Dylan Rohan - a1844790

Assessment 2: Functions, Probability and Linear Algebra

In [6]:
```python
# Importing the modules needed for this assignment
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import numpy as np
import scipy.stats as stats
import plotly.express as px
import sklearn.metrics
%matplotlib inline
```

Question 1

Question 1.a.i)

I interpret this scenario as follows: We have distinct 'containers' (people), and the 'objects' (scores) are not identical because the ranges differ between people.

Interpretted this way:

$$n = 10$$
$$r = 5$$

$$combinations = 10^5$$
$$combinations = 100000$$

Five people with ten different options (assuming no two scorers put the same weight on any of the scores, eg. judge 1 and judge 2 having an equivalent threshold on were a score of 2 sits) can produce 10,000 combinations. However, I'm yet to see a data scientist account for the differences in individual scoring interpretation. Which makes me think that as a data scientist the order typically doesn't matter. With this interpretation:

$$n = 10$$
$$r = 5$$
$$\binom{r + (n-1)}{r} = \binom{5 + (10-1)}{5} = \binom{14}{5}$$

$$\binom{14}{5} = \frac{14!}{(14-5)!5!} = \frac{14 \times 13 \times 12 \times 11 \times 10 \times 9!}{9!5!}$$
$$= 2002$$

This is interesting because it implies that a data scientist will only see 2002 possibilities when in reality there could be as many as 100,000!

Question 1.a.ii)

Order is not important because Tom, Dick and Harry is the same as Dick, Harry and Tom (a combination question), and repetition is not possible.

$$^nC_r = \frac{n!}{(n-r)!r!}$$

$$^{25}C_8 = \frac{25!}{(25-8)!8!}$$

$$= \frac{25 \times 24 \times 23 \times 22 \times 21 \times 20 \times 19 \times 18 \times 17!}{17! \times 8!}$$
$$= 1,081575$$

There are 1,081575 possible groups

Question 1.a.iii)

This appears too relate to the line "The number of ways of allocating r identical objects to n containers" in the course material. However, I'm not sure if INDIVIDUAL popcorn pieces is meant to imply that they are not identical.

If they are not identical: On your first selection you have three options, same for your second, and third, etc.

$$combinations = 3^{500}$$
$$combinations = 3.636 \times 10^{238}$$

However, it's possible that the popcorn pieces are not meant to be interpretted as distinct. In which case:

$$\binom{500+(3-1)}{3} = \frac{502!}{(502-3)!3!} = \frac{502 \times 501 \times 500 \times 499!}{499!3!}$$
$$= 20958500$$

There are $r^n = 3.636 \times 10^{238}$ ways to distribute 500 distinct peices of popcorn into three bowls, but only 20958500 ways to distribute 500 pieces of identical popcorn into three bowls; I suspect the course is after the second response but I'm having trouble interpretting the question.

⊡

Question 1.a.iv)

$$\text{Number of movies } = 10+8+6 = 24$$
$$\text{Probability of a given genre:}$$

$$\mathrm{P}(H) = \frac{10}{24} = \frac{5}{12}$$
$$\Pr(C) = 8/24 = \frac{1}{3}$$
$$\Pr(R) = 6/24 = \frac{5}{4}$$

$$\text{Number of movies with positive reviews } = 3+4+4 = 11$$
$$\text{Probability of a positive reiew, and genre given a positive review:}$$

$$\Pr(P) = 11/24$$
$$\Pr(H \backslash P) = \frac{3}{11}$$
$$\Pr(C \backslash P) = \frac{4}{11}$$
$$\Pr(R \backslash P) = \frac{4}{11}$$

$$\text{Probability of a positive review positive review for a given genre}$$

$$\Pr(P \backslash H) = \frac{3}{10}$$
$$\Pr(P \backslash C) = \frac{4}{8} = \frac{1}{2}$$
$$\Pr(P \backslash R) = \frac{4}{6} = \frac{2}{3}$$

As above $\Pr(H \backslash P) = 3/11$ but you probably want to see:

$$\Pr(H \backslash P) = \frac{\Pr(H) \times \Pr(P \backslash H)}{\Pr(P)} = \frac{(\frac{5}{12}) \times (\frac{3}{10})}{(\frac{11}{24})} = \frac{3}{11}$$

The probability that you are watching a horror movie is $\frac{3}{11}$.

Question 1.b.i)

In [2]:
```
words = ['recommend', 'hilarious', 'obvious', 'problems', 'awkward', 'boredom', 'C
count_positive = [81, 62, 34, 31, 8, 0, 216]
count_negative = [57, 19, 62, 30, 6, 12, 186]
count_total =   [138, 81, 96, 61, 14, 12, 402]


Qb = pd.concat([pd.Series(words, name = 'Words'),
                  pd.Series(count_positive, name = 'Count (+ve)'),
                  pd.Series(count_negative, name = 'Count (-ve)'),
                  pd.Series(count_total, name = 'Count (Total)')], axis=1)
print("Table 1")
Qb
```

```
Table 1
```

Out[2]:

	Words	Count (+ve)	Count (-ve)	Count (Total)
0	recommend	81	57	138
1	hilarious	62	19	81
2	obvious	34	62	96
3	problems	31	30	61
4	awkward	8	6	14
5	boredom	0	12	12
6	Column totals	216	186	402

In [3]:
```
print("See below the conditional probabilities table:")
Qb["Count (+ve)"] =Qb["Count (+ve)"].div(216)
Qb["Count (-ve)"] =Qb['Count (-ve)'].div(186)
```

```
Qb["Count (Total)"] =Qb["Count (Total)"].div(402)
Qb.iloc[0:6,:]
```

See below the conditional probabilities table:

Out[3]:

	Words	Count (+ve)	Count (-ve)	Count (Total)
0	recommend	0.375000	0.306452	0.343284
1	hilarious	0.287037	0.102151	0.201493
2	obvious	0.157407	0.333333	0.238806
3	problems	0.143519	0.161290	0.151741
4	awkward	0.037037	0.032258	0.034826
5	boredom	0.000000	0.064516	0.029851

Question 1.b.ii)

$$\begin{aligned} P(\text{ -ve}\backslash\text{"awkward" }) &= \frac{\Pr(-ve) \times \Pr(\text{ "awkward" } \backslash \text{ -ve})}{\Pr(\text{ "awkward"})} \\ &= \frac{\left(\frac{186}{402}\right) \times 0.0322258}{0.034826} \\ &= 0.429 \end{aligned}$$

There is a 42.9% chance of a review being negative given it contains the word 'awkward'.

Question 1.b.iii)

$H =$ hilarious $O =$ obvious $P =$ problems

$$\begin{aligned} \Pr(+ve\backslash H, O, P) &= \frac{\Pr(H\backslash + ve) \times \Pr(O\backslash + ve) \times \Pr(P\backslash + ve) \times \Pr(+ve)}{\Pr(H, O, P)} \\ &= \frac{0.287037 \times 0.157407 \times 0.143519 \times \left(\frac{216}{402}\right)}{\Pr(H, 0, P)} \\ &= \frac{0.003484}{\Pr(H, O, P)} \end{aligned}$$

$$\begin{aligned} \Pr(-ve\backslash H, O, P) &= \frac{\Pr(H\backslash - ve) \times \Pr(O\backslash - ve) \times \Pr(P\backslash - ve) \times \Pr(\text{-ve })}{\Pr(H, O, P)} \\ &= \frac{0.102151 \times \left(\frac{1}{3}\right) \times 0.161290 \times \left(\frac{186}{402}\right)}{\Pr(H, O, P)} \\ &= \frac{0.002541}{\Pr(H, O, P)} \end{aligned}$$

Such a review is more likely to be positive as the numerator is larger, and so $\Pr(+ve\backslash H, O, P)$ will always be larger than $\Pr(-ve\backslash H, O, P)$

Question 1.b.iv)

$B=$ "boredom" $\dot{R}=$ "recommend"

$$\Pr(\text{ -ve }\backslash B,R)=\frac{\Pr(B\backslash-ve)\times\Pr(R\backslash-ve)\times\Pr(\text{ -ve })}{\Pr(B,R)}$$

$$=\frac{0\times 0.375000\times\left(\frac{216}{402}\right)}{\Pr(B,R)}$$

$$=\frac{0}{\Pr(B,R)}$$

$$\Pr(-ve\mid B,R)=\frac{\Pr(B\backslash-ve)\times\Pr(R\backslash-ve)\times\Pr(-ve)}{\Pr(B,R)}$$

$$=\frac{0.064516\times 0.306452\times\left(\frac{186}{407}\right)}{\Pr(B,R)}$$

$$=\frac{0.009035}{\Pr(B,R)}$$

Based on the table, $\Pr(\text{ -ve }\backslash B,R)$ is equal to zero, so the review will be positive.

Question 1.b.v)

As the probability of 'boredom' occuring in a positive review is zero, it incorrectly assumes all new reviews with 'boredom' couldn't possibly be positive. This technique is therefore highly dependent on the assumption that the trainning data is representative of all possibilities.

Question 1.b.vi)

In [4]:

```
words = ['recommend', 'hilarious', 'obvious', 'problems', 'awkward', 'boredom', 'C
count_positive = [81, 62, 34, 31, 8, 0, 221]
count_negative = [57, 19, 62, 30, 6, 12, 191]
count_total =   [139, 82, 97, 62, 15, 13, 413]


Qb = pd.concat([pd.Series(words, name = 'Words'),
                  pd.Series(count_positive, name = 'Count (+ve)'),
                  pd.Series(count_negative, name = 'Count (-ve)'),
                  pd.Series(count_total, name = 'Count (Total)')], axis=1)

print("Table 2")
Qb.iloc[:,1:] = Qb.iloc[:,1:]+1
Qb
```

```
Table 2
```

Out[4]:

	Words	Count (+ve)	Count (-ve)	Count (Total)
0	recommend	82	58	140
1	hilarious	63	20	83
2	obvious	35	63	98
3	problems	32	31	63
4	awkward	9	7	16
5	boredom	1	13	14
6	Column totals	222	192	414

In [5]:

```
print("See below the conditional probabilities for table2 :")
Qb["Count (+ve)"] =Qb["Count (+ve)"].div(222)
Qb["Count (-ve)"] =Qb['Count (-ve)'].div(192)
Qb["Count (Total)"] =Qb["Count (Total)"].div(414)
Qb.iloc[0:6,:]
```

```
See below the conditional probabilities for table2 :
```

Out[5]:

	Words	Count (+ve)	Count (-ve)	Count (Total)
0	recommend	0.369369	0.302083	0.338164
1	hilarious	0.283784	0.104167	0.200483
2	obvious	0.157658	0.328125	0.236715
3	problems	0.144144	0.161458	0.152174
4	awkward	0.040541	0.036458	0.038647
5	boredom	0.004505	0.067708	0.033816

$$\begin{aligned}\Pr(+ve\backslash B,R) &= \frac{\Pr(B\backslash +ve)\times\Pr(R\backslash +ve)\times\Pr(+ve)}{\Pr(B,R)}\\ &= \frac{0.004505\times 0.369369\times\left(\frac{222}{414}\right)}{\Pr(B,R)}\\ &= \frac{0.000892}{\mathrm{Pr(B,\,R)}}\end{aligned}$$

$$\begin{aligned}\Pr(-ve\backslash B,R) &= \frac{\Pr(B\backslash -ve)\times\Pr(R\backslash -ve)\times\Pr(-ve)}{\Pr(B,R)}\\ &= \frac{0.067708\times 0.302083\times\left(\frac{192}{414}\right)}{\Pr(B,R)}\\ &= \frac{0.009486}{\Pr(B,R)}\end{aligned}$$

Such a review is more likely to be negative as $\Pr(-ve\backslash B,R)$ will always be greater than $\Pr(+ve\backslash B,R)$ as a result of the difference in numerators but a common denominator.

Question 2

Question 2.a.i)

$$X = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 3 & 9 \\ 1 & 6 & 36 \end{bmatrix}$$

$$X^\top X = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 3 & 6 \\ 0 & 9 & 36 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 1 & 3 & 9 \\ 1 & 6 & 36 \end{bmatrix}$$

$$= \begin{bmatrix} 1+1+1 & 0+3+6 & 0+9+36 \\ 0+3+6 & 0+9+36 & 0+27+216 \\ 0+9+36 & 0+27+216 & 0+81+1296 \end{bmatrix}$$

$$= \begin{bmatrix} 3 & 9 & 45 \\ 9 & 45 & 243 \\ 45 & 243 & 1377 \end{bmatrix}$$

Question 2.a.ii)

$$\left(X^\top X\right)^{-1} = \left[\begin{array}{ccc|ccc} 3 & 9 & 45 & 1 & 0 & 0 \\ 9 & 45 & 243 & 0 & 1 & 0 \\ 45 & 243 & 1377 & 0 & 0 & 1 \end{array}\right]$$

$$R_1 \to R_1/3 = \left[\begin{array}{ccc|ccc} 1 & 3 & 15 & \frac{1}{3} & 0 & 0 \\ 9 & 45 & 243 & 0 & 1 & 0 \\ 45 & 243 & 1377 & 0 & 0 & 1 \end{array}\right]$$

$$R_2 \to R_2 - 9R_1 = \left[\begin{array}{ccc|ccc} 1 & 3 & 15 & \frac{1}{3} & 0 & 0 \\ 0 & 18 & 108 & -3 & 1 & 0 \\ 45 & 243 & 1377 & 0 & 0 & 1 \end{array}\right]$$

$$R_3 \to R_3 - 45 \times R_1 = \left[\begin{array}{ccc|ccc} 1 & 3 & 15 & \frac{1}{3} & 0 & 0 \\ 0 & 18 & 108 & -3 & 1 & 0 \\ 0 & 108 & 702 & -15 & 0 & 1 \end{array}\right]$$

$$R_2 \to R_2/18 = \left[\begin{array}{ccc|ccc} 1 & 3 & 15 & \frac{1}{3} & 0 & 0 \\ 0 & 1 & 6 & \frac{-1}{6} & \frac{1}{18} & 0 \\ 0 & 108 & 702 & -15 & 0 & 1 \end{array}\right]$$

$$R_3 \to R_3 - 108R_2 = \left[\begin{array}{ccc|ccc} 1 & 3 & 15 & \frac{1}{3} & 0 & 0 \\ 0 & 1 & 6 & \frac{-1}{6} & \frac{1}{18} & 0 \\ 0 & 0 & 54 & 3 & -6 & 1 \end{array}\right]$$

$$R_3 \to R_3/54 = \left[\begin{array}{ccc|ccc} 1 & 3 & 15 & \frac{1}{3} & 0 & 0 \\ 0 & 1 & 6 & \frac{-1}{6} & \frac{1}{18} & 0 \\ 0 & 0 & 1 & \frac{1}{18} & \frac{-1}{9} & \frac{1}{54} \end{array}\right]$$

$$R_2 \to R_2 - R_3 = \left[\begin{array}{ccc|ccc} 1 & 3 & 15 & \frac{1}{3} & 0 & 0 \\ 0 & 1 & 5 & \frac{-2}{9} & \frac{1}{6} & -\frac{1}{54} \\ 0 & 0 & 1 & \frac{1}{18} & \frac{-1}{9} & \frac{1}{54} \end{array}\right]$$

$$R_1 \to R_1 - 3R_2 = \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & \frac{-1}{2} & \frac{1}{18} \\ 0 & 1 & 5 & \frac{-2}{9} & \frac{1}{6} & \frac{-1}{54} \\ 0 & 0 & 1 & \frac{1}{18} & \frac{-1}{9} & \frac{1}{54} \end{array}\right]$$

$$R_2 \to R_2 - 5R_3 = \left[\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & \frac{-1}{2} & \frac{1}{18} \\ 0 & 1 & 0 & \frac{-1}{2} & \frac{13}{18} & \frac{-1}{9} \\ 0 & 0 & 1 & \frac{1}{18} & \frac{-1}{9} & \frac{1}{54} \end{array}\right]$$

$$\left(X^\top X\right)^{-1} = \begin{bmatrix} 1 & \frac{-1}{2} & \frac{1}{18} \\ \frac{-1}{2} & \frac{13}{18} & \frac{-1}{9} \\ \frac{1}{18} & \frac{-1}{9} & \frac{1}{54} \end{bmatrix}$$

iii) Write out the formula for calculating the regression coefficients $\hat{\beta}$, and determine the order of $\hat{\beta}$, including reasoning. Then, calculate $\hat{\beta}$. You may use R/Python/a calculator for this last step.

Question 2.a.iii)

$$\hat{\beta} = \left(x^\top x\right)^{-1} x^\top y$$

$$\hat{\beta} = \begin{bmatrix} 1 & \frac{-1}{2} & \frac{1}{18} \\ \frac{-1}{2} & \frac{13}{18} & \frac{-1}{9} \\ \frac{1}{18} & \frac{-1}{9} & \frac{1}{54} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 3 & 6 \\ 0 & 9 & 36 \end{bmatrix} \times \begin{bmatrix} 34 \\ 17 \\ 20 \end{bmatrix}$$

$$= \begin{bmatrix} 1+0+0 & 1-\frac{3}{2}+\frac{1}{2} & 1-3+2 \\ \frac{-1}{2}+0+0 & \frac{-1}{2}+\frac{13}{6}-1 & \frac{-1}{2}+\frac{13}{3}-4 \\ \frac{1}{18}+0+0 & \frac{1}{18}-\frac{1}{3}+\frac{1}{6} & \frac{1}{18}-\frac{2}{3}+\frac{2}{3} \end{bmatrix} \begin{bmatrix} 34 \\ 17 \\ 20 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ \frac{-1}{2} & \frac{2}{3} & \frac{-1}{6} \\ \frac{1}{18} & \frac{-1}{9} & \frac{1}{18} \end{bmatrix} \begin{bmatrix} 34 \\ 17 \\ 20 \end{bmatrix}$$

$$= \begin{bmatrix} 34+0+0 \\ -17+\frac{34}{3}-\frac{10}{3} \\ \frac{17}{9}-\frac{17}{9}+\frac{10}{9} \end{bmatrix}$$

$$\hat{\beta} = \begin{bmatrix} 34 \\ -9 \\ \frac{10}{9} \end{bmatrix}$$

Question 2.a.iv)

$$y = \beta_0 + \hat{\beta}_1 x_{1i}^2 + \hat{\beta} x_{2i} + \varepsilon_i$$
$$y = 34 - 9x_{1i}^2 + \frac{10}{9} x_{2i} + \varepsilon_i$$

$$\begin{aligned} &\text{When } (x, y) = (0, 34): \\ &34 = 34 + -9(0) + \tfrac{10}{9}(0) + \varepsilon_1 \\ &34 = 34 + \varepsilon_1 \quad \therefore \varepsilon_1 = 0 \end{aligned}$$

$$
\begin{aligned}
&\text{When } (x, y) = (3, 17): \\
&17 = 34 - 9(3) + \tfrac{10}{9}(9) + \varepsilon_2 \\
&17 = 34 - 27 + 10 + \varepsilon_2, \quad \therefore \varepsilon_2 = 0
\end{aligned}
$$

$$
\begin{aligned}
&\text{When } (x, y) = (6, 20): \\
&20 = 34 - 9(6) + \tfrac{10}{9}(36) + \varepsilon_3 \\
&20 = 34 - 54 + 40 + \varepsilon_3, \quad \therefore \varepsilon_3 = 0
\end{aligned}
$$

The error terms are zero because the residuals of each point are zero. The model perfectly predicts the equation because there can be no variation in y for any value of x, therfore there will be no noise or error associated with the curve.

Question 2.b

In [6]:

```
# Reading in the csv file
df = pd.read_csv('simulated_sentiment_data_t5.csv')
df
```

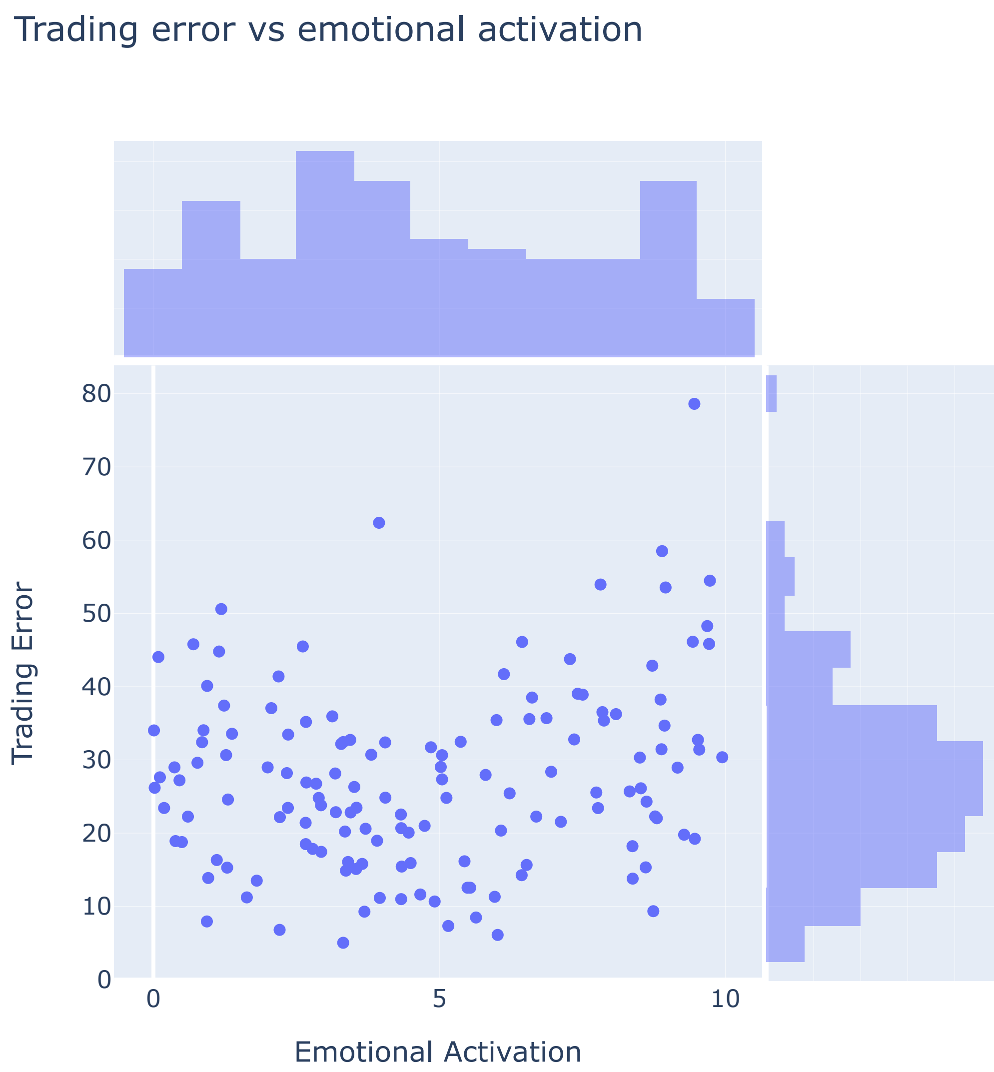
Out[6]:

	activation	trading_error
0	4.327314	22.544251
1	4.852512	31.717818
2	9.430645	46.129748
3	7.123846	21.546863
4	5.372109	32.469794
...	...	...
136	3.709443	20.600183
137	6.128338	41.701795
138	6.447178	46.103249
139	4.340873	15.436870
140	6.620448	38.509721

141 rows × 2 columns

In [7]:

```
# Observing the shape and spread of the data both as variables and as a relationsh
fig = px.scatter(df, x="activation", y="trading_error", marginal_x="histogram", ma
                  width=600, height=600,
                title = "Trading error vs emotional activation",
                labels={
                "trading_error": "Trading Error",  "activation": "Emotional Activa
            })
fig.show()
```




```
In [8]: # Creating the model that willl find a quadratic line of best fit (polynomial fit w
x = df.activation
y = np.sqrt(df.trading_error)
# This transformation of y reduced the mean squared error from 132.46666522507738 t
model = np.poly1d(np.polyfit(x, y, 2))
```

Question 2.b.i)

```
In [9]: # Displaying the equation for this curve
print("The equation for this curve is:\n")
print(model)
```

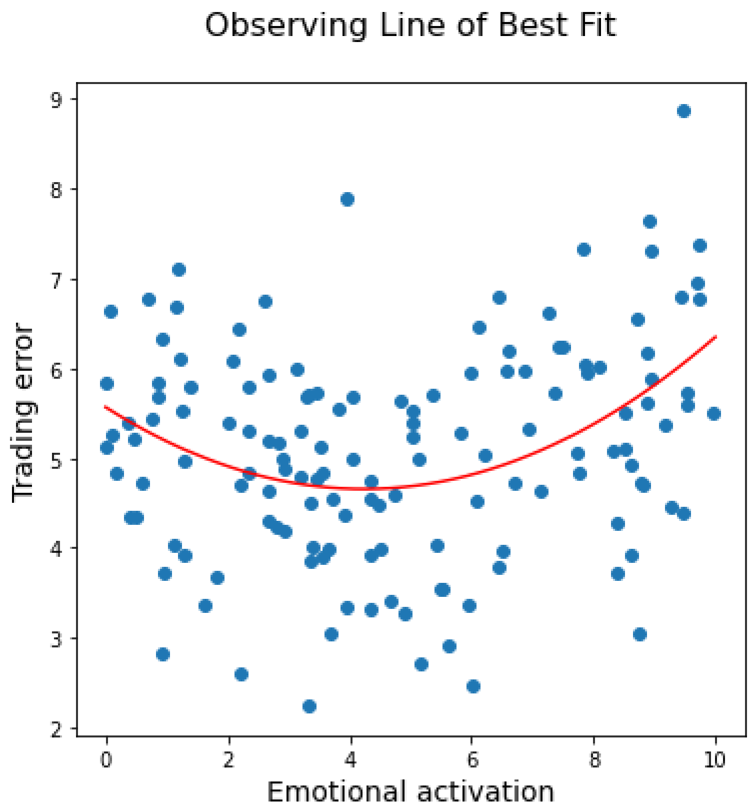
```
The equation for this curve is:

         2
0.05099 x - 0.4316 x + 5.567
```

Question 2.b.ii)

```
In [10]: # Mapping the regression line over a scatterplot
polyline = np.linspace(0, 10, 50)
fig = plt.figure(figsize=(6,6))
plt.scatter(x, y)
```

```
plt.plot(polyline, model(polyline), color ='r')
plt.ylabel("Trading error", size=14)
plt.xlabel("Emotional activation", size=14)
plt.title("Observing Line of Best Fit\n", size=16)
plt.show()
```



In [11]:
```
# Using the R-squared value to asses the fit
model2 = np.poly1d(np.polyfit(df.activation, df.trading_error, 2))

metrics = pd.DataFrame({'Model': ['Without transformations', 'with transformations
                        'MSE': [sklearn.metrics.mean_squared_error(df.trading_error
                                sklearn.metrics.mean_squared_error(y, model(x))],
                        'RMSE' : [np.sqrt(sklearn.metrics.mean_squared_error(df.tra
                                  np.sqrt(sklearn.metrics.mean_squared_error(y, mode
print("Below demonstrates the improvement obtained by taking the squareroot of trad
metrics
```

```
Below demonstrates the improvement obtained by taking the squareroot of trading er
ror
```

Out[11]:

	Model	MSE	RMSE
0	Without transformations	132.466665	11.509416
1	with transformations	1.228573	1.108410

Question 2.b.iii)

$$\sqrt{y} = \sqrt{25} = 0.05099x^2 - 0.4316x + 5.567$$
$$5 = 0.05099x^2 - 0.4316x + 5.567$$
$$0 = 0.05099x^2 - 0.4316x + 0.567$$
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$
$$x = \frac{-(-0.4316) \pm \sqrt{(-0.4316)^2 - 4 \times (.05099) \times (0.567)}}{2(0.05099)}$$
$$x = 1.626 \text{ and } x = 6.83829$$

As there are two values for x, the function doesn't pass the horizontal line test and so there cannot be an inverse. Usually you would let y=x, but when x has more than one value for a given y there is no way to obtain an inverse. An inverse would be possible if the domain were restricted to $[4.232202393, \infty)$ because in that scenario there is a single value of x for every y.