

자유형 과제: scikit-learn로 등위회귀 구현해 보기

[9주/2차시 과제내용]: scikit-learn 로 등위회귀 구현해 보기

선형회귀 및 등위회귀 구현

- 강의: `np.polyfit()`으로 선형회귀 분석하기
- 강의: 텐서플로우로 선형회귀 분석하기
- 과제 : `scikit-learn`로 등위회귀 분석하기

과제 : scikit-learn로 회귀분석 구현해 보기

- scikit-learn 패키지는 파이썬 Machine Learning (기계학습) 패키지이다.
- 설치 및 관련 API 메소드는 아래의 링크에서 확인할 수 있다.
- <https://scikit-learn.org/stable/install.html>
- 선형회귀와 등위(Isotonic) 회귀 분석의 차이점을 분석하는 과제를 수행한다.
- 선형회귀와 등위(Isotonic) 회귀 분석 코드는 아래 링크 참조.
 - https://scikit-learn.org/stable/auto_examples/plot_isotonic_regression.html#sphx-glr-auto-examples-plot-isotonic-regression-py
- 분석하는 방법은 모든 방법을 동원한다. (강의한 내용을 따라 데이터 시각화, 디버깅, 등등)

과제 내용

- IsotonicRegression () 메소드의 내용 분석 (구현)
- LinearRegression () 메소드의 내용 분석 (구현)
- check_random_state() 메소드의 내용 분석 (구현)
- 분석 방법
- 분석하는 방법은 가능한 모든 방법을 동원한다.
- 강의 내용을 따라 데이터 시각화, GU디버깅, 콘솔 디버깅, 예제 이용 등
- ```
from sklearn.linear_model import LinearRegression
from sklearn.isotonic import IsotonicRegression
from sklearn.utils import check_random_state
```

# 등위회귀분석 코드

```
print(__doc__)

Author: Nelle Varoquaux <nelle.varoquaux@gmail.com>
Alexandre Gramfort <alexandre.gramfort@inria.fr>
License: BSD

import numpy as np
import matplotlib.pyplot as plt
from matplotlib.collections import LineCollection

from sklearn.linear_model import LinearRegression
from sklearn.isotonic import IsotonicRegression
from sklearn.utils import check_random_state

n = 100
x = np.arange(n)
rs = check_random_state(0)
y = rs.randint(-50, 50, size=(n,)) + 50. * np.log1p(np.arange(n))
```

# 등위회귀분석 코드

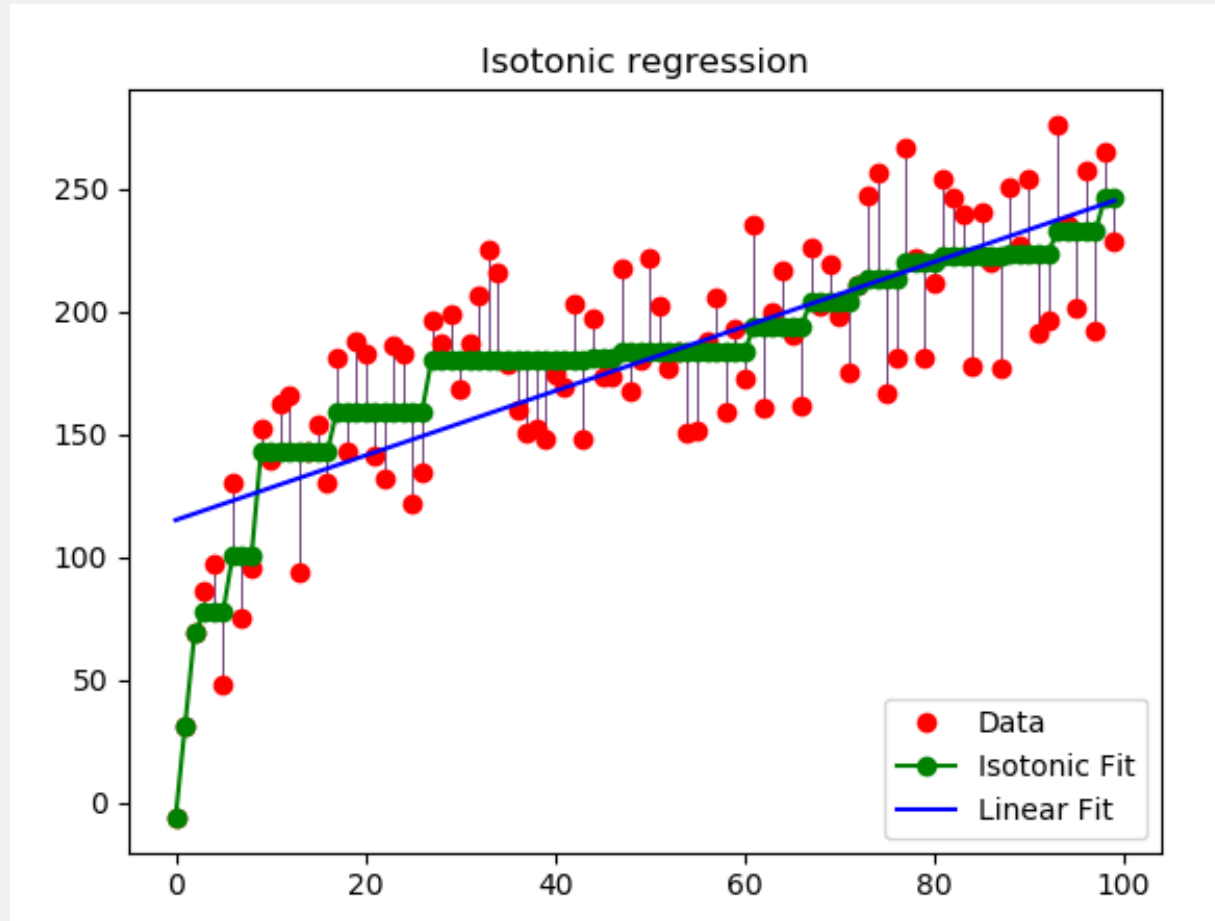
```

Fit IsotonicRegression and LinearRegression models
ir = IsotonicRegression()
y_ = ir.fit_transform(x, y)
lr = LinearRegression()
lr.fit(x[:, np.newaxis], y) # x needs to be 2d for LinearRegression

Plot result
segments = [[[i, y[i]], [i, y_[i]]] for i in range(n)]
lc = LineCollection(segments, zorder=0)
lc.set_array(np.ones(len(y)))
lc.set_linewidths(np.full(n, 0.5))
fig = plt.figure()
plt.plot(x, y, 'r.', markersize=12)
plt.plot(x, y_, 'g.-', markersize=12)
plt.plot(x, lr.predict(x[:, np.newaxis]), 'b-')
plt.gca().add_collection(lc)
plt.legend(('Data', 'Isotonic Fit', 'Linear Fit'), loc='lower right')
plt.title('Isotonic regression')
plt.show()
```

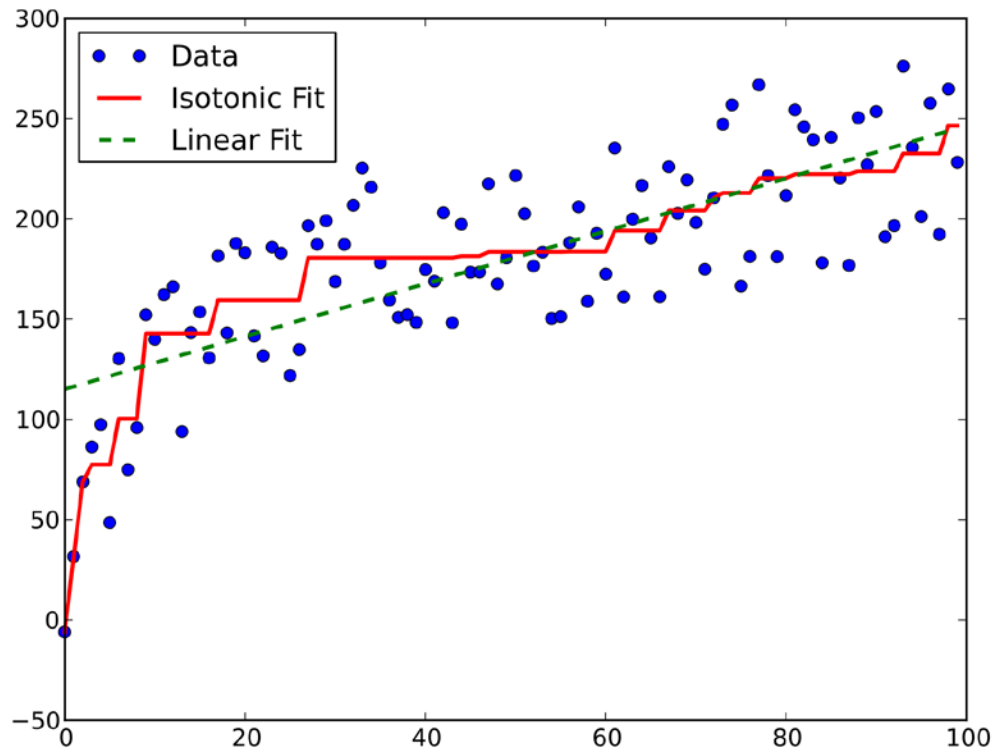
# 선형회귀 분석과 등위(Isotonic) 회귀분석 시각화

- 선형회귀 분석과 등위 (Isotonic) 회귀 분석




# 등위(Isotonic) 회귀분석 시각화

- 등위회귀의 이점은 함수가 단조 증가하는 한 선형 회귀에 의해 부과된 모든 선형 함수 형태에 의해 제약 받지 않는다.







# 등위(Isotonic) 회귀분석이란

- 등위 회귀 분석은 함수의 비 감축 근사(a non-decreasing approximation )를 찾아내는 회귀방법이다.
- 비감축 근사(a non-decreasing approximation )를 수행함으로 훈련 데이터의 평균 제곱 오차를 최소화한다.
- 이러한 모델의 이점은 선형성과 같이 대상 함수에 대한 어떠한 형식도 가정하지 않는다.

# 등위회귀분석의 정의

- 정의를 잘 분석해 보는 것이 중요하다.
- 아래의 케이스 데이터 셋에 대해서 분석을 실시하는 것이 필요하다.
- $\min \sum w_i (y[i] - y_{-}[i]) ** 2$
- subject to  $y_{-}[i] \leq y_{-}[j]$  whenever  $X[i] \leq X[j]$
- and  $\min(y_{-}) = y_{-}\min, \max(y_{-}) = y_{-}\max$ 
  - $y[i]$  are inputs (real numbers)
  - $y_{-}[i]$  are fitted
  - $X$  specifies the order. If  $X$  is non-decreasing then  $y_{-}$  is non-decreasing.
  - $w[i]$  are optional strictly positive weights (default to 1.0)

<https://scikit-learn.org/stable/modules/generated/sklearn.isotonic.IsotonicRegression.html#sklearn.isotonic.IsotonicRegression>

# 과제 제출

- 가상강의실에 제출
- 일정
  - 제출: 5월 21일 화요일 수업시작 전 (기한 엄수)
- 본 과제는 자유형 과제로 모든 학생이 제출하는 것이 아님
  - 원하는 학생만 제출함
  - 제출한 학생 중에 일부 학생에 대해 발표 기회를 제공함
  - 발표와 제출 내용이 만족할 시 보너스 성적 제공