Assignment 4

**Question 1:**

1. load the data into Pandas dataframe. Extract two dataframes with the above 4 features: df 0 for surviving patients (DEATH EVENT = 0) and df 1 for deceased patients (DEATH EVENT = 1)

<AA> Here is the filtered data frame with the 4 features of interest:

1. creatinine phosphokinase

2. serum creatinine

3. serum sodium

4. platelets

**Death\_Event = 0**

creatinine\_phosphokinase platelets ... serum\_sodium DEATH\_EVENT

14 80 427000.0 ... 138 0

20 52 276000.0 ... 137 0

23 63 368000.0 ... 135 0

33 159 302000.0 ... 138 0

38 2656 305000.0 ... 137 0

.. ... ... ... ... ...

294 61 155000.0 ... 143 0

295 1820 270000.0 ... 139 0

296 2060 742000.0 ... 138 0

297 2413 140000.0 ... 140 0

298 196 395000.0 ... 136 0

[203 rows x 5 columns]

**Death\_Event = 1**

creatinine\_phosphokinase platelets ... serum\_sodium DEATH\_EVENT

0 582 265000.00 ... 130 1

1 7861 263358.03 ... 136 1

2 146 162000.00 ... 129 1

3 111 210000.00 ... 137 1

4 160 327000.00 ... 116 1

.. ... ... ... ... ...

220 582 263358.03 ... 134 1

230 166 62000.00 ... 127 1

246 2017 314000.00 ... 138 1

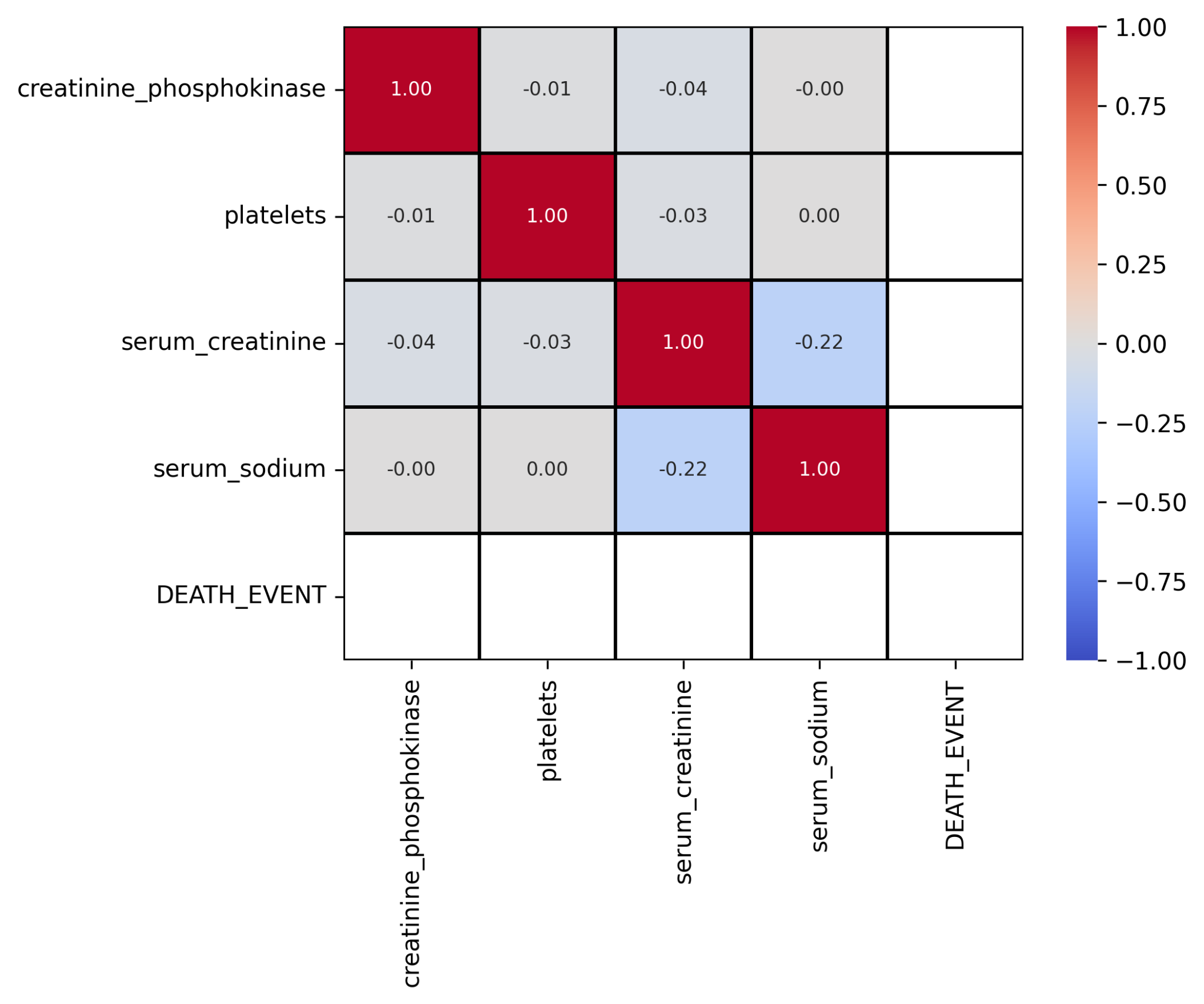
262 258 198000.00 ... 129 1

266 1199 263358.03 ... 134 1

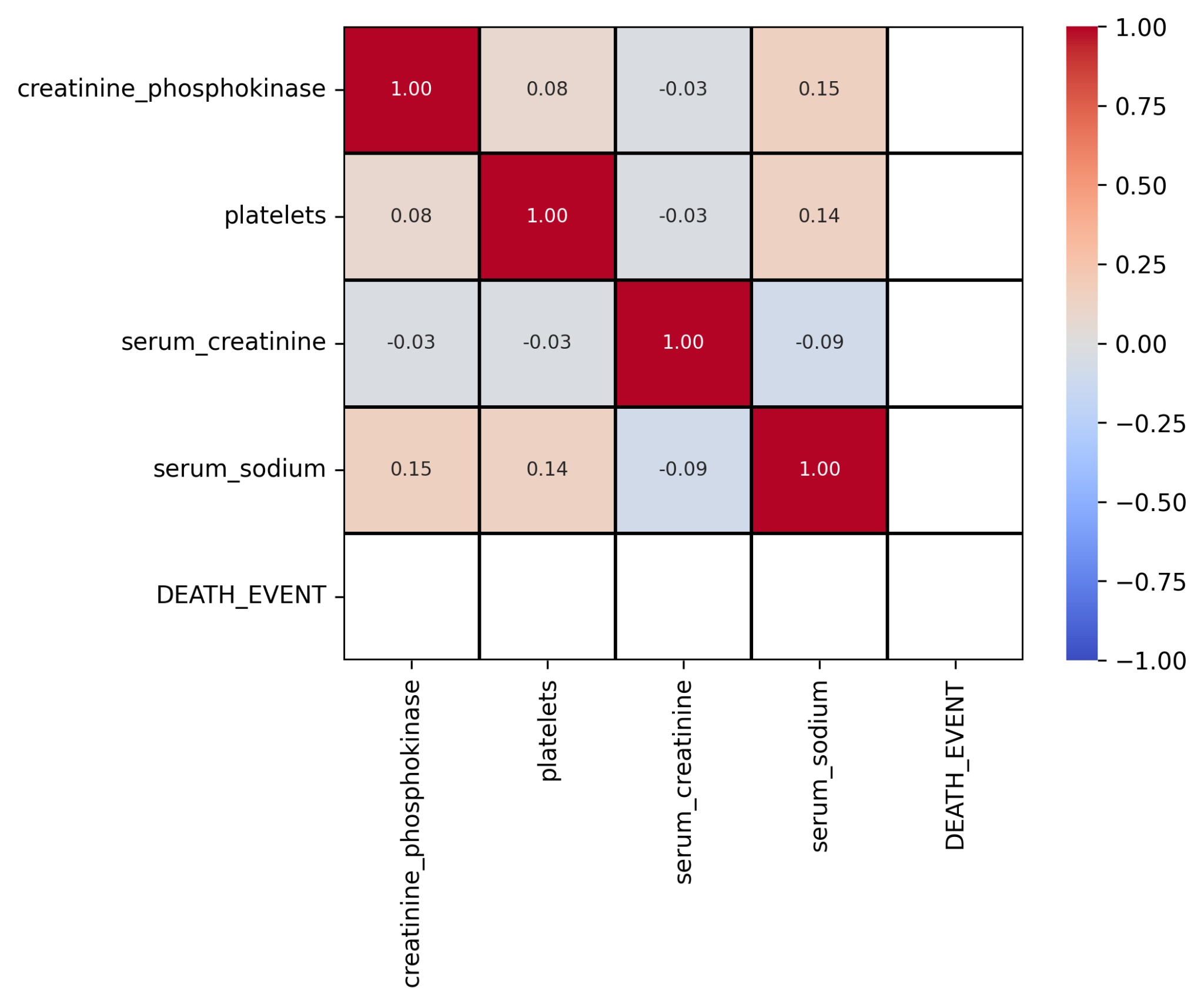
2. for each dataset, construct the visual representations of correponding correlation matrices M0 (from df 0) and M1 (from df 1) and save the plots into two separate files.

<AA> The heap maps for each dataframe is stored as df\_0 and df\_1

**df0\_surviving\_patients**



**df1\_deceased\_patients**

****

3. examine your correlation matrix plots visually and answer the following:

(a) which features have the highest correlation for surviving patients?

<AA> Based on the heat map, **platelets and serum\_sodium** turned out to have **highest positive** correlation among surviving patients with correlation of positive 0.00.

**creatinine\_phosphokinase and platelets** have **highest negative** correlation of -0.01

(b) which features have the lowest correlation for surviving patients?

<AA> **serum\_sodium and serum\_creatinine** has the **lowest negative** correlation of -0.22 among surviving patients.

**creatinine\_phosphokinase and serum\_sodium** have the **lowest positive** correlation of 0.00

(c) which features have the highest correlation for deceased patients?

<AA> Based on the heat map, **creatinine\_phosphokinase and serum\_sodium** turned out to have **highest positive** correlation among surviving patients with correlation of 0.15.

**creatinine\_phosphokinase and serum\_creatinine** turned out to have **highest negative** correlation of -0.03

(d) which features have the lowest correlation for deceased patients?

<AA> **serum\_creatinine and serum\_sodium** have **lowest negative** correlation among deceased patients of -0.09.

**creatinine\_phosphokinase and platelets** have **lowest positive** correlation of 0.08.

(e) are results the same for both cases?

<AA> Same features pairs for **lowest negative** correlation features only.

**Question 2:**

In this question you will compare a number of different models using linear systems (including linear regression). You choose one feature X as independent variable X and another feature Y as dependent. Your choice of X and Y will depend on your facilitator group as follows:

1. Group 1: X: creatinine phosphokinase (CPK), Y : platelets

2. Group 2: X: platelets, Y : serum sodium

3. Group 3: X: serum sodium, Y : serum creatinine

4. Group 4: X: platelets, Y : serum creatinine

We will now look for the best model (from the list below) that best explains the relationship for surviving and deceased patients. Consider surviving patients (DEATH EVENT = 0). Extract the corresponding columns for X and Y . For each of the models below, we will take 50/50 split, fit model with Xtrain and predict Ytest using Xtest. From the predicted values Pred(yi) we compute the residuals ri = yi − Pred(yi). We can then estimate the loss function (SSE sum of the squared of residuals)

You do the same analysis for deceased patients. You will consider the following models for both deceased and surviving patients:

1. y = ax + b (simple linear regression)

<AA>

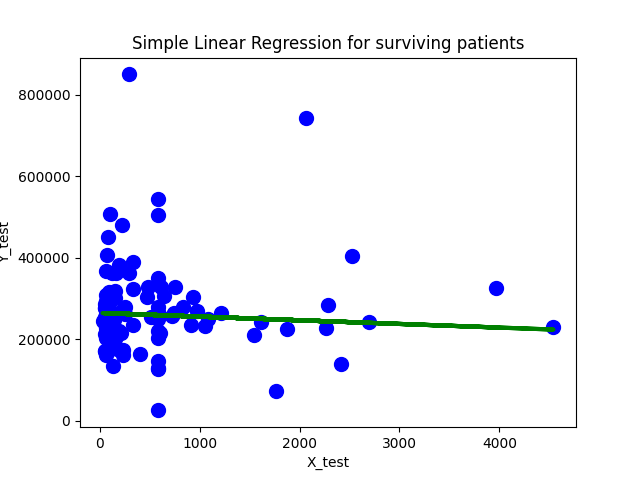
Simple Linear Regression for surviving patients score is 0.0064837195160663

Slope is : -8.974668497120126

Intercept is : 264309.5808763228

SSE (Sum of Squared Errors) is : platelets 1.323136e+12

dtype: float64



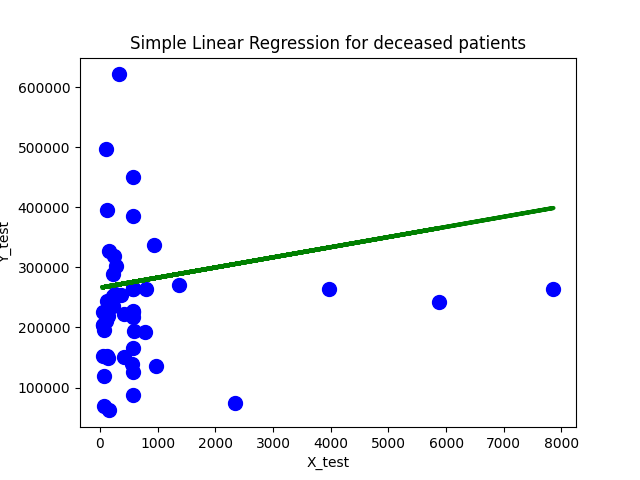
Simple Linear Regression for deceased patients score is 0.05333780337844696

Slope is : 16.889472765323813

Intercept is : 265963.4267818397

SSE (Sum of Squared Errors) is : platelets 6.529056e+11

dtype: float64



2. y = ax2 + bx + c (quadratic)

<AA>

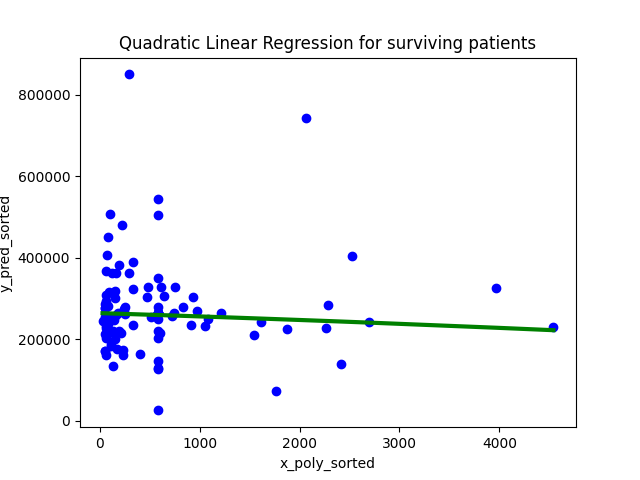
Quadratic Polynomial Regression for surviving patients score is 0.00651157727159446

Slope is : [[ 0.00000000e+00 -7.78649490e+00 -3.21114887e-04]]

Intercept is : [263963.19426985]

SSE (Sum of Squared Errors) is : platelets 1.322724e+12

dtype: float64



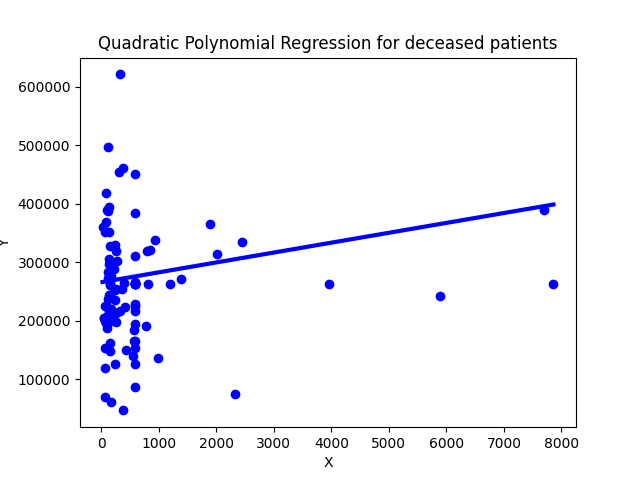
Quadratic Polynomial Regression for deceased patients score is 0.05333780931123122

Slope is : [[ 0.00000000e+00 1.69059861e+01 -2.37947364e-06]]

Intercept is : [265958.01683427]

SSE (Sum of Squared Errors) is : platelets 6.529151e+11

dtype: float64



3. y = ax3 + bx2 + cx + d (cubic spline)

<AA>

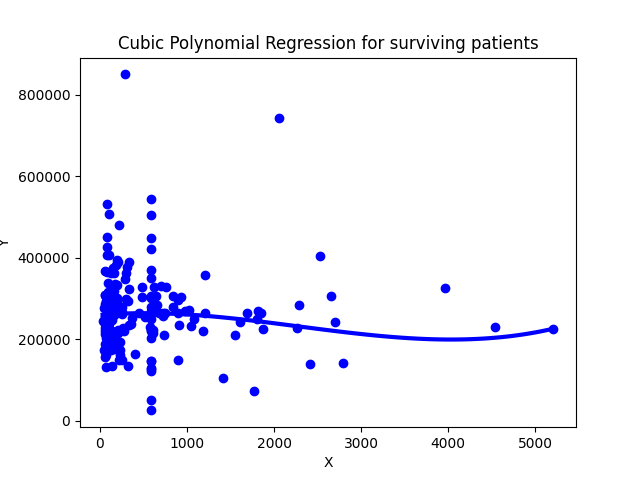
Cubic Polynomial Regression for surviving patients score is 0.009713914587271222

Slope is : [[ 1.88328890e+01 -2.04149082e-02 2.98904497e-06]]

Intercept is : [259129.46851901]

SSE (Sum of Squared Errors) is : platelets 1.338190e+12

dtype: float64



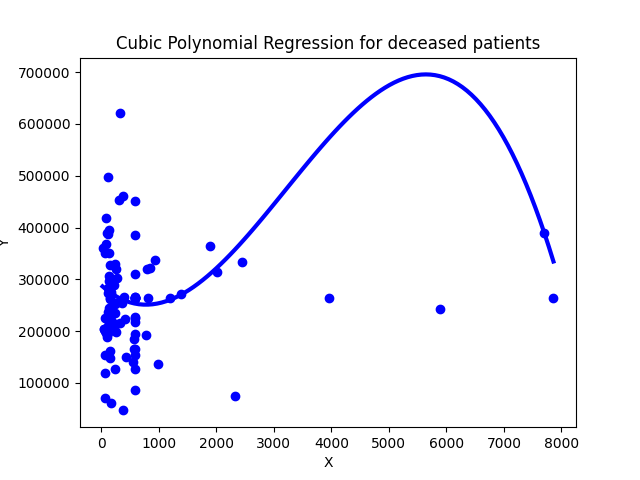
Cubic Polynomial Regression for deceased patients score is 0.0949525843524861

Slope is : [[-1.01306730e+02 7.42499912e-02 -7.71161827e-06]]

Intercept is : [288602.76166098]

SSE (Sum of Squared Errors) is : platelets 9.440193e+11

dtype: float64



4. y = a log x + b (GLM - generalized linear model)

<AA>

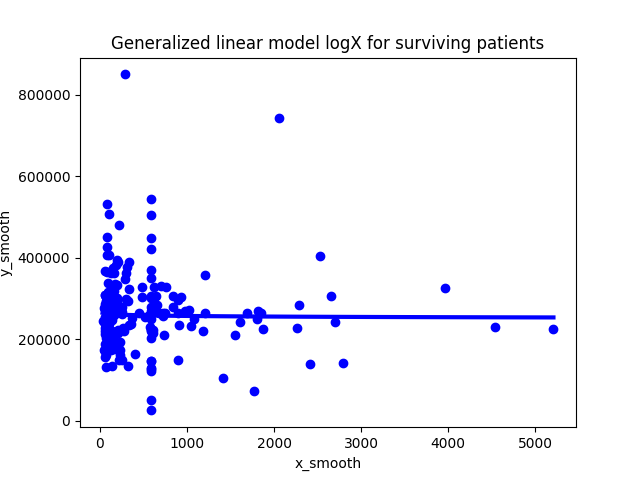
Generalized linear model logX for surviving patients score is 0.0009060491117462988

Slope is : [[-2216.24894752]]

Intercept is : [272297.95486106]

SSE (Sum of Squared Errors) is : platelets 1.315832e+12

dtype: float64



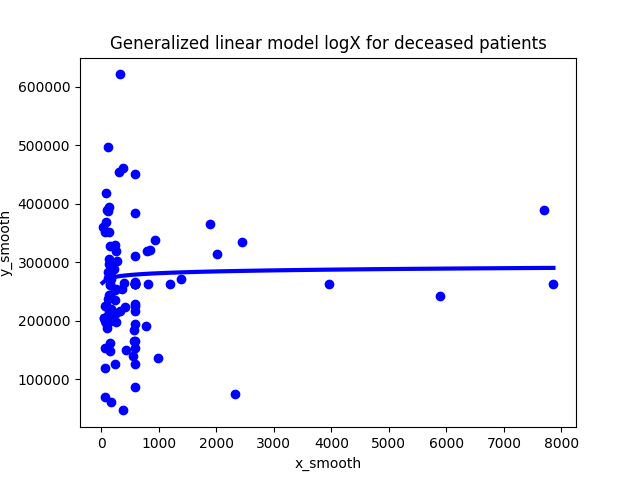
Generalized linear model logX for deceased patients score is 0.0031715854532758936

Slope is : [[4419.51430869]]

Intercept is : [250822.85792697]

SSE (Sum of Squared Errors) is : platelets 6.158507e+11

dtype: float64



5. log y = a log x + b (GLM - generalized linear model)

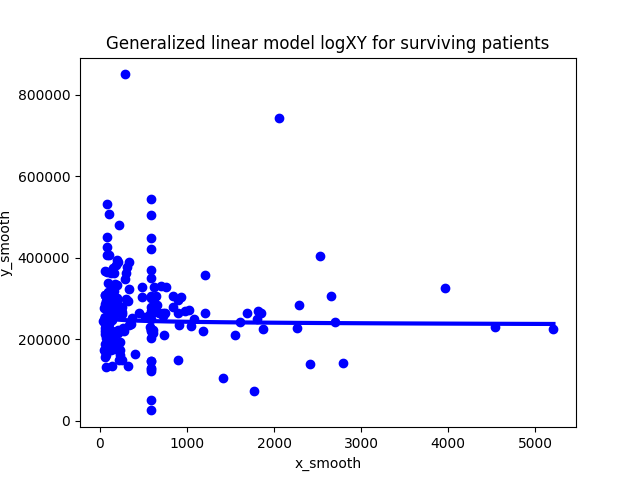
<AA>

Generalized linear model logXY for surviving patients score is 0.0018372494697552755

Slope is : [-0.01356359]

Intercept is : 12.493206031332187

SSE (Sum of Squared Errors) is : 17.841136537858077

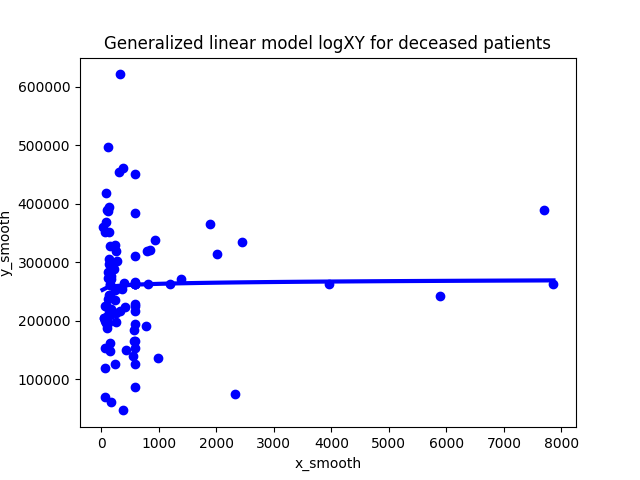
****

Generalized linear model logXY for deceased patients score is 0.0009199721559858265

Slope is : [0.01072277]

Intercept is : 12.406651661766794

SSE (Sum of Squared Errors) is : 12.395267666818743



For each of the model below, you will do the following (for both deceased and surviving patients)

(a) fit the model on Xtrain

(b) print the weights (a, b, . . .)

(c) compute predicted values using Xtest

(d) plot (if possible) predicted and actual values in Xtest

(e) compute (and print) the corresponding loss function

**Question 3:**

**Summarize your results from question 2 in a table like shown below:**

| Model | SSE(Death event = 0) | SSE(Death event = 1) |
| --- | --- | --- |
| y = ax + b  y = ax2 + bx + c  **y = ax3 + bx2 + cx + d**  y = a log x + b  **log y = a log x + b** | 1.323136e+12  1.322724e+12  1.338190e+12  1.315832e+12  17.841136537858077 | 6.529056e+11  6.529151e+11  9.440193e+11  6.158507e+11  12.406651661766794 |

1. which model was the best (smallest SSE) for surviving patients? for deceased patients?

<AA> Min SSE for both surviving and deceased patients were from generalized linear model with both X(independent) and Y(dependent) variables were transformed to log before fitting into the model for training.

2. which model was the worst (largest SSE) for surving patients? for deceased patients?

<AA> The largest SSE came from cublic spline model where the X feature data is transformed polynomial degree =3 before being fed for the model training. This model also generated a curve in the linear model creating more residual towards the entire data points.