica de València (UPV)

**Finalidad:** Prestación del servicio público de educación superior en base al interés público de la UPV (Art. 6.1.e del RGPD).

**Ejercicio de derechos y segunda capa informativa:** Podrán ejercer los derechos reconocidos en el RGPD y la LOPDGD de acceso, rectificación, oposición, supresión, etc., escribiendo al correo [dpd@upv.es](mailto:dpd@upv.es).

Para obtener más información sobre el tratamiento de sus datos puede visitar el siguiente enlace: <https://www.upv.es/contenidos/DPD>.

**Propiedad Intelectual:** Uso exclusivo en el entorno del aula virtual. Queda prohibida la difusión, distribución o divulgación de la grabación de las clases y particularmente su compartición en redes sociales o servicios dedicados a compartir apuntes. La infracción de esta prohibición puede generar responsabilidad disciplinaria, administrativa y/o civil.

## Sessions de la pràctica 2

### Sessió 1:

- Familiaritzar-se amb l'entorn de treball (Google Colab)
- Analitzar conjunts de dades (datasets): iris, digits, olivetti, openml

### Sessió 2:

- Aplicació de l'algorisme del Perceptró a tasques de classificació: conjunt de dades iris.
- **Exercici**: Aplicar el Perceptró a digits i olivetti.

### Sessió 3:

- Aplicació de la Regressió Logística a tasques de classificació: conjunt de dades iris.

#### Exemple d'examen:

- Aplicació de Regressió Logística a un conjunt de dades de OpenML.

### Sessió 4 (**examen**):

- Es demanarà l'aplicació del Regressió Logística per a una tasca diferent d'OpenML.
- Caldrà pujar la solució de **l'Exercici**.

## Sessió 3: Regressió Logística (RL)

- Anem a utilitzar la regressió logística aplicada a tasques de classificació.
  - Emprarem la funció `LogisticRegression` de `sklearn`.

**Aprenentatge per mínima NLL:** l'aprenentatge per màxima versemblança plantejat com un problema de minimització:

$$\mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmin}} \operatorname{NLL}(\mathbf{W})$$

**Descens per gradient aplicat a la regressió logística**

$$\mathbf{W}_0 = \mathbf{0}; \quad \mathbf{W}_{i+1} = \mathbf{W}_i - \eta_i \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n (\mu_n - y_n)^t$$

## Sessió 3: Regressió Logística (RL)

### En sklearn:

```
LogisticRegression(random_state=23,  
                    solver=solver,  
                    tol=tol,  
                    C=C,  
                    max_iter=max_iter).fit(X_train, y_train)
```

On:

- L'argument "solver" especifica quin algorisme utilitzar per a l'optimització dels paràmetres del model.
- L'argument "tol" especifica la tolerància per a la detenció del procediment d'optimització.
- L'argument "C" és l'invers de la força de regularització. La regularització pot ajudar a prevenir el sobreajust reduint la magnitud dels paràmetres del model.
- "Max\_iter" especifica el nombre màxim d'iteracions que el solucionador ha d'utilitzar per trobar els paràmetres del model.

## Sessió 3: Regressió Logística (RL)

### Nocions importants:

#### Subajust (Underfitting):

- Es produeix quan el model no aconsegueix captar la complexitat del conjunt de dades.
- Això passa perquè el model és massa simple per representar les dades.
- Genera un percentatge d'error alt tant en entrenament com en test.

#### Sobreajust (Overfitting):

- Es produeix quan el model aprèn no només els patrons generals de les dades, sinó també el soroll i les irregularitats del conjunt d'entrenament.
- El model podria tenir dificultats per classificar correctament, ja que només "recorda" les dades d'entrenament.
- Genera un percentatge d'error baix en entrenament però alt en test.

## Sessió 3: Regressió Logística (RL)

### Solver

La funció LogisticRegression de sklearn ofereix la possibilitat d'utilitzar diversos solvers:

**liblinear:** Una bona opció per a conjunts de dades xicotets; es limita a resoldre problemes d'una classe front a la resta, no per a determinar la probabilitat de cada classe.

**sag, saga:** Bones opcions per a conjunts de dades grans.

**sag, saga, lbfgs, newton-cg:** Només aquests solvers gestionen la pèrdua multinomial per a problemes multi-classe.

**newton-cholesky:** Bona opció quan el nombre de mostres  $\gg$  nombre de característiques; es limita a resoldre problemes de classificació binària i problemes d'una classe front a la resta.

## Sessió 3: Regressió Logística (RL)

### Solver (RECOMANACIONS)

Utilitzar **sag**, **saga**, **lbfgs**, **newton-cg**, que són per a problemes multi-classe.

La llibreria sklearn utilitza **lbfgs** per defecte.

Es pot prioritzar **sag** o **saga** per a conjunts de dades grans

## Sessió 3: Regressió Logística (RL)

### Regularització

S'afegeix una penalització per evitar el sobreajustament (overfitting), és a dir, per aconseguir que l'algorisme generalitze bé en conjunts de prova.

Penalitz els coeficients alts de la funció lineal (sense incloure el terme independent).

Quan una característica de les mostres només ocorre en una classe (i en la resta de classes els valors d'aquesta característica són 0), l'algorisme de regressió logística assignarà coeficients molt alts a aquesta característica. En aquest cas, és probable que el model s'ajuste massa al conjunt d'entrenament.

$$\operatorname{argmin}_W \text{NLL}(\mathbf{W}) + \lambda R(\mathbf{W})$$

paràmetre  $\lambda \in \mathbb{R}^{>0}$


terme de regularització



## Sessió 3: Regressió Logística (RL)

### Regularització

Hi ha diversos tipus de regularització  $R(\mathbf{W})$ . La llibreria sklearn aplica per defecte una regularització anomenada regularització L2 o de Gauss.

$\lambda = \frac{1}{C}$   paràmetre C que utilitza la funció LogisticRegression de sklearn

- Per defecte,  $C=1.0$ .
- Per valors propers a 0,  $\lambda$  és molt gran i, per tant, s'aplica la màxima regularització, la qual cosa pot implicar una possibilitat de subajustament en el conjunt d'entrenament.
- Per valors molt alts de C,  $\lambda$  és molt xicotet i, per tant, s'aplica la mínima regularització, la qual cosa pot implicar una possibilitat de sobreajustament en el conjunt d'entrenament.
- $C \rightarrow 0$  puede provocar **subajuste**
- $C \rightarrow$  valores altos puede provocar **sobreajuste**

## Sessió 3: Regressió Logística (RL)

### Tolerància

Umbral de tolerància" (tol) per al criteri d'aturada (per acabar l'entrenament)

L'algorisme de RL deixarà de buscar un mínim un cop s'arriba a certa tolerància, és a dir, quan es troba prou a prop de l'objectiu.

Per defecte,  $\text{tol} = 0.0001$  ( $1e^{-4}$ )

### Early stopping

Una forma indirecta d'aplicar "regularització" és controlant el nombre **màxim d'iteracions** que s'executa l'algorisme de RL.

## Sessió 3

### Exemple en Iris (irisRL.ipynb)

- Aplica l'algorisme de Regressió Logística al conjunt de dades Iris amb diferents valors per als paràmetres de solver, tolerància, C i max\_iter.
- Mostra els resultats de la següent manera:

#	Iter	solver	C	tol	Ete
#					
	10	lbfgs	0.010	0.0001	0.133
	10	lbfgs	0.010	0.0100	0.133
	10	lbfgs	0.010	1.0000	0.133
	10	lbfgs	0.010	100.0000	0.600
	10	lbfgs	0.010	10000.0000	0.600
#					
	10	lbfgs	0.100	0.0001	0.000
	10	lbfgs	0.100	0.0100	0.000
	10	lbfgs	0.100	1.0000	0.000
	10	lbfgs	0.100	100.0000	0.600
	10	lbfgs	0.100	10000.0000	0.600
#					
	10	lbfgs	1.000	0.0001	0.000
	10	lbfgs	1.000	0.0100	0.000
	10	lbfgs	1.000	1.0000	0.000
	10	lbfgs	1.000	100.0000	0.600
	10	lbfgs	1.000	10000.0000	0.600
#					
	10	lbfgs	10.000	0.0001	0.000
	10	lbfgs	10.000	0.0100	0.000
	10	lbfgs	10.000	1.0000	0.000
	10	lbfgs	10.000	100.0000	0.600
	10	lbfgs	10.000	10000.0000	0.600
#					
	10	lbfgs	100.000	0.0001	0.000
	10	lbfgs	100.000	0.0100	0.000
	10	lbfgs	100.000	1.0000	0.000

- Indicar quina creus que és la millor combinació d'iteracions, solver, C i tolerància.

## Sessió 3

### Exemple d'examen

- Aplicar l'algorisme RL a un conjunt de dades d'OpenML per a diferents valors de **solver**, tolerància, C i **max\_iter**
  - Consultar **01\_exemple\_examen.ipynb**

#### Com serà l'examen?

- Mateixa idea de l'exemple.
- Es proporcionarà un conjunt de dades identificat amb el seu ID.
- Es demanarà executar l'algorisme RL per a diferents valors de solver, tolerància, C i max\_iter i observar l'error d'entrenament i de test.
- Es plantejarà alguna qüestió respecte a quins paràmetres utilitzar.


## Bloc 2

### Aprenentatge Automàtic

## Pràctica 2:

### Sessió 3

## Aplicació dels algoritmes de Perceptró i Regressió Logística a diversos conjunts de dades

**DOCENCIA VIRTUAL**

**Responsable del Tratamiento:** Universitat Politècnica de València (UPV)

**Finalidad:** Prestación del servicio público de educación superior en base al interés público de la UPV (Art. 6.1.e del RGPD).

**Ejercicio de derechos y segunda capa informativa:** Podrán ejercer los derechos reconocidos en el RGPD y la LOPDGGDD de acceso, rectificación, oposición, supresión, etc., escribiendo al correo [dpd@upv.es](mailto:dpd@upv.es).

Para obtener más información sobre el tratamiento de sus datos puede visitar el siguiente enlace: <https://www.upv.es/contenidos/DPD>.

**Propiedad Intelectual:** Uso exclusivo en el entorno del aula virtual. Queda prohibida la difusión, distribución o divulgación de la grabación de las clases y particularmente su compartición en redes sociales o servicios dedicados a compartir apuntes. La infracción de esta prohibición puede generar responsabilidad disciplinaria, administrativa y/o civil.