Question 1: Fit a Linear Regression (20)

Complete each sub-questions with either codes or written answers. Make sure to print your result if necessary. We are going to use the data from Homework 1, the diabetes data.

In [23]:

```python
from sklearn.datasets import load_diabetes
data = load_diabetes()
X, y = data.data, data.target
print(data.DESCR)
```

```
.. _diabetes_dataset:

Diabetes dataset
----------------

Ten baseline variables, age, sex, body mass index, average blood
pressure, and six blood serum measurements were obtained for each of n =
442 diabetes patients, as well as the response of interest, a
quantitative measure of disease progression one year after baseline.

**Data Set Characteristics:**

  :Number of Instances: 442

  :Number of Attributes: First 10 columns are numeric predictive values

  :Target: Column 11 is a quantitative measure of disease progression one year after baseline

  :Attribute Information:
      - age     age in years
      - sex
      - bmi     body mass index
      - bp      average blood pressure
      - s1      tc, total serum cholesterol
      - s2      ldl, low-density lipoproteins
      - s3      hdl, high-density lipoproteins
      - s4      tch, total cholesterol / HDL
      - s5      ltg, possibly log of serum triglycerides level
      - s6      glu, blood sugar level

Note: Each of these 10 feature variables have been mean centered and scaled by the standard deviation times
the square root of `n_samples` (i.e. the sum of squares of each column totals 1).

Source URL:
https://www4.stat.ncsu.edu/~boos/var.select/diabetes.html

For more information see:
Bradley Efron, Trevor Hastie, Iain Johnstone and Robert Tibshirani (2004) "Least Angle Regression," Annals
of Statistics (with discussion), 407-499.
(https://web.stanford.edu/~hastie/Papers/LARS/LeastAngle_2002.pdf)
```

1a. Plot a scatter plot between the target value and BMI. Show the estimated linear regression line and describe what you have seen.

In [16]:

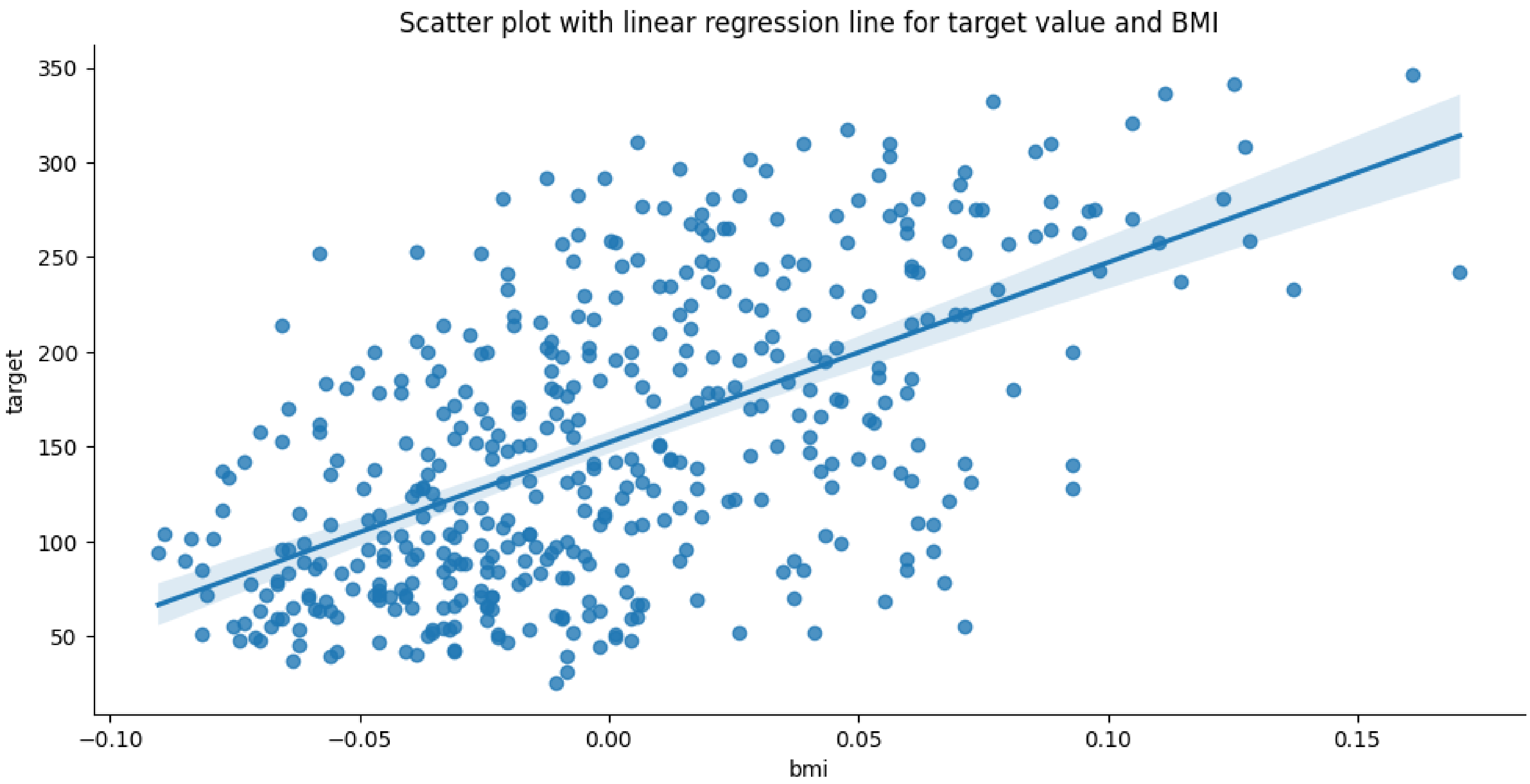
```
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt

df = pd.DataFrame(data.data, columns=data.feature_names)
df['target'] = data.target

sns.lmplot(x = 'bmi', y = 'target', data = df, aspect = 2)

plt.title('Scatter plot with linear regression line for target value and BMI')
plt.show()

# It seems like there is a correlation between BMI and the target value, as the data points follow a
# general linear increasing trend, although the data points do not completely follow the regression line.
```

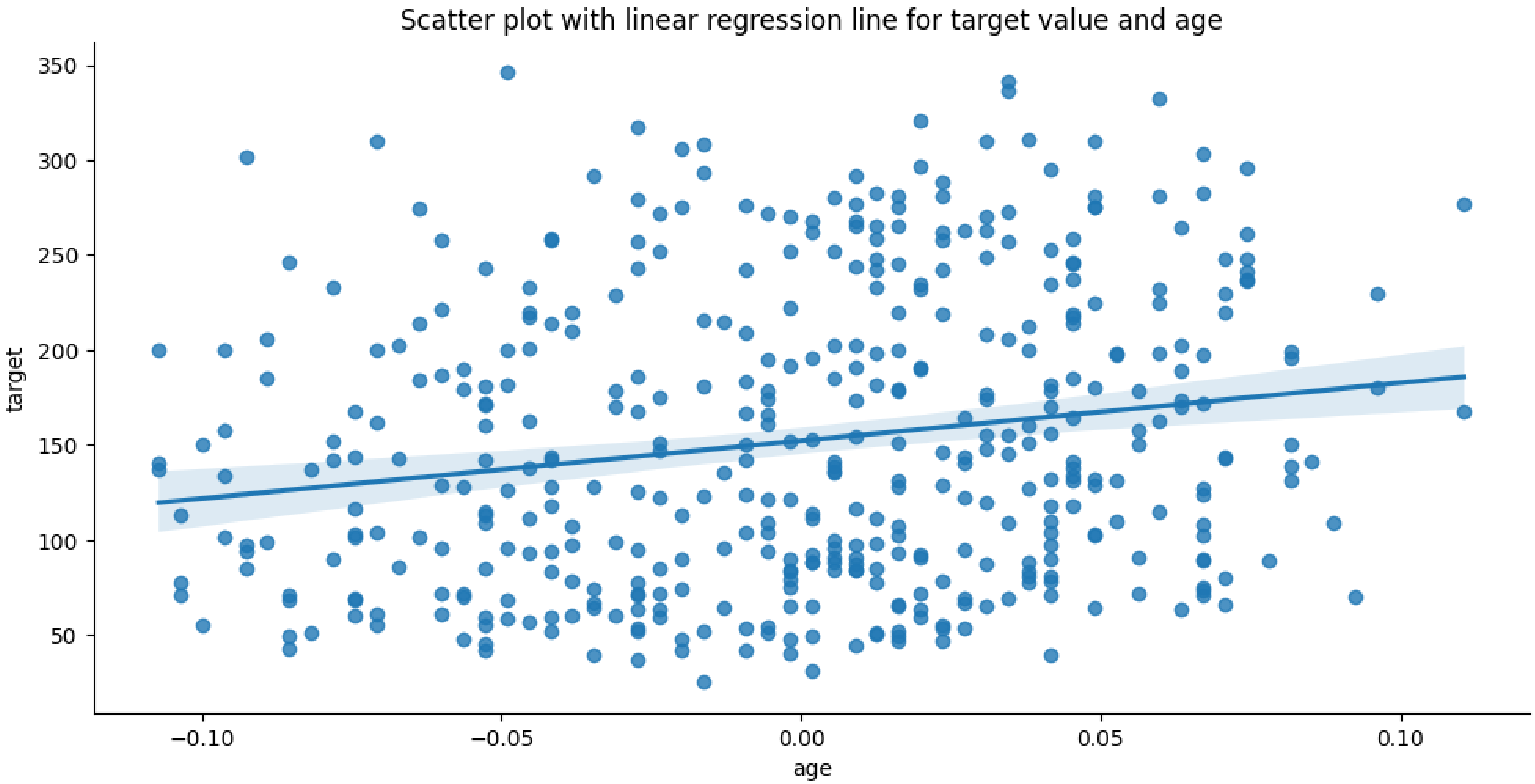


1b. Plot a scatter plot between the target value and one other feature in the data. Show the estimated linear regression line and describe what you have seen.

In [18]:

```
sns.lmplot(x = 'age', y = 'target', data = df, aspect = 2)
plt.title('Scatter plot with linear regression line for target value and age')
plt.show()

# It does not seem like there is a correlation between age and the target value, as the data
# points are completely scattered and do not really follow the regression line.
```



1c. Fit a linear model between the target value and BMI. Print the intercept and slope. Describe how BMI affects the target value.

In [45]:

```
import sklearn
from sklearn.linear_model import LinearRegression

lm = LinearRegression()

lm.fit(df[['bmi']], df['target'])

print("The estimated intercept is %.3f" %lm.intercept_)
print("The estimated coefficient is %.3f" %lm.coef_)
```

```
# Based on the large coefficient, there seems to be a very large positive slope, indicating there
# is a strong positive relationship between BMI and the target value.
```

```
The estimated intercept is 152.133
The estimated coefficient is 949.435
```

1d. Fir a linear model between the target value and all features. Print the coefficients for all features.

In [43]:

```
import sklearn
from sklearn.linear_model import LinearRegression

lm = LinearRegression()

X = df.drop(['target'], axis = 1)

lm.fit(df, df['target'])

import pandas as pd
from pandas import DataFrame

coef_df = DataFrame(X.columns)
coef_df.columns = ['Feature']
coef_df["Coefficient Estimate"] = pd.Series(lm.coef_)
print(coef_df)
```

```
  Feature  Coefficient Estimate
0     age         -3.284365e-13
1     sex          3.245182e-13
2     bmi          3.332404e-13
3      bp          1.180507e-13
4      s1         -5.568149e-13
5      s2          3.795514e-13
6      s3          1.560352e-13
7      s4          1.002938e-13
8      s5          3.990990e-13
9      s6          2.879112e-14
```

1e. Do you think linear model is a good model to use in this case? (Think about whether linear regression assumptions are satisfied.)

Not necessarily, although the first plot I made had a somewhat linear relationship, the second one did not, the points seemed to be scattered everywhere. So a linear model is appropriate for some of the variables such as BMI vs. target, but not all variables.

Question 2: Gradient Descent in Linear Regression (30)

Write Python code to implement the gradient descent method in linear regression. You can use the functions as a start (or you can start from scratch). Verify your code with sample data and LinearRegression in sklearn.

In [55]:

```
import numpy as np

# Generate a sample data with target y and feature X, feel free to change
x = np.random.normal(loc=5, scale=3, size=100)
y = 2 * x + np.random.normal(loc=0, scale=1, size=100)
X = np.column_stack((np.ones(100), x))
```

In [101…

```
# Implement the gradient descent functions. Import any module if you need.
import numpy as np

def compute_cost(predictions, y):

    n = len(y)
    # Compute the squared errors
    error = y - predictions

    squared_error = np.square(error)

    # Calculate the mean squared error (MSE)
    MSE = np.mean(squared_error)

    return MSE

def gradient_descent(X_b, y, theta, learning_rate, num_iterations):

    cost_history = []

    n = len(y)

    for i in range(num_iterations):
        prediction = np.dot(X_b, theta)
        error = y - prediction
        gradient = -(2/n) * np.dot(X_b.T, error)
        theta = theta - learning_rate * gradient

        cost = compute_cost(prediction, y)

        cost_history.append(cost)

    return theta, cost_history

learning_rate = 0.01
number_interactions = 1000
theta = np.zeros(X.shape[1])

t, c = gradient_descent(X, y, theta, learning_rate, number_interactions)

print('theta', t)
print('cost_history', c)

# verification with sklearn
import sklearn
from sklearn.linear_model import LinearRegression

lm = LinearRegression()

lm.fit(X, y)

coef_df["Coefficient Estimate"] = pd.Series(lm.coef_)

print('verification', coef_df)
```

```
theta [0.0180889  1.99758148]
cost_history [123.00492750654055, 18.131917831731847, 3.3401345610824857, 1.253627643450937, 0.959106379141
8872, 0.9173338858916148, 0.9112124472812887, 0.9101220449150237, 0.9097439251943323, 0.9094689287434881,
0.9092111066844087, 0.9089583053843688, 0.9087087798401595, 0.9084622534724712, 0.9082186571733964, 0.90797
79514543524, 0.907740101357164, 0.9075050729112738, 0.9072728326299265, 0.907043347435018, 0.90681658464246
03, 0.9065925119561271, 0.9063710974630547, 0.9061523096288704, 0.9059361172932974, 0.9057224896657157, 0.9
055113963207774, 0.9053028071940719, 0.9050966925778475, 0.9048930231167757, 0.9046917698037743, 0.90449290
39758739, 0.9042963973101366, 0.9041022218196219, 0.9039103498494002, 0.9037207540726153, 0.90353340748659
2, 0.9033482834089896, 0.9031653554740021, 0.9029845976286037, 0.9028059841288381, 0.9026294895361514, 0.90
24550887137691, 0.9022827568231169, 0.9021124693202823, 0.9019442019525189, 0.9017779307547948, 0.901613632
0463738, 0.9014512824274499, 0.9012908587758081, 0.9011323382435363, 0.9009756982537672, 0.900820916497466
3, 0.9006679709302527, 0.9005168397692594, 0.9003675014900319, 0.9002199348234616, 0.9000741187527581, 0.89
99300325104538, 0.8997876555754479, 0.8996469676700832, 0.8995079487572581, 0.8993705790375733, 0.899234838
9465091, 0.8991007091516446, 0.8989681705498994, 0.8988372042648145, 0.8987077916438642, 0.898579914255799
1, 0.898453553888022, 0.8983286925439916, 0.8982053124406635, 0.8980833960059529, 0.8979629258762349, 0.897
8438848938717, 0.8977262561047664, 0.8976100227559511, 0.8974951682931994, 0.8973816763586688, 0.8972695307
885722, 0.8971587156108738, 0.8970492150430158, 0.8969410134896708, 0.8968340955405192, 0.8967284459680556,
0.8966240497254201, 0.8965208919442551, 0.8964189579325865, 0.8963182331727338, 0.8962187033192401, 0.89612
03541968297, 0.8960231717983896, 0.8959271422829757, 0.8958322519738374, 0.8957384873564754, 0.895645835076
712, 0.8955542819387924, 0.8954638149035038, 0.8953744210863188, 0.8952860877555598, 0.8951988023305865, 0.
895112552380039, 0.8950273256198921, 0.8949431099120545, 0.8948598932622935, 0.8947776638186987, 0.8946964
09869959, 0.8946161198436962, 0.8945367823048168, 0.8944583859538819, 0.8943809196255, 0.894304372286733 7,
0.8942287330355317, 0.8941539910991747, 0.8940801358327399, 0.8940071567175878, 0.8939350433598606, 0.89386
37854890057, 0.8937933729563088, 0.8937237957334527, 0.893655043911085, 0.8935871076974107, 0.8935199774167
935, 0.8934536435083826, 0.8933880965247468, 0.8933233271305305, 0.8932593261011241, 0.893196084321352, 0.8
931335927841705, 0.8930718425893883, 0.893010824942398, 0.8929505311529234, 0.8928909526337806, 0.892832080
8996574, 0.8927739075659038, 0.8927164243473358, 0.8926596230570601, 0.8926034956053027, 0.892548033998259
8, 0.8924932303369598, 0.8924390768161367, 0.8923855657231182, 0.8923326894367295, 0.8922804404262056, 0.89
22288112501192, 0.8921777945553234, 0.892127383075899, 0.8920775696321261, 0.8920283471294561, 0.8919797085
575054, 0.891931646989054, 0.8918841555790606, 0.891837227563687, 0.8917908562593359, 0.8917450350616956,
0.8916997574448047, 0.8916550169601192, 0.891610807235593, 0.8915671219747747, 0.8915239549559061, 0.891481
3000310388, 0.891439151125159, 0.8913975022353199, 0.8913563474297895, 0.891315680847203, 0.891275496695731
6, 0.8912357892522523, 0.8911965528615393, 0.8911577819354528, 0.8911194709521473, 0.8910816144552824, 0.89
10442070532463, 0.8910072434183895, 0.8909707182862644, 0.8909346264548761, 0.8908989627839407, 0.890863722
194155, 0.8908288996664712, 0.8907944902413836, 0.8907604890182202, 0.8907268911544477, 0.8906936918649794,
0.8906608864214945, 0.8906284701517647, 0.890596438438989, 0.8905647867211356, 0.8905335104902921, 0.890502
6052920245, 0.8904720667247422, 0.8904418904390705, 0.8904120721372324, 0.8903826075724345, 0.8903534925482
652, 0.890324722918094, 0.8902962945844828, 0.8902682034986015, 0.8902404456596514, 0.8902130171142966, 0.8
901859139560987, 0.8901591323249626, 0.8901326684065853, 0.8901065184319137, 0.890080678676607, 0.890055145
4605062, 0.8900299151471113, 0.8900049841430614, 0.8899803488976247, 0.8899560059021913, 0.889931951689774
9, 0.8899081828345183, 0.8898846959512049, 0.8898614876947781, 0.8898385547598635, 0.889815893880298, 0.889
793501828667, 0.8897713754158413, 0.8897495114905253, 0.8897279069388077, 0.8897065586837168, 0.88968546368
```

```
4785, 0.8896646189376124, 0.8896440214734411, 0.8896236683587325, 0.8896035566947477, 0.8895836836171368,
0.8895640462955287, 0.8895446419331305, 0.8895254677663276, 0.8895065210642893, 0.8894877991285827, 0.88946
92992927848, 0.8894510189221069, 0.8894329554130145, 0.8894151061928605, 0.8893974687195179, 0.889380040481
0152, 0.8893628189951825, 0.8893458018092955, 0.8893289864997256, 0.8893123706715977, 0.8892959519584468,
0.8892797280218804, 0.8892636965512483, 0.8892478552633102, 0.8892322019019129, 0.8892167342376684, 0.88920
14500676342, 0.8891863472150046, 0.8891714235287945, 0.8891566768835389, 0.8891421051789848, 0.889127706339
7967, 0.8891134783152572, 0.8890994190789783, 0.8890855266286103, 0.8890717989855581, 0.8890582341946979,
0.8890448303241011, 0.8890315854647567, 0.8890184977303006, 0.8890055652567473, 0.888992786202223, 0.888980
1587467052, 0.8889676810917619, 0.8889553514602965, 0.8889431680962935, 0.8889311292645695, 0.8889192332505
261, 0.8889074783599054, 0.8888958629185479, 0.8888843852721551, 0.8888730437860538, 0.8888618368449628, 0.
8888507628527635, 0.8888398202322716, 0.8888290074250138, 0.8888183228910046, 0.8888077651085269, 0.8887973
325739175, 0.8887870238013498, 0.8887768373226246, 0.8887667716869595, 0.8887568254607827, 0.88874699722752
95, 0.8887372855874405, 0.88872768915736, 0.8887182065705429, 0.8887088364764563, 0.8886995775405897, 0.888
6904284442637, 0.8886813878844424, 0.8886724545735479, 0.888663627239278, 0.8886549046244215, 0.88864628548
66845, 0.8886377685985084, 0.8886293527468985, 0.8886210367332488, 0.888612819373174, 0.8886046994963374,
0.888596675946288, 0.8885887475802922, 0.8885809132691739, 0.8885731718971521, 0.8885655223616827, 0.888557
9635733001, 0.8885504944554645, 0.8885431139444069, 0.888535820988977, 0.888528614550495, 0.888521493602603
1, 0.8885144571311199, 0.8885075041338933, 0.8885006336206626, 0.8884938446129117, 0.8884871361437348, 0.88
84805072576966, 0.8884739570106945, 0.8884674844698282, 0.8884610887132617, 0.8884547688300976, 0.888448523
9202426, 0.8884423530942824, 0.8884362554733535, 0.8884302301890179, 0.888424276383141, 0.8884183932077676,
0.888412579825002, 0.8884068354068887, 0.8884011591352937, 0.88839555020179, 0.8883900078075395, 0.88838453
1163181, 0.8883791194887191, 0.8883737720134094, 0.8883684879756524, 0.888363266622883, 0.8883581072114639,
0.8883530090065794, 0.8883479712821313, 0.8883429933206353, 0.8883380744131182, 0.8883332138590175, 0.88832
8410966082, 0.8883236650502719, 0.8883189754356638, 0.8883143414543504, 0.8883097624463503, 0.8883052377595
098, 0.8883007667494127, 0.8882963487792872, 0.8882919832199155, 0.8882876694495441, 0.8882834068537965, 0.
8882791948255832, 0.8882750327650174, 0.8882709200793286, 0.8882668561827783, 0.8882628404965778, 0.8882588
724488021, 0.8882549514743141, 0.8882510770146779, 0.888247248518085, 0.8882434654392712, 0.88823972723944
2, 0.888236033386194, 0.8882323833534401, 0.8882287766213338, 0.8882252126761963, 0.8882216910104413, 0.888
2182111225062, 0.8882147725167759, 0.888211374703517, 0.8882080171988049, 0.8882046995244554, 0.88820142120
79575, 0.8881981817824058, 0.8881949807864342, 0.8881918177641489, 0.8881886922650649, 0.8881856038440415,
0.8881825520612181, 0.8881795364819528, 0.8881765566767599, 0.8881736122212485, 0.8881707026960622, 0.88816
78276868199, 0.8881649867840554, 0.8881621795831611, 0.888159405684329, 0.8881566646924938, 0.8881539562172
768, 0.8881512798729309, 0.8881486352782844, 0.8881460220566888, 0.8881434398359618, 0.888140888248338, 0.8
881383669304147, 0.8881358755230991, 0.8881334136715602, 0.8881309810251738, 0.888128577237478, 0.888126201
9661195, 0.8881238548728058, 0.8881215356232594, 0.8881192438871666, 0.8881169793381332, 0.888114741653637,
0.8881125305149813, 0.888110345607251, 0.8881081866192659, 0.8881060532435384, 0.8881039451762279, 0.888101
8621170983, 0.888099803769476, 0.8880977698402057, 0.8880957600396105, 0.8880937740814501, 0.88809181168287
98, 0.8880898725644095, 0.8880879564498665, 0.8880860630663523, 0.888084192144207, 0.8880823434169699, 0.88
80805166213405, 0.8880787114971423, 0.8880769277872854, 0.8880751652377299, 0.8880734235974489, 0.888071702
6183941, 0.8880700020554596, 0.8880683216664471, 0.8880666612120319, 0.8880650204557283, 0.888063399163855
6, 0.8880617971055055, 0.8880602140525096, 0.8880586497794054, 0.8880571040634055, 0.8880555766843656, 0.88
80540674247527, 0.8880525760696142, 0.8880511024065483, 0.888049646225672, 0.888048207319593, 0.88804678548
```

```
33785, 0.8880453805145274, 0.8880439922129411, 0.8880426203808939, 0.8880412648230063, 0.8880399253462172,
0.8880386017597541, 0.8880372938751099, 0.8880360015060121, 0.8880347244683978, 0.8880334625803896, 0.88803
22156622653, 0.8880309835364365, 0.8880297660274207, 0.8880285629618174, 0.8880273741682835, 0.888026199477
5077, 0.8880250387221884, 0.8880238917370076, 0.8880227583586096, 0.8880216384255748, 0.8880205317784007,
0.8880194382594762, 0.8880183577130589, 0.8880172899852556, 0.8880162349239975, 0.8880151923790203, 0.88801
41622018428, 0.8880131442457444, 0.8880121383657453, 0.8880111444185862, 0.888010162262706, 0.8880091917582
242, 0.8880082327669194, 0.8880072851522095, 0.8880063487791332, 0.8880054235143304, 0.8880045092260223, 0.
888003605783994, 0.8880027130595743, 0.8880018309256205, 0.8880009592564967, 0.88800097928057, 0.887999246
8176302, 0.8879984058039979, 0.8879975747673822, 0.8879967535894235, 0.8879959421531693, 0.887995140343051
4, 0.8879943480448752, 0.8879935651457993, 0.8879927915343215, 0.887992027100262, 0.8879912717347489, 0.887
9905253302007, 0.887989787780313, 0.8879890589800417, 0.8879883388255901, 0.8879876272143915, 0.88798692404
50973, 0.8879862292175599, 0.8879855426328201, 0.8879848641930934, 0.8879841938017543, 0.887983531363325,
0.887982876783459, 0.8879822299689288, 0.8879815908276146, 0.8879809592684877, 0.8879803352016004, 0.887979
7185380712, 0.887979109190074, 0.8879785070708236, 0.8879779120945648, 0.8879773241765601, 0.8879767432307
63, 0.8879761691813738, 0.8879756019396956, 0.8879750414272537, 0.8879744875642183, 0.887973940271707, 0.88
79733994717731, 0.8879728650873941, 0.8879723370424631, 0.8879718152617734, 0.8879712996710125, 0.887970790
1967483, 0.8879702867664204, 0.887969789308329, 0.8879692977516249, 0.8879688120263003, 0.8879683320631758,
0.8879678577938951, 0.8879673891509107, 0.887966926067478, 0.8879664684776426, 0.8879660163162347, 0.887965
5695188561, 0.8879651280218721, 0.8879646917624044, 0.8879642606783199, 0.8879638347082222, 0.8879634137914
44, 0.8879629978680369, 0.8879625868787648, 0.8879621807650931, 0.8879617794691823, 0.8879613829338792, 0.8
879609911027074, 0.8879606039198623, 0.8879602213302, 0.8879598432792314, 0.8879594697131131, 0.88795910057
86416, 0.8879587358232439, 0.8879583753949699, 0.8879580192424874, 0.8879576673150719, 0.8879573195626009,
0.8879569759355473, 0.8879566363849701, 0.8879563008625107, 0.8879559693203821, 0.8879556417113661, 0.88795
53179888038, 0.8879549981065896, 0.8879546820191646, 0.8879543696815123, 0.8879540610491476, 0.887953756078
1152, 0.8879534547249796, 0.8879531569468218, 0.8879528627012314, 0.8879525719463017, 0.8879522846406224,
0.8879520007432745, 0.8879517202138248, 0.88795144301232, 0.8879511690992798, 0.8879508984356937, 0.8879506
309830123, 0.8879503667031446, 0.8879501055584513, 0.88794984751174, 0.8879495925262588, 0.887949340565691
5, 0.8879490915941538, 0.8879488455761866, 0.8879486024767518, 0.8879483622612259, 0.8879481248953968, 0.88
79478903454592, 0.8879476585780072, 0.8879474295600318, 0.8879472032589159, 0.8879469796424302, 0.887946758
6787253, 0.8879465403363311, 0.8879463245841521, 0.887946111391459, 0.8879459007278889, 0.8879456925634385,
0.8879454868684609, 0.8879452836136598, 0.8879450827700879, 0.8879448843091404, 0.8879446882025519, 0.88794
44944223923, 0.8879443029410633, 0.8879441137312933, 0.887943926766135, 0.8879437420189602, 0.8879435594634
567, 0.8879433790736249, 0.8879432008237726, 0.8879430246885134, 0.8879428506427621, 0.8879426786617298, 0.
8879425087209234, 0.8879423407961393, 0.8879421748634609, 0.8879420108992558, 0.8879418488801722, 0.8879416
887831346, 0.8879415305853416, 0.8879413742642626, 0.8879412197976337, 0.8879410671634556, 0.88794091633998
96, 0.8879407673057552, 0.8879406200395266, 0.8879404745203296, 0.8879403307274393, 0.8879401886403762, 0.8
879400482389039, 0.8879399095030263, 0.887939772412984, 0.8879396369492527, 0.8879395030925391, 0.887939370
8237788, 0.8879392401241339, 0.8879391109749903, 0.8879389833579538, 0.8879388572548491, 0.887938732647716,
0.8879386095188082, 0.8879384878505889, 0.8879383676257302, 0.8879382488271085, 0.8879381314378056, 0.88793
80154411018, 0.8879379008204766, 0.8879377875596056, 0.8879376756423578, 0.8879375650527941, 0.887937455775
1636, 0.8879373477939028, 0.8879372410936328, 0.8879371356591572, 0.8879370314754599, 0.8879369285277025,
0.8879368268012232, 0.8879367262815338, 0.8879366269543175, 0.8879365288054288, 0.8879364318208884, 0.88793
```

```
63359868843, 0.8879362412897669, 0.8879361477160492, 0.8879360552524048, 0.887935963885664, 0.8879358736028
147, 0.8879357843909986, 0.8879356962375098, 0.8879356091297934, 0.8879355230554427, 0.8879354380022, 0.887
9353539579512, 0.8879352709107259, 0.8879351888486972, 0.8879351077601769, 0.8879350276336173, 0.8879349484
57605, 0.8879348702208651, 0.887934792912254, 0.8879347165207616, 0.887934641035508, 0.8879345664457423, 0.
8879344927408411, 0.8879344199103077, 0.8879343479437688, 0.8879342768309751, 0.887934206561799, 0.88793413
71262314, 0.8879340685143842, 0.8879340007164851, 0.8879339337228781, 0.8879338675240218, 0.887933802110488
4, 0.8879337374729616, 0.8879336736022349, 0.8879336104892117, 0.8879335481249039, 0.8879334865004295, 0.88
7933425607011, 0.8879333654359765, 0.887933305978756, 0.8879332472268817, 0.8879331891719859, 0.88793313180
57999, 0.8879330751201544, 0.8879330191069748, 0.8879329637582845, 0.8879329090662008, 0.8879328550229337,
0.8879328016207865, 0.8879327488521532, 0.8879326967095191, 0.8879326451854574, 0.8879325942726293, 0.88793
25439637854, 0.8879324942517591, 0.8879324451294707, 0.8879323965899242, 0.887932348626207, 0.8879323012314
869, 0.8879322543990144, 0.8879322081221196, 0.8879321623942116, 0.8879321172087772, 0.8879320725593819, 0.
8879320284396666, 0.8879319848433461, 0.8879319417642131, 0.8879318991961312, 0.8879318571330381, 0.8879318
155689434, 0.8879317744979269, 0.8879317339141395, 0.8879316938118006, 0.8879316541851994, 0.88793161502869
18, 0.8879315763367015, 0.8879315381037174, 0.8879315003242945, 0.8879314629930515, 0.8879314261046729, 0.8
87931389653904, 0.8879313536355538, 0.8879313180444924, 0.8879312828756504, 0.8879312481240192, 0.887931213
7846501, 0.8879311798526515, 0.8879311463231916, 0.8879311131914936, 0.8879310804528405, 0.887931048102568
4, 0.8879310161360706, 0.8879309845487942, 0.8879309533362398, 0.8879309224939632, 0.8879308920175703, 0.88
79308619027223, 0.887930832145129, 0.8879308027405526, 0.8879307736848048, 0.8879307449737484, 0.8879307166
03293, 0.8879306885693993, 0.8879306608680737, 0.8879306334953712, 0.8879306064473937, 0.8879305797202889,
0.8879305533102492, 0.8879305272135147, 0.8879305014263679, 0.887930475945136, 0.8879304507661903, 0.887930
4258859445, 0.887930401300855, 0.8879303770074206, 0.8879303530021814, 0.8879303292817181, 0.88793030584265
3, 0.8879302826816469, 0.8879302597954026, 0.8879302371806591, 0.8879302148341967, 0.887930192752832, 0.887
9301709334204, 0.8879301493728543, 0.8879301280680635, 0.8879301070160127, 0.8879300862137042, 0.8879300656
581754, 0.8879300453464988, 0.8879300252757808, 0.8879300054431638, 0.8879299858458229, 0.8879299664809664,
0.8879299473458377, 0.8879299284377099, 0.8879299097538911, 0.8879298912917202, 0.8879298730485671, 0.88792
98550218345, 0.8879298372089544, 0.88792981960739, 0.8879298022146342, 0.8879297850282106, 0.88792976804567
09, 0.8879297512645963, 0.8879297346825971, 0.8879297182973113, 0.8879297021064062, 0.8879296861075747, 0.8
879296702985389, 0.887929654677047, 0.887929639240874, 0.8879296239878218, 0.8879296089157175, 0.8879295940
224154, 0.8879295793057935, 0.8879295647637561, 0.8879295503942319, 0.8879295361951749, 0.8879295221645627,
0.8879295083003966, 0.887929494600702, 0.8879294810635284, 0.8879294676869475, 0.8879294544690539, 0.887929
4414079655, 0.8879294285018219, 0.8879294157487851, 0.8879294031470379, 0.887929390694787, 0.88792937839025
8, 0.8879293662316988, 0.8879293542173773, 0.8879293423455827, 0.8879293306146249, 0.8879293190228317, 0.88
79293075685534, 0.8879292962501576, 0.8879292850660333, 0.8879292740145872, 0.8879292630942452, 0.887929252
3034523, 0.8879292416406714, 0.887929231104384, 0.8879292206930899, 0.8879292104053053, 0.8879292002395656,
0.8879291901944231, 0.8879291802684467, 0.8879291704602237, 0.8879291607683558, 0.8879291511914633, 0.88792
91417281828, 0.8879291323771656, 0.8879291231370806, 0.8879291140066108, 0.8879291049844575, 0.887929096069
3347, 0.8879290872599722, 0.887929078555116, 0.887929069953526, 0.8879290614539781, 0.8879290530552606, 0.8
879290447561782, 0.8879290365555478, 0.8879290284522031, 0.8879290204449884, 0.887929012532764, 0.887929004
7144031, 0.8879289969887924, 0.8879289893548308, 0.8879289818114315, 0.8879289743575208, 0.887928966992035
9, 0.8879289597139288, 0.887928952522162, 0.8879289454157118, 0.8879289383935663, 0.8879289314547252, 0.887
9289245981999, 0.8879289178230141, 0.8879289111282028, 0.8879289045128131, 0.887928897975902, 0.88792889151
```

```
65389, 0.8879288851338036, 0.8879288788267875, 0.887928872594592, 0.8879288664363295, 0.8879288603511232,
0.8879288543381059, 0.8879288483964216, 0.8879288425252239, 0.8879288367236771, 0.8879288309909542, 0.88792
88253262394, 0.8879288197287252, 0.8879288141976152, 0.8879288087321207, 0.8879288033314642, 0.887928797994
8761, 0.8879287927215965, 0.8879287875108739, 0.887928782361967, 0.8879287772741415, 0.8879287722466738, 0.
8879287672788473, 0.8879287623699541, 0.8879287575192959, 0.8879287527261814, 0.8879287479899282, 0.8879287
433098613, 0.8879287386853144, 0.8879287341156292, 0.8879287296001543, 0.8879287251382466, 0.88792872072927
15, 0.8879287163726002, 0.8879287120676125, 0.887928707813695, 0.8879287036102422, 0.8879286994566555, 0.88
79286953523433, 0.8879286912967204, 0.8879286872892096, 0.887928683329241, 0.8879286794162495, 0.8879286755
496778, 0.887928671728976, 0.8879286679535991, 0.8879286642230102, 0.8879286605366774, 0.8879286568940757,
0.8879286532946868, 0.8879286497379977, 0.8879286462235021, 0.8879286427506987, 0.8879286393190939, 0.88792
86359281986, 0.8879286325775295, 0.8879286292666098, 0.8879286259949679, 0.8879286227621377, 0.887928619567
6588, 0.8879286164110759, 0.8879286132919401, 0.8879286102098071, 0.8879286071642373, 0.8879286041547979,
0.8879286011810598, 0.8879285982425985, 0.8879285953389973, 0.8879285924698412, 0.8879285896347221, 0.88792
85868332367, 0.8879285840649852, 0.8879285813295741, 0.8879285786266132, 0.8879285759557181, 0.887928573316
5073, 0.8879285707086063, 0.8879285681316427, 0.8879285655852499, 0.8879285630690654, 0.8879285605827302,
0.8879285581258911, 0.8879285556981974, 0.8879285532993036, 0.8879285509288681, 0.8879285485865533, 0.88792
8546272026, 0.8879285439849555, 0.8879285417250172, 0.8879285394918885, 0.8879285372852516, 0.8879285351047
922, 0.8879285329501997, 0.8879285308211674, 0.8879285287173918, 0.887928526638574, 0.8879285245844168, 0.8
879285225546285, 0.8879285205489199, 0.887928518567005, 0.887928516608602, 0.8879285146734313, 0.8879285127
61218]
verification       0         1  Coefficient Estimate
0   1.0  7.737035              0.000000
1   1.0  4.094805              1.997691
2   1.0  1.856212                   NaN
3   1.0  3.707321                   NaN
4   1.0  4.859375                   NaN
..  ...       ...                   ...
95  1.0  7.816982                   NaN
96  1.0 -1.013363                   NaN
97  1.0  0.122137                   NaN
98  1.0 -3.615189                   NaN
99  1.0  3.351242                   NaN

[100 rows x 3 columns]
```

In [95]:

```
# Run your code with the sample X and y. Then compare with the result with LinearRegression in sklearn.
```

Question 3 MLE for Poisson Regression (30)

In the lecture we have showed you how to achieve the MLE for the linear regression. Now let's try to replicate it with another regression: Poisson regression. We will discuss it more later so don't worry if you have never heard about it.

Poisson regression only has one parameter λ, which is also the mean of the distribution (like the μ in the linear regression). The pdf is

$$f(y_i|X_i, \lambda) = \frac{\lambda^{X_i} e^{-\lambda}}{X_i!}$$

while $X_i!$ is the factorial of X_i.

Now try to esimate the MLE for the λ in Poisson regression. Latex is preferred but written and submitted by picture is also accpetable. (Hint: review the lecture note and follow the steps: write the likelihood, calculate the log-likelihood, take the derivative and solve the equation when the derivative is equal to 0. The answer is a simple one and there is no factorial included.)

$$L(\lambda) = \prod_{i=1}^{n} \frac{\lambda^{X_i} e^{-\lambda}}{X_i!}$$

$$log(L(\lambda)) = \sum_{i=1}^{n} log(\frac{\lambda^{X_i} e^{-\lambda}}{X_i!})$$

$$log(L(\lambda)) = \sum_{i=1}^{n} (log(\lambda^{X_i}) + log(e^{-\lambda})) - log(X_i!)$$

$$= \sum_{i=1}^{n} X_i log(\lambda) - \lambda$$

(drop log(X_i!) as there is no lambda it is not needed for final solution)

$$\frac{d}{d\lambda} = \sum_{i=1}^{n} \frac{X_i}{\lambda} - 1$$

$$\sum_{i=1}^{n}(\frac{X_i}{\lambda}-1)=0$$

$$\sum_{i=1}^{n}\frac{X_i}{\lambda}-\sum_{i=1}^{n}1=0$$

$$\sum_{i=1}^{n}\frac{X_i}{\lambda}=n$$

$$\frac{\sum_{i=1}^{n}X_i}{\sum_{i=1}^{n}\lambda}=n$$

$$\frac{\sum_{i=1}^{n}X_i}{n*\lambda}=n$$

$$\sum_{i=1}^{n}X_i=n*n*\lambda$$

$$\frac{\sum_{i=1}^{n}X_i}{n^2}=\lambda$$

Question 4 Another MLE practice (20)

Let's make a random pdf, when $0 \leq X_i \leq \theta$, we have:

$$f(y_i|X_i,\theta)=2*\frac{(\theta-X_i)}{\theta^2}$$

For all the other value, let $f(y_i|X_i,\theta)=0$.

1. First, prove it is a valid pdf. Hint: a valid pdf means all the values are positive and the interval of pdf should be equal to 1.
2. Calculate the MLE estimator. Stop after taking the deriative since the equation doesn't have a simple closed-form solution

1. The values are always positive because X is between 0 and theta, so the numerator will never be below 0. The integral from 0 to theta also equals 0, as show below:

$$\int_0^\theta 2 * \frac{\theta - X_i}{\theta^2}$$

$$= 2 * \int_0^\theta \frac{\theta}{\theta^2} - \frac{X_i}{\theta^2}$$

$$= 2 * \int_0^\theta \frac{\theta}{\theta^2} - \int_0^\theta \frac{X_i}{\theta^2}$$

$$= 2 * \int_0^\theta \frac{1}{\theta} dx - \int_0^\theta \frac{X_i}{\theta^2} dx$$

$$= 2 * (|_0^\theta \frac{X_i}{\theta} - |_0^\theta \frac{X_i^2}{2 * \theta^2})$$

$$= 2 * ((\frac{\theta}{\theta} - \frac{0}{\theta}) - (\frac{\theta^2}{2 * \theta^2} - \frac{0}{2 * \theta^2}))$$

$$= 2 * (1 - \frac{1}{2})$$

$$= 1$$

2.

$$L(\theta) = \prod_{i=1}^{n} 2 * \frac{\theta - X_i}{\theta^2}$$

$$log(L(\theta)) = \sum_{i=1}^{n} log(2 * \frac{\theta - X_i}{\theta^2})$$

$$log(L(\theta)) = \sum_{i=1}^{n} log(2) + (log(\theta - X_i) - log(\theta^2))$$

$$\frac{d}{d\theta} log(L(\theta)) = \sum_{i=1}^{n} \frac{1}{\theta - X_i} - \frac{2}{\theta}$$

In [ ]: