DATA MINING B(3)

11-0. Answer for 10th quiz

Multiresponse Linear Regression

- For weather data, use "temperature" and "humidity(humidity), conduct multiresponse linear regression to predict "play".

- Assign "1" to play=yes and "0" to play="no", then conduct linear regression with 2 explanatory variables "temperature" and "humidity" and an objective variable "play".

- We will get
  - y0 = -0.0011temp – 0.0163hum + 2.0584

Multiresponse Linear Regression

- Assign "0" to play=yes and "1" to play="no", then conduct linear regression with 2 explanatory variables "temperature" and "humidity" and an objective variable "play".

- We will get
  - $y1 = 0.0011\text{temp} + 0.0163\text{hum} - 1.0584$

- Compare y0 and y1 assigning explanatory variables of the instance we want to predict.

Multiresponse Linear Regression

- Classify to the class depending on the larger one, y0 and y1.
  - y0 > y1 → yes
  - y1 > y0 → no

- In this case, 3 instances classified incorrectly.

temperature	humidity	play	y0(yes=1)	y1(yes=0)	pred
85	85	no	0.575575	0.424425	yes
80	90	no	0.49933	0.50067	no
83	86	yes	0.561424	0.438576	yes
70	96	yes	0.412228	0.587772	no
68	80	yes	0.675976	0.324024	yes
65	70	no	0.84274	0.15726	yes
64	65	yes	0.925573	0.074427	yes
72	95	no	0.426379	0.573621	no
69	70	yes	0.838348	0.161652	yes
75	80	yes	0.66829	0.33171	yes
75	70	yes	0.83176	0.16824	yes
72	90	yes	0.508114	0.491886	yes
81	75	yes	0.743437	0.256563	yes
71	91	no	0.492865	0.507135	no

Logistic Regression

- Explanatory variables and subjective variable are same as linear regression.
- Conduct regression, we will obtain the logit with following coefficients and intercept.
  - coef [w1, w2] = [-0.01293111 -0.07815206]
  - Intercept w0=[8.01861614]

- Predicting with this result, we will find that 3 instances are classified incorrectly.
  - Miss-classified instances are not same as the result of linear regression.

Code: python3 with scikit-learn (0.23)

```python
1.   from sklearn.linear_model import LogisticRegression
2.   from sklearn.linear_model import LinearRegression
3.   import numpy as np
4.   import pandas as pd
5.
6.   # データを読み込む
7.   data = pd.read_csv("weather.csv")
8.
9.   #回帰分析に使うデータの指定
10.  x = data.iloc[:,[1,2]] #説明変数
11.  y = data.iloc[:,4] #目的変数

12.  y0 = y.map({"yes": 1, "no": 0})
13.  y1 = y.map({"yes": 0, "no": 1})

14.
15.  #モデルの設定
16.  ln0_model = LinearRegression()
17.  ln1_model = LinearRegression()
18.  lr_model = LogisticRegression()

19.  #回帰分析の実行
20.  ln0_res = ln0_model.fit(x,y0)
21.  ln1_res = ln1_model.fit(x,y1)
22.  lr_res = lr_model.fit(x,y0)

23.  #結果の詳細を表示
24.  print('linear yes=1 coef and inter', ln0_res.coef_, ln0_res.intercept_)
25.  print('linear yes=0 coef and inter', ln1_res.coef_, ln1_res.intercept_)
26.  print('logistic yes=1 coef and inter', lr_res.coef_, lr_res.intercept_)
27.  print('ans y yes=1', y0.values.reshape([14]))
28.  print('ans y yes=0', y1.values.reshape([14]))
29.  print('linear yes=1', np.round(ln0_model.predict(x)))
30.  print('linear yes=1', ln0_model.predict(x))
31.  print('linear yes=0',np.round(ln1_model.predict(x)))
32.  print('miss=',sum(np.logical_xor(np.round(ln0_model.predict(x)), y0)))
33.  print('logistic yes=1', lr_res.predict(x))
34.  print('miss=',sum(np.logical_xor(np.round(lr_model.predict(x)), y0)))
```