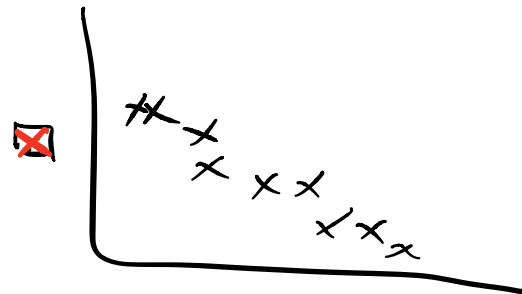
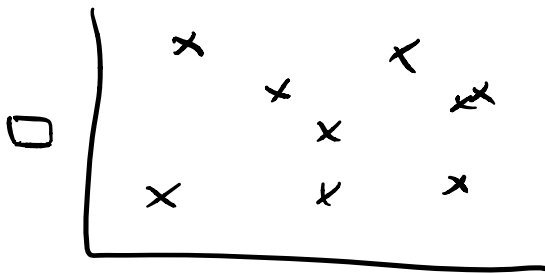
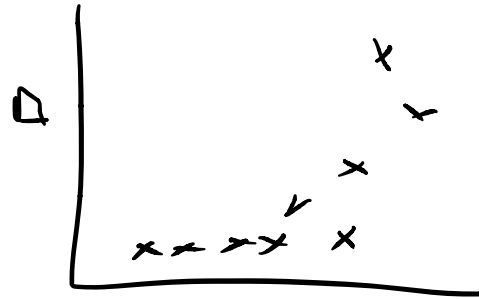
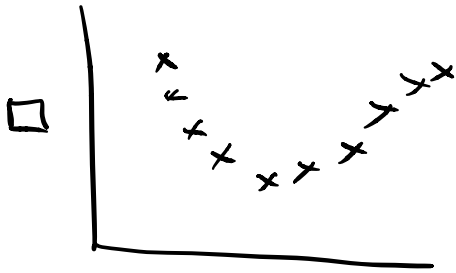


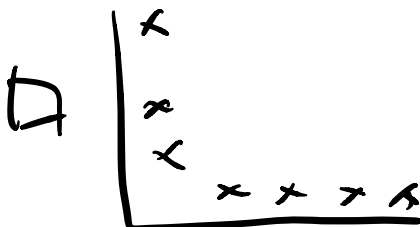
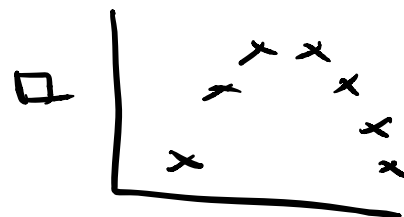
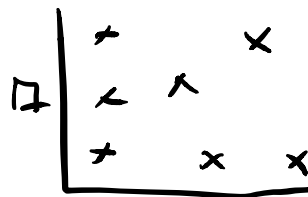
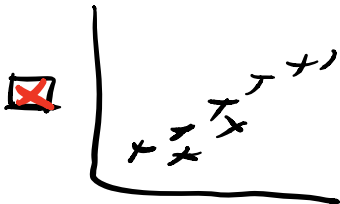
PROBLEM #1

Which plots would a linear fit make sense for



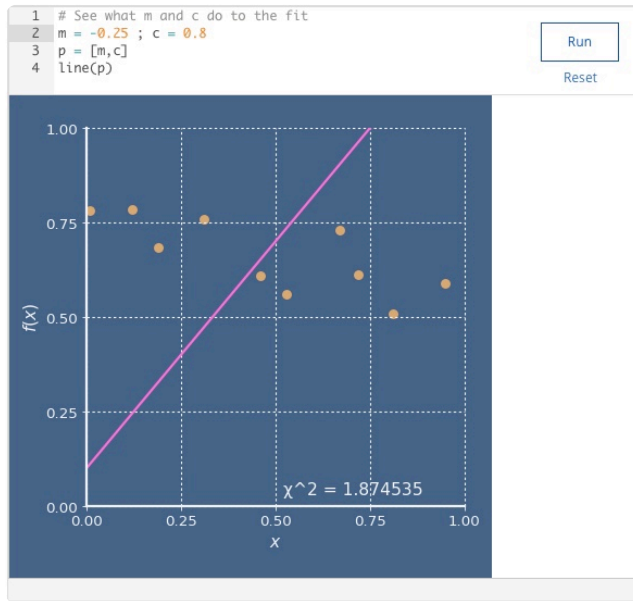
PROBLEM #2.

Which plots would make sense for a linear fit.



PROBLEM #3

Find reasonably optimal parameters for m and c.

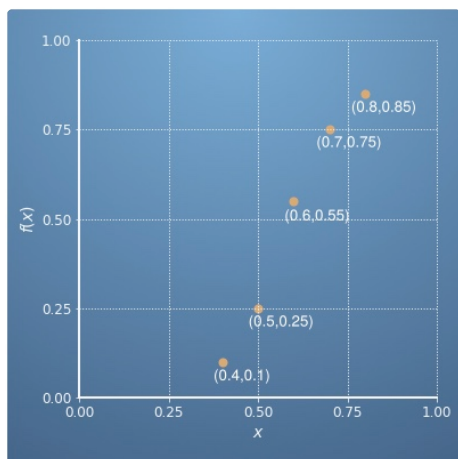


$$m = -0.27$$

$$c = 0.8$$

PROBLEM #4

Use a linear model and gradients to form a minimization problem, use the minimization formula to find the parameters m and c which best fit the five points below.



The Chi-squared test can be used to check goodness of fit,

$$\chi^2 = \sum_i (y_i - mx_i - c)^2$$

By forming a Lagrangian, then formulating it's gradient ∇L , then solving the system of equations can find general solutions for the parameters m and c,

$$m = \frac{\sum_i (x_i - \bar{x}) y_i}{\sum_i (x_i - \bar{x})^2} \quad c = \bar{y} - m\bar{x}$$

$$\bar{x} = (0.4 + 0.5 + 0.6 + 0.7 + 0.8) \times 5^{-1}$$

$$= 0.6$$

$$\bar{y} = (0.1 + 0.25 + 0.55 + 0.75 + 0.85) \times 5^{-1}$$

$$= 0.5$$

$$m = \frac{\begin{matrix} -0.02 & & -0.025 & & 0 \\ (0.4 - 0.6) \times 0.1 & + & (0.5 - 0.6) \times 0.25 & + & (0.6 - 0.6) \times 0.55 \\ & + & (0.7 - 0.6) \times 0.75 & + & (0.8 - 0.6) \times 0.85 \end{matrix}}{\begin{matrix} (0.4 - 0.6)^2 \times 0.04 & + & (0.5 - 0.6)^2 \times 0.01 & + & (0.6 - 0.6)^2 \times 0.2 \\ + & (0.7 - 0.6)^2 \times 0.01 & + & (0.8 - 0.6)^2 \times 0.04 \end{matrix}}$$

$$= 2$$

$$c = 0.5 - (m)0.6$$

$$= 0.5 - 2(0.6)$$

$$= -0.7$$

PROBLEM #5

Implement linear fit.

```

1 # Here the function is defined
2 def linfit(xdat,ydat):
3     # Here xbar and ybar are calculated
4     xbar = np.sum(xdat)/len(xdat)
5     ybar = np.sum(ydat)/len(ydat)
6
7     # Insert calculation of m and c here. If nothing is here the data will
8     # be plotted with no linear fit
9     xdelta = [delta - xbar for delta in xdat]
10    m = np.sum([x*y for x,y in zip(xdelta, ydat)]) / np.sum([s**2 for s in
11    xdelta])
12    c = ybar - m*xbar
13
14    # Return your values as [m, c]
15    return [m, c]
16
17 # Produce the plot - don't put this in the next code block
18 line()

```

Run

Reset

PROBLEM #6

Use `stats.linregress(xdat, ydat)`

```
1 from scipy import stats
2
3 # Use the stats.linregress() method to evaluate regression
4 regression = stats.linregress(xdat, ydat)
5
6 line(regression)
```

Run

Reset