

Assume data.csv is the observed data in form of input-output pair. Please use the family of exponential function, $y = \alpha e^{\beta x}$ with $\beta < 0$, for the regression analysis and find the optimal coefficients, α and β . You may use JavaScript, python, or other language to solve this problem. In addition, compare the difference of estimation error for linear regression and exponential regression.

```
let x=[0,1,2,3,4,5,6,7,8,9,10]
let y=[9735,4597,2176,1024,483,229,108,52,24,11,6]
```

線性回歸結果：

截距 (alpha): 5120.636

斜率 (beta): -688.764

相關性 (correlation): -0.757

Alpha 代表 $X=0$ 時，Y 的估計值為 5120.636。

Beta 為 -688.764，負斜率表明 Y 與 X 之間的負相關。

相關係數越接近 1 表示正向線性相關性越強，相關係數越接近 -1 表示負向線性相關性越強。而此線性回歸結果中的相關係數為 -0.757，可以說明 X 和 Y 之間有負相關性，表示隨著 X 的增加，Y 呈現下降趨勢，程度為中度強度，但沒有到十分強烈。

指數回歸結果：

係數 (a): 9620.329

指數 (b): -0.746

相關性：-0.99999

係數 a 表示當 $X=0$ 時，Y 的估計值為 9620.329。

指數 b 結果為 -0.746，代表隨著 X 增加，Y 成指數遞減的趨勢，其絕對值越大，遞減速度越快。

線性回歸主要用於測量因變數(Y)與一或多個自變數(X)之間的線性關係，線性回歸目標是找出最佳的直線，使得直線與觀察的數據點之間的誤差最小，線性回歸會受極值的大幅影響，極值會嚴重影響回歸線的結果。

線性回歸中假設 X 和 Y 之間的線性關係存在，表示為 $y = \alpha + \beta x$ ，其中 α 為截距， β 為斜率。

不同於線性回歸，指數回歸中假設因變數 Y 與自變數 X 之間的關係為指數形式，適用於因變量呈現非線性增長或衰減的情況，常用於描述成長率或減少率隨自變量變化的模式。

指數型回歸表示為 $y = \alpha e^{\beta x}$ ，其中 α 和 β 是回歸參數，且 $\beta < 0$ 。

研究者可以根據不同數據的特性及研究分析目的來選擇適合的回歸模型，以達到最佳的分析預測結果。

linear regression

0 9735 linearexponential regression.js:12
1 4597 linearexponential regression.js:12
2 2176 linearexponential regression.js:12
3 1024 linearexponential regression.js:12
4 483 linearexponential regression.js:12
5 229 linearexponential regression.js:12
6 108 linearexponential regression.js:12
7 52 linearexponential regression.js:12
8 24 linearexponential regression.js:12
9 11 linearexponential regression.js:12
10 6 linearexponential regression.js:12
alpha 5120.636363636364 beta -688.7636363636364
linearexponential regression.js:14
correlation is -0.757019814984881

exponential regression.

a 9620.329408784626 b -0.7461780351670065 linearexponential regression.js:30
a 9620.329408784626 b -0.7461780351670065 linearexponential regression.js:21
0 9620.329408784626 linearexponential regression.js:22
1 4561.723310041821 linearexponential regression.js:22
2 2163.056863559917 linearexponential regression.js:22
3 1025.6683005508223linearexponential regression.js:22
4 486.34665157320825 linearexponential regression.js:22
5 230.61360614288708 linearexponential regression.js:22
6 109.35129329295124 linearexponential regression.js:22
7 51.85169056084271 linearexponential regression.js:22
8 24.586794843062844 linearexponential regression.js:22
9 11.658452677555992 linearexponential regression.js:22
10 5.528151176368649 linearexponential regression.js:22