Esame 2020

Machine Learning

SUPSI – Primavera 2020

Attenzione: punti aggiuntivi saranno accreditati in ogni esercizio dell'esame per l'implementazione di grafici di analisi preliminare dei dati, plots di visualizzazione dei risultati, commenti ed osservazioni, calcolo di indicatori aggiuntivi di qualità, ecc.
In sintesi, potete aggiungere a corredo dell'esercizio che state svolgendo qualsiasi elemento in più che vi sembra interessante e, se corretto, sarà valutato positivamente.

Task I.1. **[30 punti]** Nel portale iCorsi, nell'ultima sezione, trovate un dataset salaryData. Il dataset è molto semplice: ogni istanza dice a quanto corrisponde il salario annuale di un individuo a seconda degli anni di esperienza nel settore in cui lavora (in verità, i numeri non sono molto realistici). Si chiede di:

(a) Fare uno split del dataset in 70% training set e 30% validation set.

(b) Costruire un regressore lineare su questo training set.

(c) Fare predizioni sul validation set, calcolare il valore di R^2 e una metrica di qualità a scelta (MSE, RMSE, ecc.).

(d) Implementare allo stesso modo una ridge regression con cross-validation sull'iperparametro. Calcolare il nuovo valore di R^2 e il nuovo valore della metrica di qualità scelta (MSE, RMSE, ecc.).

Task I.2. **[30 punti]** Generate una funzione di tipo $y = x^3$ nell'intervallo $[-5, 5]$. Aggiungete del rumore di tipo Gaussiano (random.normal) con valore medio $\mu = 0$ e deviazione standard $\sigma = 0.2$ a questi dati. Questo sarà il vostro dataset.

(a) Fare regressione lineare con modelli di ordine polinomiale dal grado 1 fino al grado 10.

(b) Valutare i risultati ottenuti secondo una qualche metrica di propria scelta (MSE, RMSE, ecc.).

(c) Scegliere un ordine polinomiale tra quelli appena testati e fare regressione ridge per diversi valori di α.

(d) Scegliere un ordine polinomiale tra quelli appena testati e fare regressione LASSO per diversi valori di α.

(e) Come si può mettere in luce la differenza sostanziale tra la regressione Ridge e quella LASSO? Quali sono i valori più interessanti da osservare quando si confrontano questi due regolarizzatori?

Task I.3. **[30 punti]** Usare il seguente script per generare un dataset sintetico:

```
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns; sns.set() # for plot styling
import numpy as np
from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=300, centers=4,
cluster_std=0.60, random_state=0)
plt.scatter(X[:, 0], X[:, 1], s=50)
```

Si richiede di:

(a) Fare clustering tramite $k-$Means e commentare i risultati.

Modificare poi i dati in questo modo:

```
from sklearn.datasets import make_moons
X, y = make_moons(200, noise=.05, random_state=0)
plt.scatter(X[:, 0], X[:, 1], s=50);
```

Ora $k-$Means non riesce più a individuare i due cluster, anche se questi sono visibili a occhio nudo.

(b) Come si può utilizzare $k-$Means per fare un clustering che abbia una qualche utilità? Scrivere uno script che tenti di risolvere il problema. (*Hint:* bisogna sfruttare in qualche modo (vedi slides) il valore di k)

Task I.4. **[30 punti]** Usare il seguente script per generare un dataset sintetico:

```
import numpy as np
import pandas as pd
import sklearn
import sklearn.datasets as ds
import sklearn.model_selection as ms
import sklearn.svm as svm
import matplotlib.pyplot as plt
%matplotlib inline
X = np.random.randn(200, 2)
y = X[:, 0] + X[:, 1] > 1
```

Si richiede di:

(a) Classificare i dati (l'array `y` contiene solo due possibili valori, cioè le classi) con SVM lineare.

(b) Classificare i dati con Logistic Regression.

(c) Classificare i dati con kNN (scegliere k come si ritiene migliore).

(d) Argomentare quale classificatore ha funzionato meglio.

Modificare poi i dati in questo modo:

```
y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)
```

Ora le due classi sono non separabili linearmente, ma un classificatore nonlineare potrebbe riuscire a risolvere il task. Si richiede di:

(e) Classificare i dati con SVM nonlineare (kernel RBF, quello di default, è sufficiente)

(f) Provare a classificare i dati con kNN (scegliere k come si preferisce).

(g) Argomentare quale classificatore ha funzionato meglio.

(h) **Bonus** Classificare i dati con Logistic Regression di ordine superiore al primo.