Aprendizaje Automático

Departamento de Informática – UC3M

TUTORIAL 3 – Búsqueda hiperparámetros en regresiones

2025

Tutorial 3

Recordando teoría

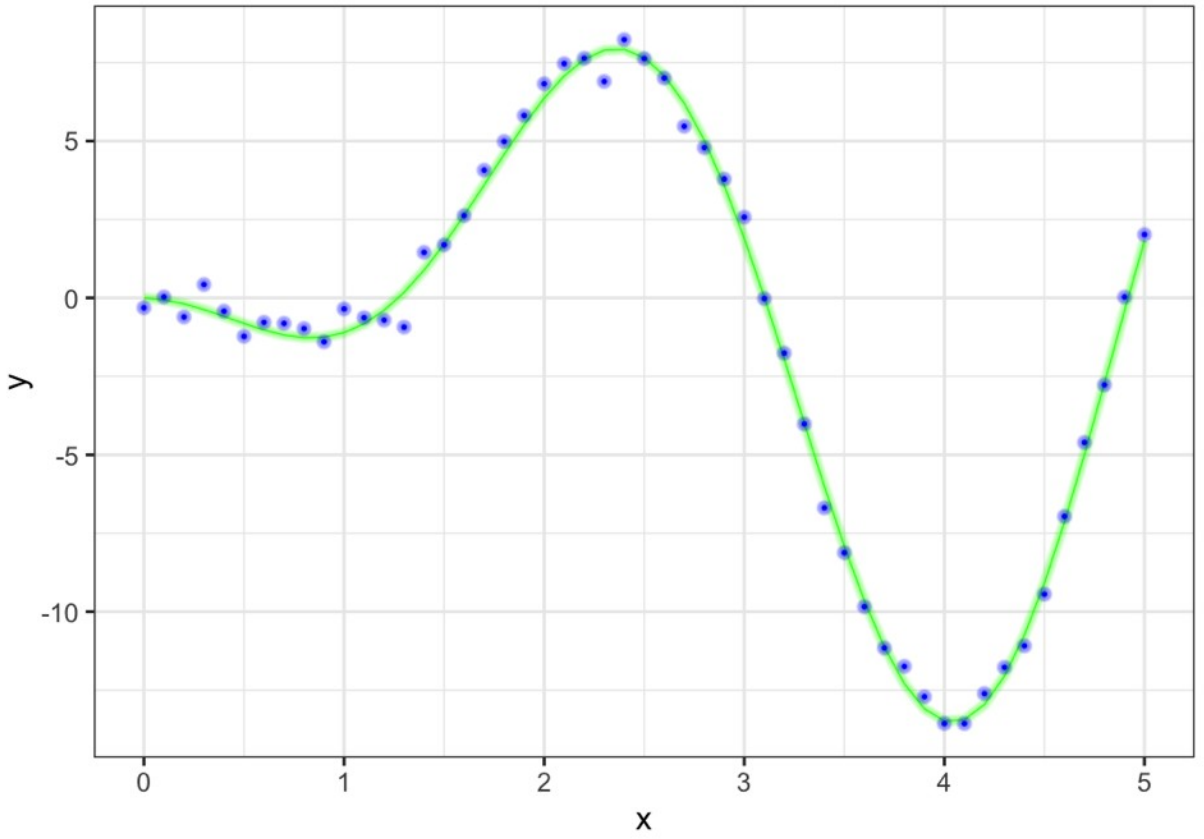
- Los datos son los puntos azules
  - x es una secuencia uniforme, $x = 0, 0.1, 0.2, \ldots, 5$
  - y puede expresarse como:

$$y(x) = f(x) + \boldsymbol{\varepsilon}$$

donde $\boldsymbol{\varepsilon}$ es ruido

- En verde se muestra la curva “real” que no conocemos
- **Objetivo**

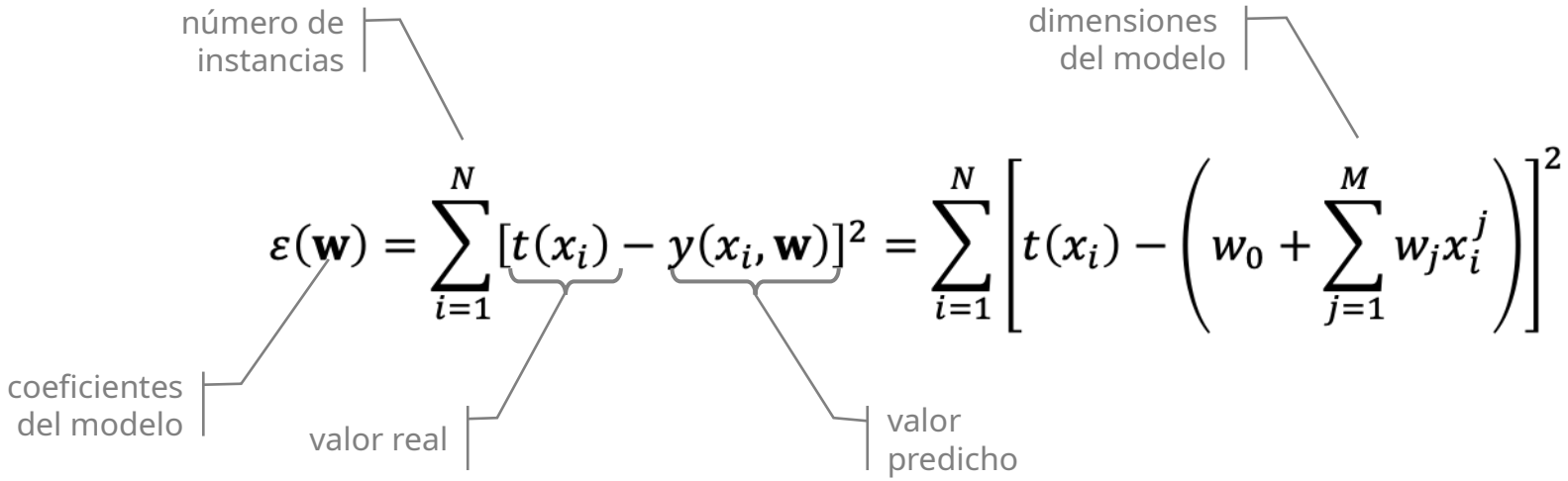
  Ajustar una curva a los puntos



Regresión

Regularización

- Objetivo
  Ajustar automáticamente la complejidad del modelo
- El error, **ε**, la suma de los errores al cuadrado, residuos al cuadrado



Tutorial 3

Recordando teoría. Regularización

En sklearn λ es α

- **Ridge**, o ℓ_2

$$\hat{l}(\mathbf{w}) = \varepsilon(\mathbf{w}) + \lambda \mathbf{w}^{\mathbf{T}}\mathbf{w} = \varepsilon(\mathbf{w}) + \lambda \sum_{j=1}^{M} w_j^2$$

- **Lasso**, o ℓ_1

$$\hat{l}(\mathbf{w}) = \varepsilon(\mathbf{w}) + \lambda \sum_{j=1}^{M} |w_j|$$

- **Elastic net**, ℓ_1 y ℓ_2 , se realiza una combinación de ambas.

En sklearn α es l1_ratio.

$$\hat{l}(\mathbf{w}) = \varepsilon(\mathbf{w}) + \lambda \left[(1-\alpha) \sum_{j=1}^{M} w_j^2 + \alpha \sum_{j=1}^{M} |w_j| \right]$$

Tutorial 3

Determinants of Wages from the 1985 Current Population Survey

- **EDUCATION**: Number of years of education.
- **SOUTH**: Indicator variable for Southern Region (1=Person lives in South, 0=Person lives elsewhere).
- **SEX**: Indicator variable for sex (1=Female, 0=Male).
- **EXPERIENCE**: Number of years of work experience.
- **UNION**: Indicator variable for union membership (1=Union member, 0=Not union member).
- **WAGE**: Wage (dollars per hour).
- **AGE**: Age (years).
- **RACE**: Race (1=Other, 2=Hispanic, 3=White).
- **OCCUPATION**: Occupational category (1=Management, 2=Sales, 3=Clerical, 4=Service, 5=Professional, 6=Other).
- **SECTOR**: Sector (0=Other, 1=Manufacturing, 2=Construction).
- **MARR**: Marital Status (0=Unmarried, 1=Married)

Tutorial 3

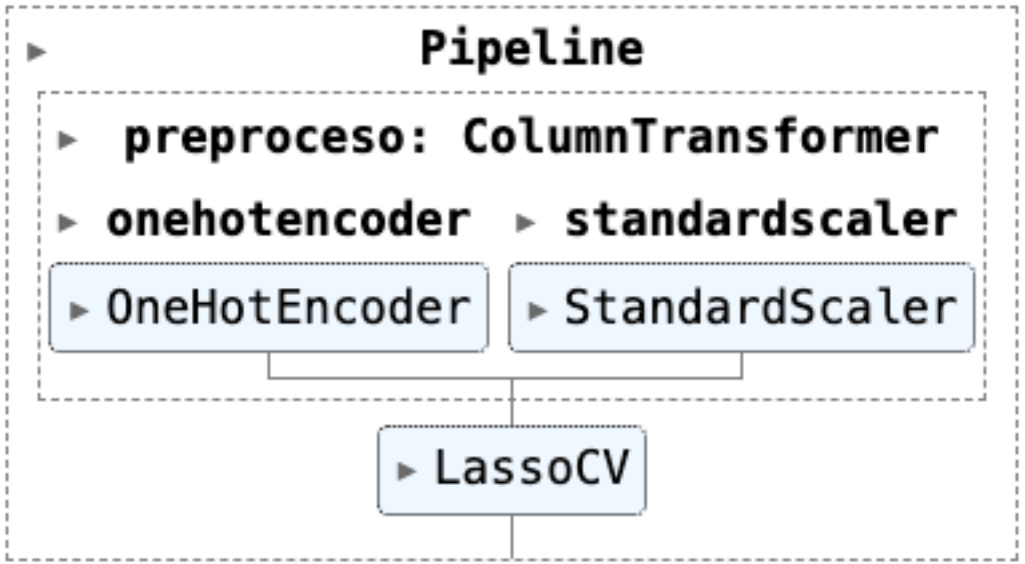
ColumnTransformer

```
categorical_columns = X_train.select_dtypes(include="category").columns
numerical_columns = X_train.select_dtypes(exclude="category").columns

# Otra forma
# categorical_columns = ["RACE", "OCCUPATION", "SECTOR", "MARR", "UNION", "SEX", "SOUTH"]
# numerical_columns = ["EDUCATION", "EXPERIENCE", "AGE"]

preprocessor = make_column_transformer(
                    (OneHotEncoder(drop="if_binary"), categorical_columns),
                    (StandardScaler(), numerical_columns))

# Otra forma
# preprocesor = ColumnTransformer(
#                     [('categoricos', OneHotEncoder(drop="if_binary"),categorical_columns),
#                      ('numericos', StandardScaler(), numerical_columns)])
```

Tutorial 3

Pipeline

```
pipe_regrLasso = Pipeline([
                    ('preproceso', preprocessor),
                    ('regresor', LassoCV(
                                    alphas = np.logspace(-9, 3, 200),
                                    cv = 3))
                ])

np.random.seed(42)
pipe_regrLasso.fit(X = X_train, y = y_train)
```



Tutorial 3

Ridge/Rasso/Elastic Net