Question 1

To prove the given expression for the response function R_l using equations (1), (4), and (5), let's substitute the expressions for p_t and $C(l)$ into the definition of R_l from equation (2):

$$
\begin{aligned}
R_l &= \langle (p_{t+l} - p_t)\epsilon_t \rangle \\
&= \left\langle \left(\sum_{t'<t+l} \left[G\left((t+l)-t'\right) V_{t'}^{\alpha} \epsilon_{t'} \right] + \varepsilon_{t+l} \right) - \left(\sum_{t'<t} \left[G\left(t-t'\right) V_{t'}^{\alpha} \epsilon_{t'} \right] + \varepsilon_t \right) \right\rangle \epsilon_t.
\end{aligned}
$$

terms involving $G(t)$, $V_{t'}^{\alpha}$, and $\epsilon_{t'}$:

$$
\begin{aligned}
&\left(\sum_{t'<t+l} G\left((t+l)-t'\right) V_{t'}^{\alpha} \epsilon_{t'} \right) - \left(\sum_{t'<t} G\left(t-t'\right) V_{t'}^{\alpha} \epsilon_{t'} \right) \\
&= \sum_{0<t'\leq l} G(t') V_{t'}^{\alpha} \epsilon_{t'} - \sum_{0<t'} G(t') V_{t'}^{\alpha} \epsilon_{t'} + \sum_{t'>l} G(t') V_{t'}^{\alpha} \epsilon_{t'}.
\end{aligned}
$$

Using $C(l) \sim \bar{V}^{\alpha} c(l)$ (from equation 5):

$$
\begin{aligned}
&\left(\sum_{t'<t+l} G(t') V_{t'}^{\alpha} \epsilon_{t'} \right) - \left(\sum_{t'<t} G(t') V_{t'}^{\alpha} \epsilon_{t'} \right) \\
&= \bar{V}^{\alpha} \left[\sum_{0<t'\leq l} G(t') c(t'-l) + \sum_{t'>l} G(t') c(t'-l) - \sum_{0<t'} G(t') c(t') \right].
\end{aligned}
$$

Substituting into the definition of R_l:

$$
R_l \sim \left\langle \bar{V}^{\alpha} \left[\sum_{0<t'\leq l} G(t') c(t'-l) + \sum_{t'>l} G(t') c(t'-l) - \sum_{0<t'} G(t') c(t') \right] + \varepsilon_{t+l} - \varepsilon_t \right\rangle \epsilon_t
$$

Since ε_t and ε_{t+l} are independent of the summation terms, we can take them out of the ensemble average:

$$
R_l \sim \bar{V}^{\alpha} \left[\sum_{0<t'\leq l} G(t') c(t'-l) + \sum_{t'>l} G(t') c(t'-l) - \sum_{0<t'} G(t') c(t') \right] \langle \epsilon_{t+l} - \epsilon_t \rangle .
$$

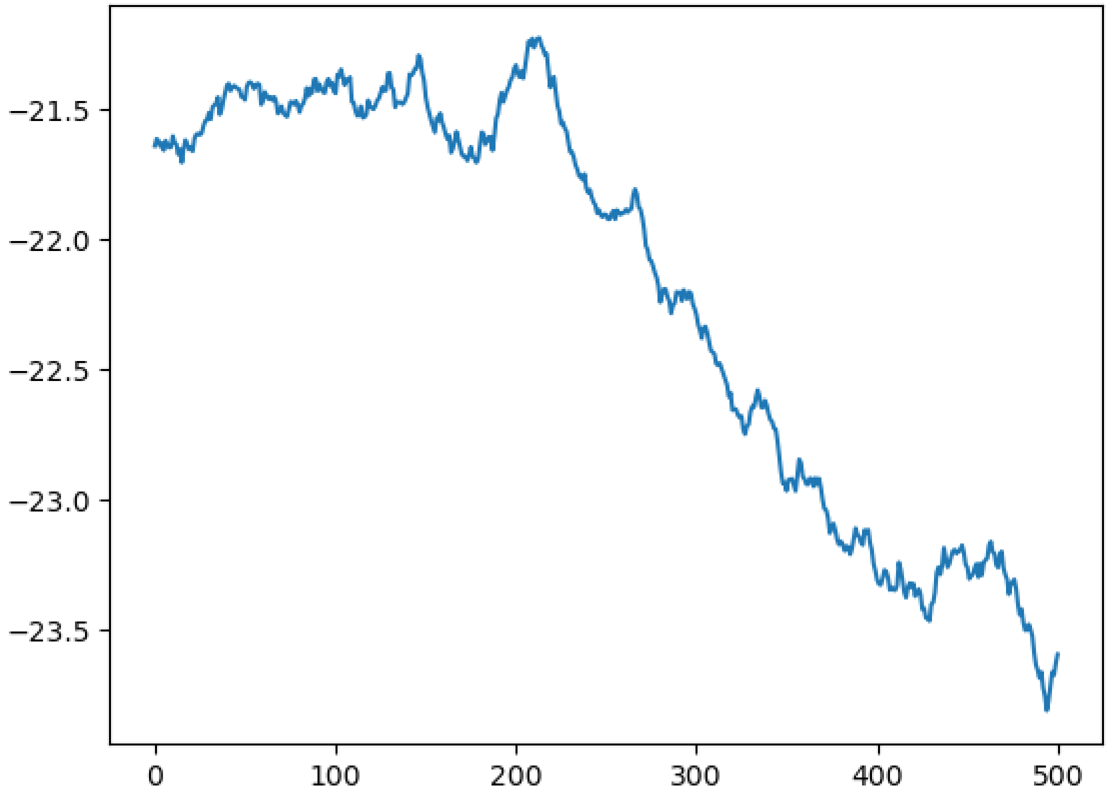
Using $\langle \epsilon_{t+l} - \epsilon_t \rangle = c(l)$:

$$
R_l \sim \bar{V}^{\alpha} \left[\sum_{0<t'\leq l} G(t') c(t'-l) + \sum_{t'>l} G(t') c(t'-l) - \sum_{0<t'} G(t') c(t') \right] c(l).
$$

Question 2

In [1]:
```
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
```

```
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")
dataset=pd.read_csv('pp1_md_201607_201607.csv')
dataset['WeightedAvgPrice']=dataset.groupby('Date')['Size'].transform(lambda
dataset['Midpoint']=(dataset['BP1']+dataset['SP1'])/2
dataset['PriceSpread']=dataset['SP1']-dataset['BP1']
response_function=[]
for lag in range(501):
    dataset['Response_modified']=(dataset.groupby('Date')['WeightedAvgPrice
    daily_response=dataset.groupby('Date')['Response_modified'].mean()
    response_function.append(daily_response.mean())
plt.plot(response_function)
plt.show()
```

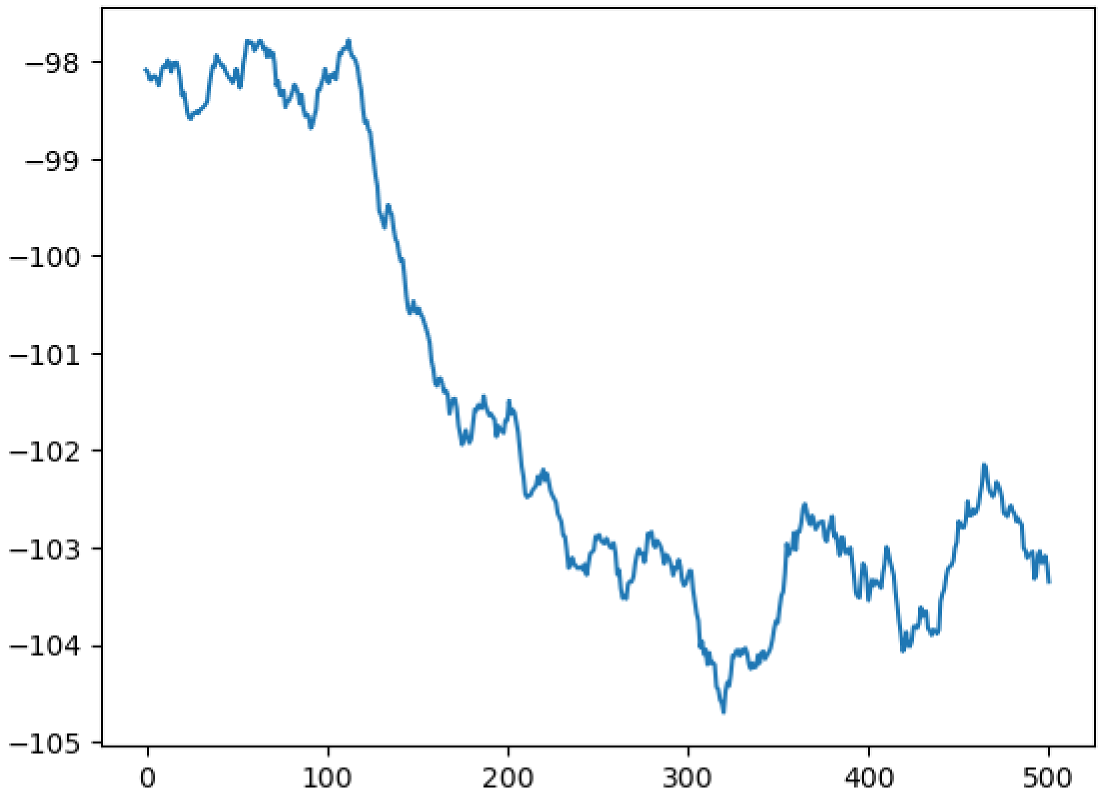


Question 3 -> Response Function is higher for higher trade sizes

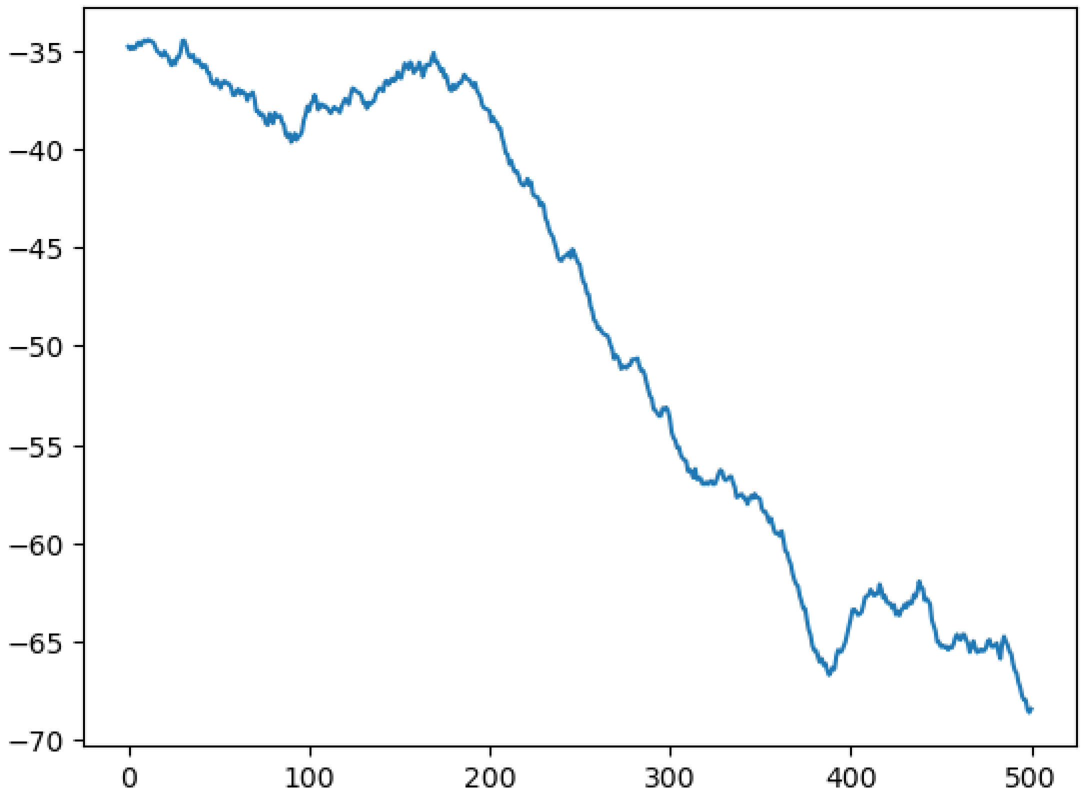
In [2]:

```
size_bins=[0,2,5,10,15,20,30,40,55,90,100000]
response_functions=[]
for i in range(len(size_bins)-1):
    size_bin_data=dataset[(dataset['Size']>size_bins[i])&(dataset['Size']<=s
    response_function_temp=[]
    for lag in range(501):
        size_bin_data['Response']=(size_bin_data.groupby('Date')['WeightedAv
        response_function_temp.append(size_bin_data.groupby('Date')['Respons
    response_functions.append(response_function_temp)
for i,size_bin_response in enumerate(response_functions):
    print(f'{size_bins[i]}<V<{size_bins[i+1]}:')
    plt.plot(size_bin_response)
    plt.show()
```

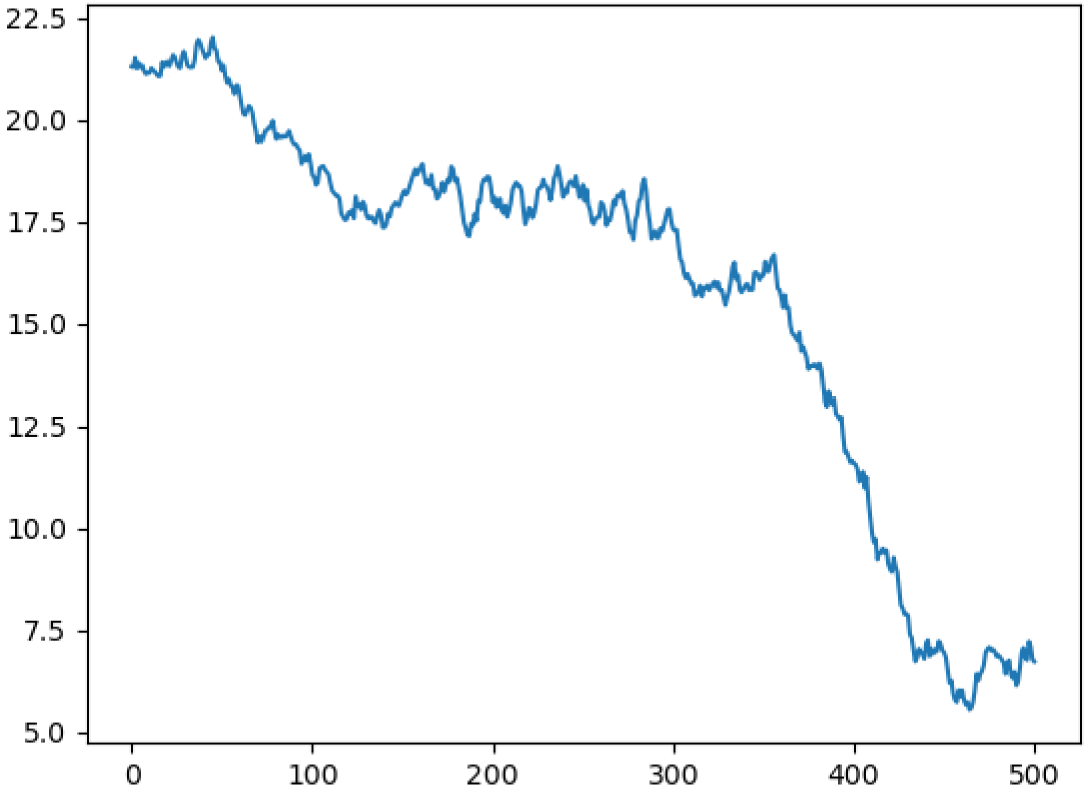
```
0<V<2:
```


−98
−99
−100
−101
−102
−103
−104
−105
0
100
200
300
400
500

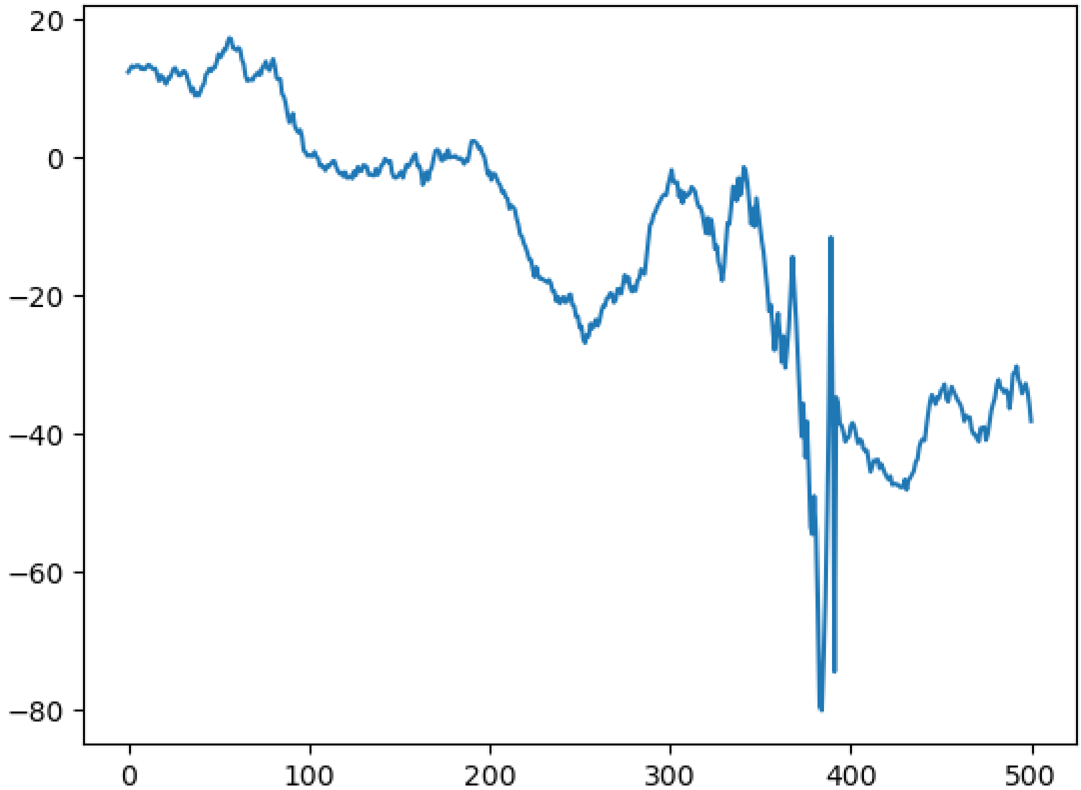
2<V<5:


−35
−40
−45
−50
−55
−60
−65
−70
0
100
200
300
400
500

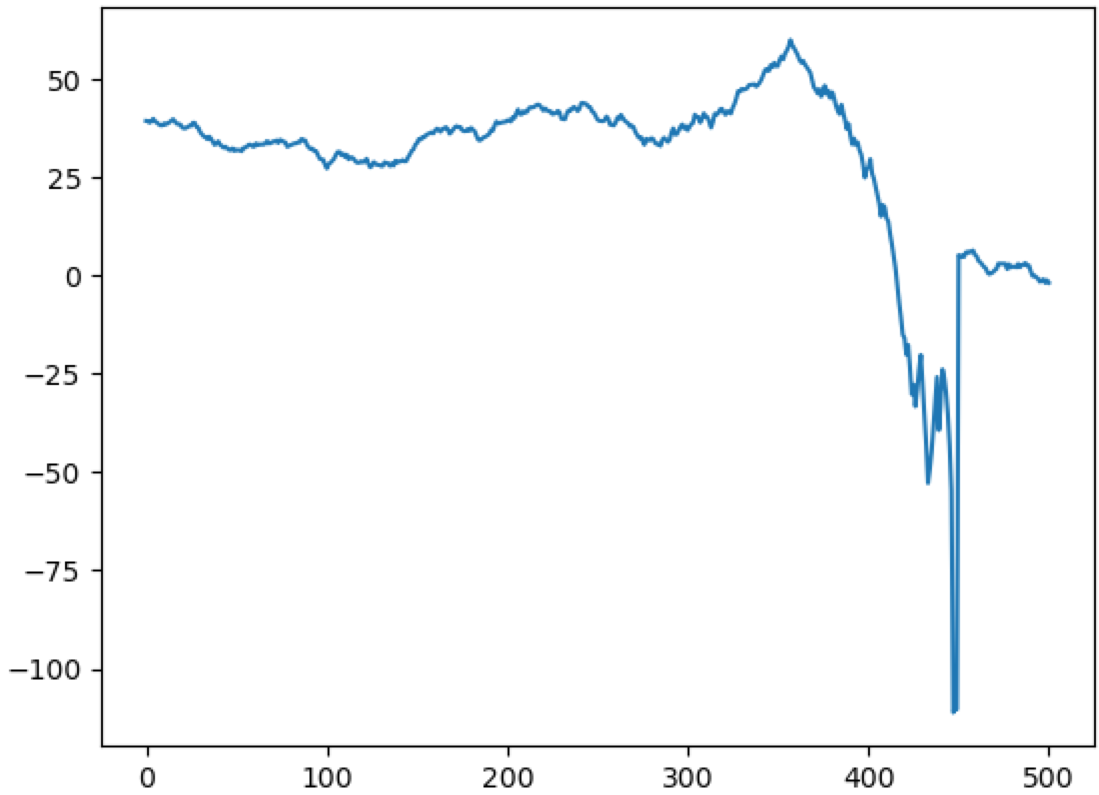
5<V<10:


22.5
20.0
17.5
15.0
12.5
10.0
7.5
5.0
0
100
200
300
400
500

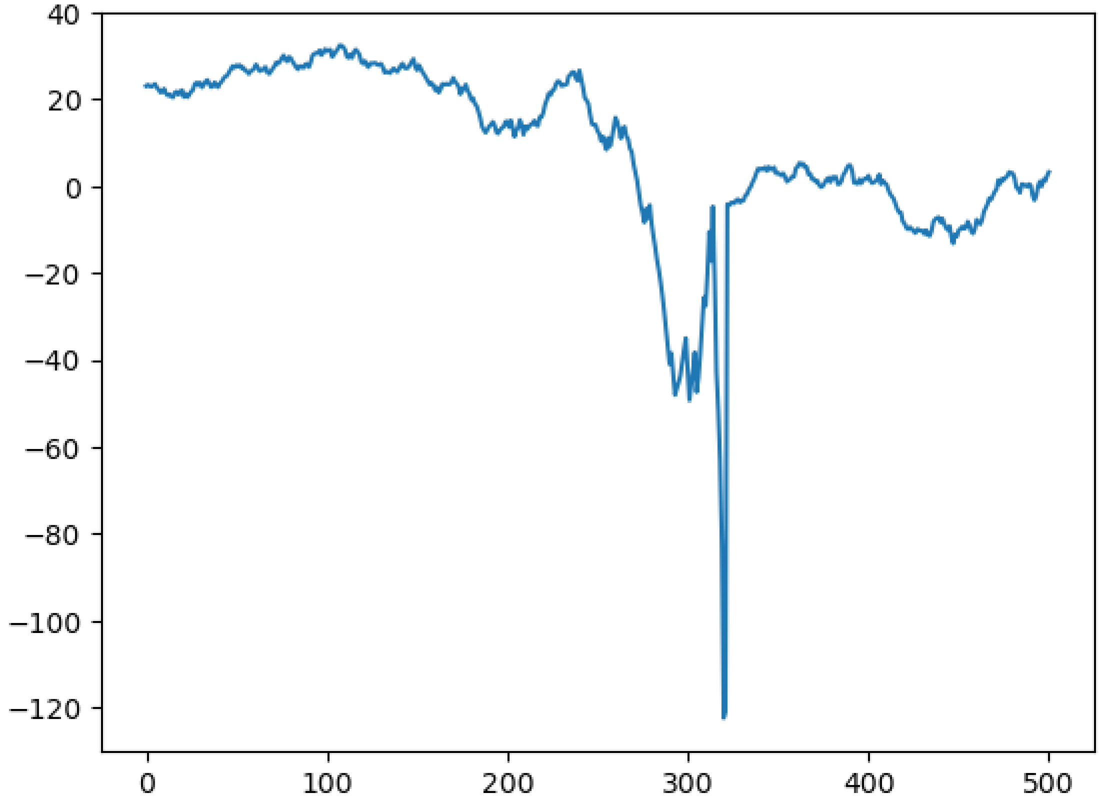
10<V<15:


20
0
−20
−40
−60
−80
0
100
200
300
400
500

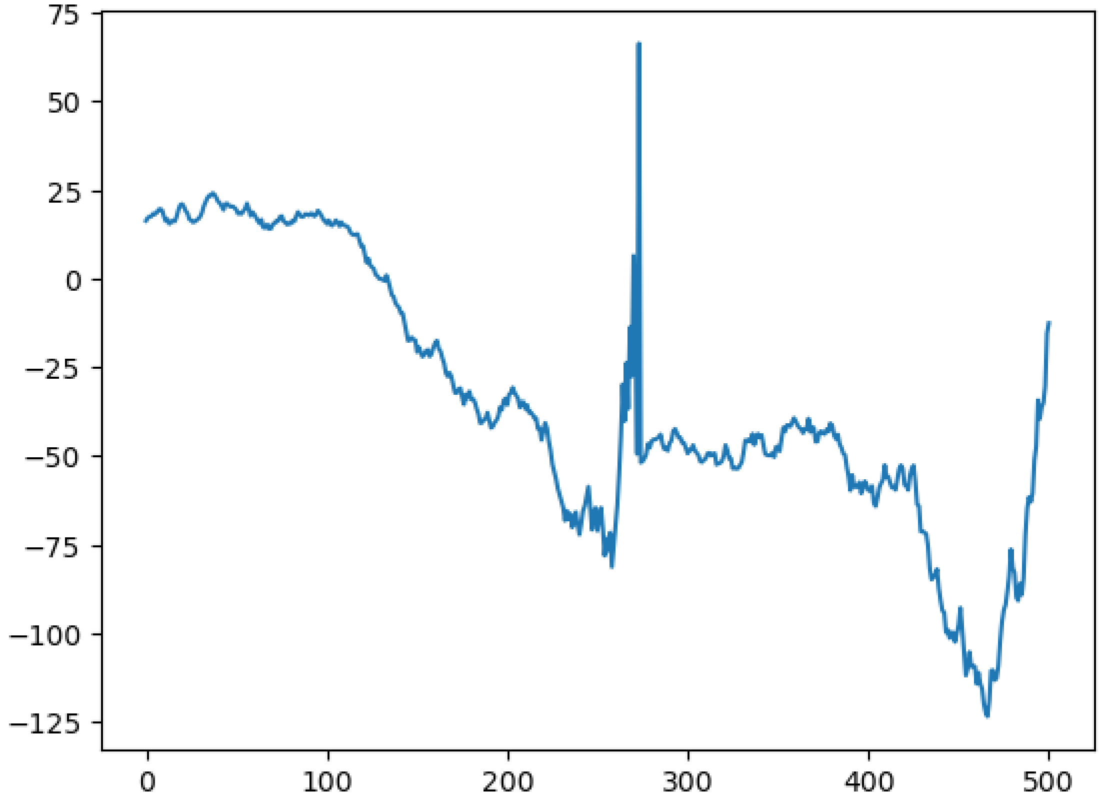
15<V<20:


50
25
0
−25
−50
−75
−100
0
100
200
300
400
500

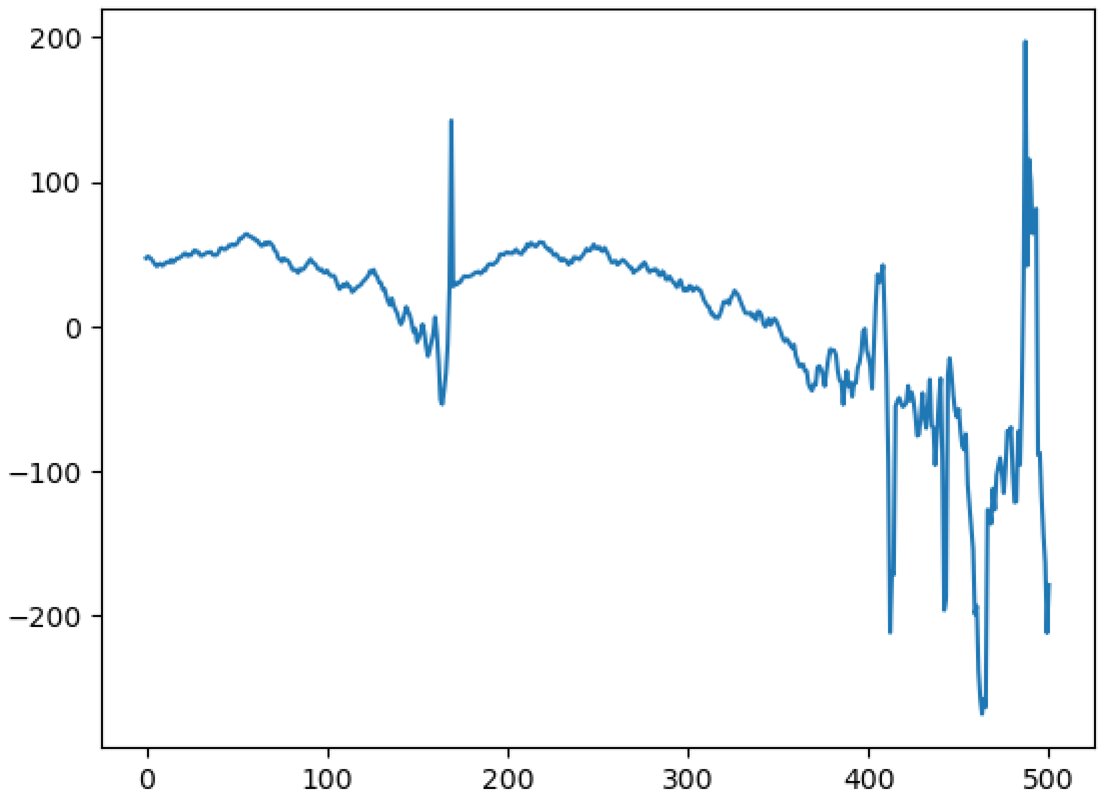
20<V<30:


40
20
0
−20
−40
−60
−80
−100
−120
0
100
200
300
400
500

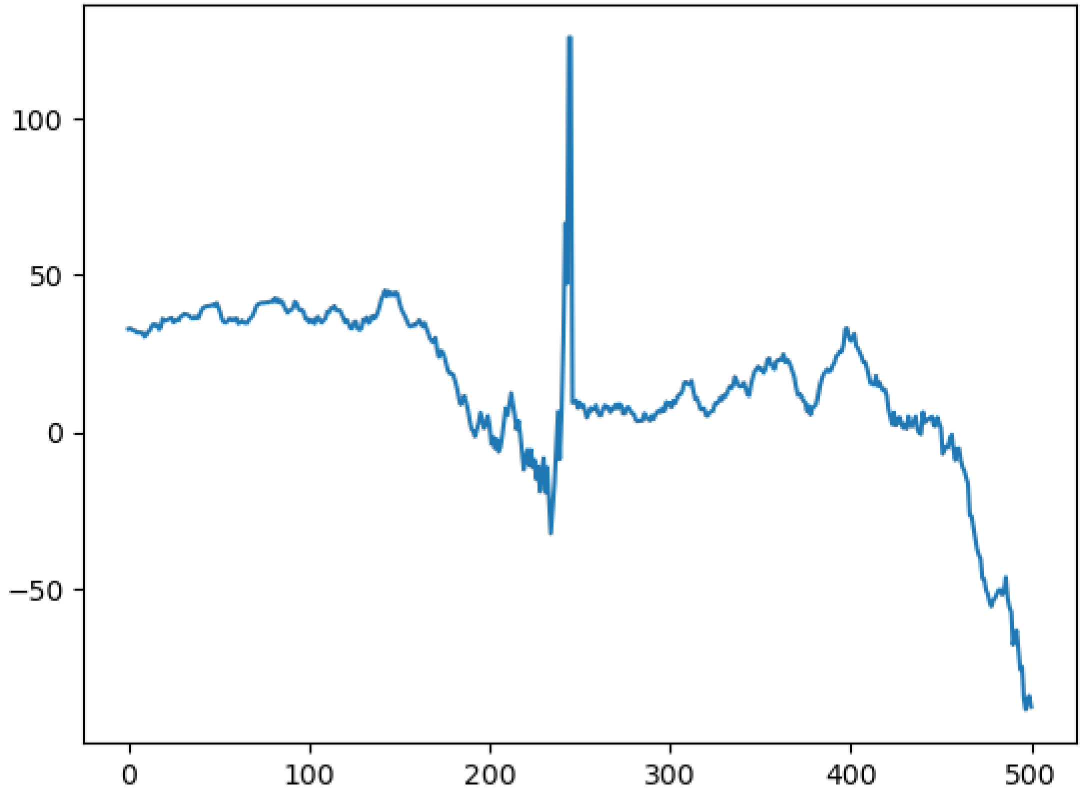
30<V<40:


75
50
25
0
−25
−50
−75
−100
−125
0
100
200
300
400
500

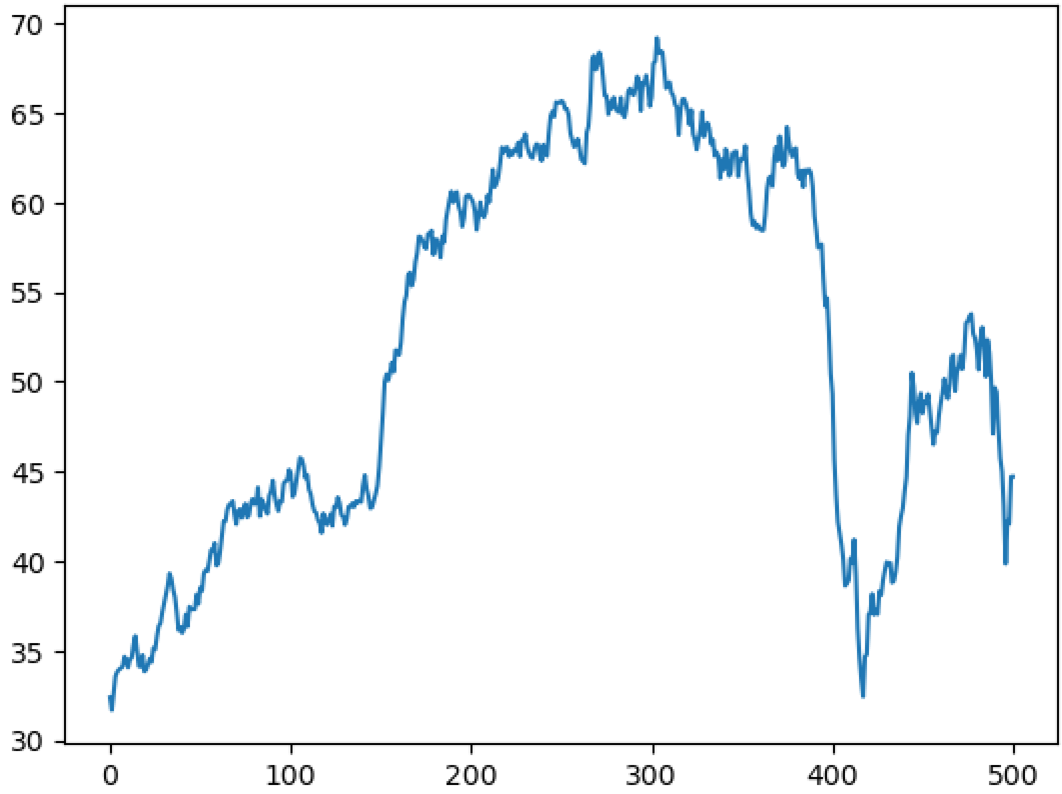
40<V<55:


200
100
0
−100
−200
0
100
200
300
400
500

55<V<90:



90<V<100000:

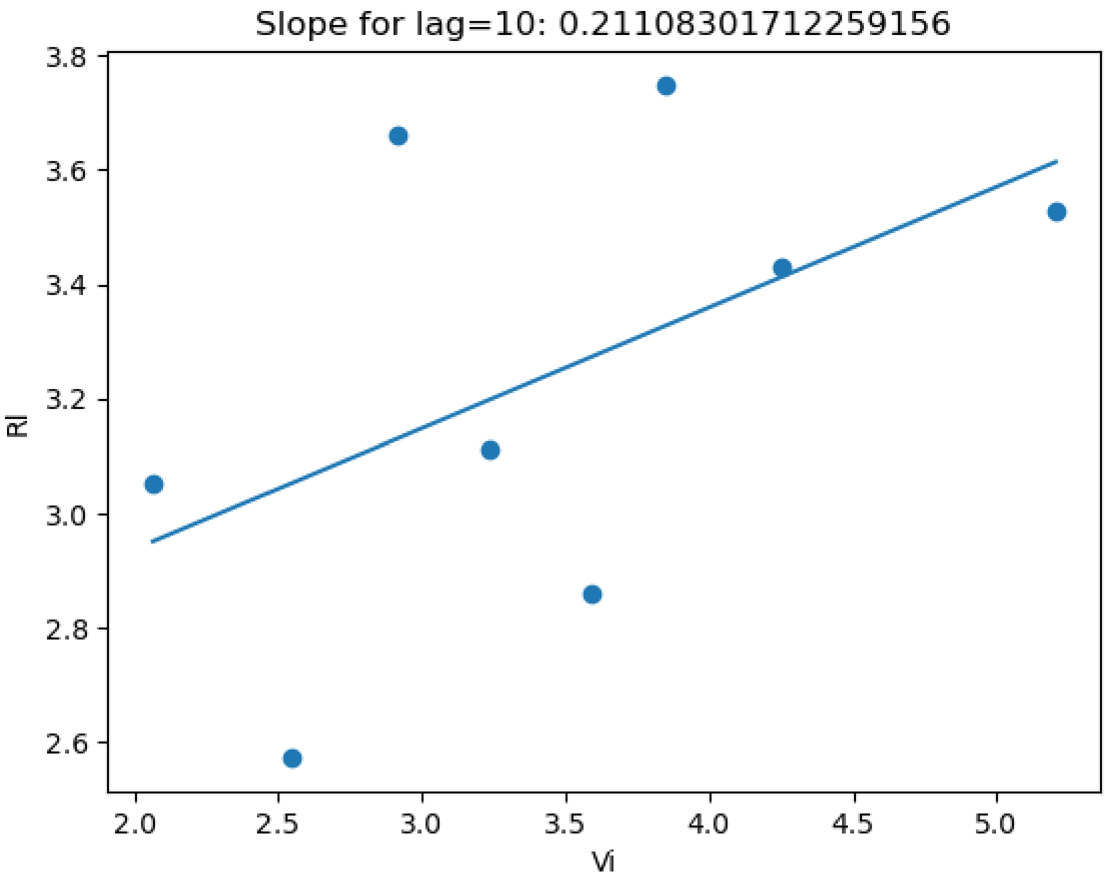


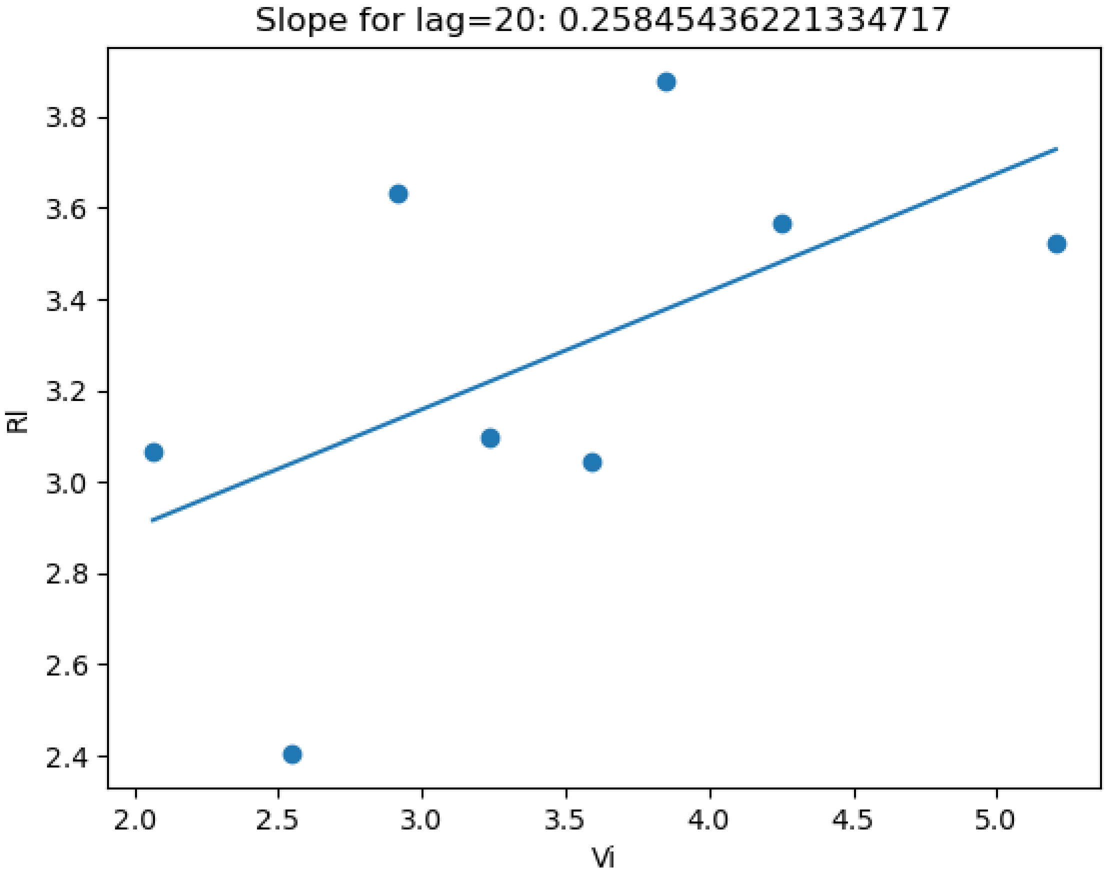
Question 4

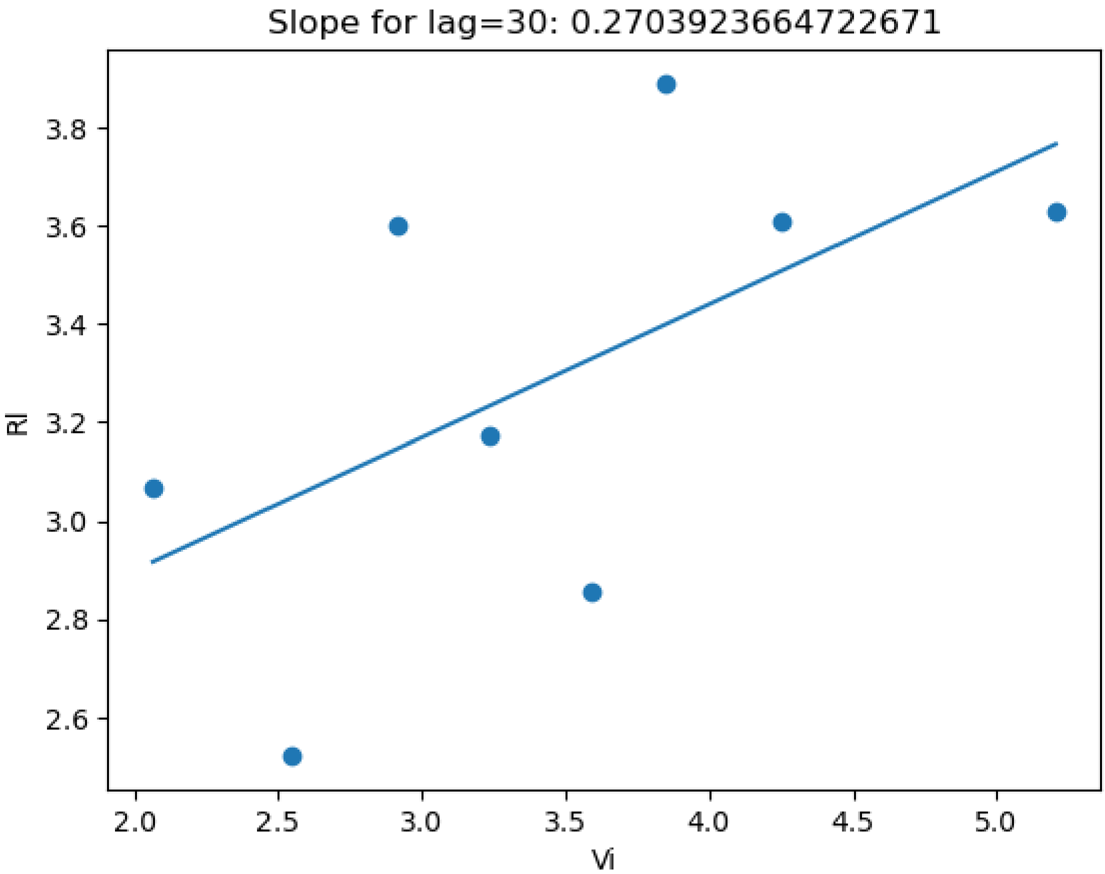
In [3]:

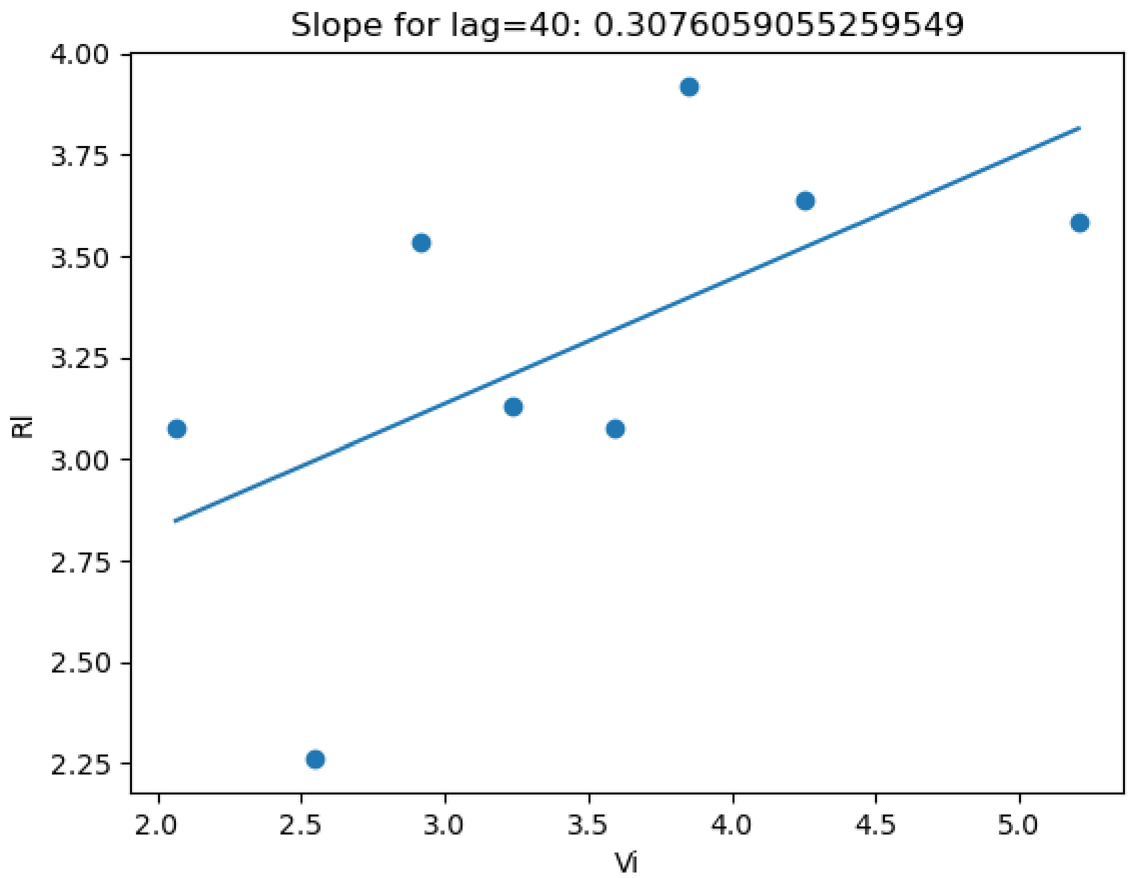
```
size_bins=[0,2,5,10,15,20,30,40,55,90,100000]
log_response_functions=[]
log_average_trade_sizes=[]
for lag in [10,20,30,40,50,75,100,125,150,175,200,250]:
    response_functions=[]
    avg_trade_sizes=[]
```

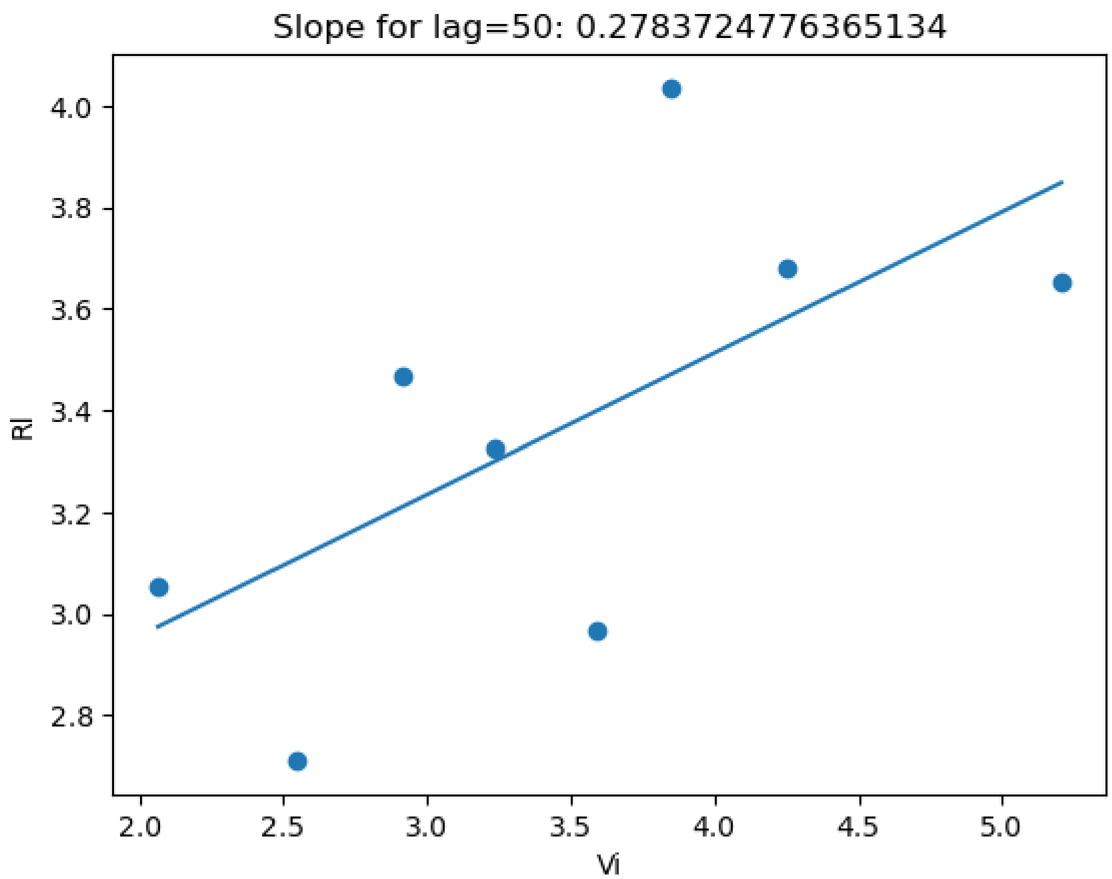
```
    for i in range(len(size_bins)-1):
        size_bin_data=dataset[(dataset['Size']>size_bins[i])&(dataset['Size
        size_bin_data['Response']=(size_bin_data.groupby('Date')['WeightedAv
        response_functions.append(np.log(size_bin_data.groupby('Date')['Resp
        avg_trade_sizes.append(np.log(size_bin_data['Size'].mean()))
    log_response_functions.append(response_functions)
    log_average_trade_sizes.append(avg_trade_sizes)
for i,lag in enumerate([10,20,30,40,50,75,100,125,150,175,200,250]):
    X=np.array(log_average_trade_sizes[i]).reshape(-1,1)
    y=np.array(log_response_functions[i])
    valid_indices=~np.isnan(X.flatten())&~np.isnan(y)
    X=X[valid_indices].reshape(-1,1)
    y=y[valid_indices]
    model = LinearRegression().fit(X,y)
    plt.scatter(X.flatten(),y)
    plt.plot(X.flatten(),model.predict(X))
    plt.xlabel('Vi')
    plt.ylabel('Rl')
    plt.title(f'Slope for lag={lag}: {model.coef_[0]}')
    plt.show()
```

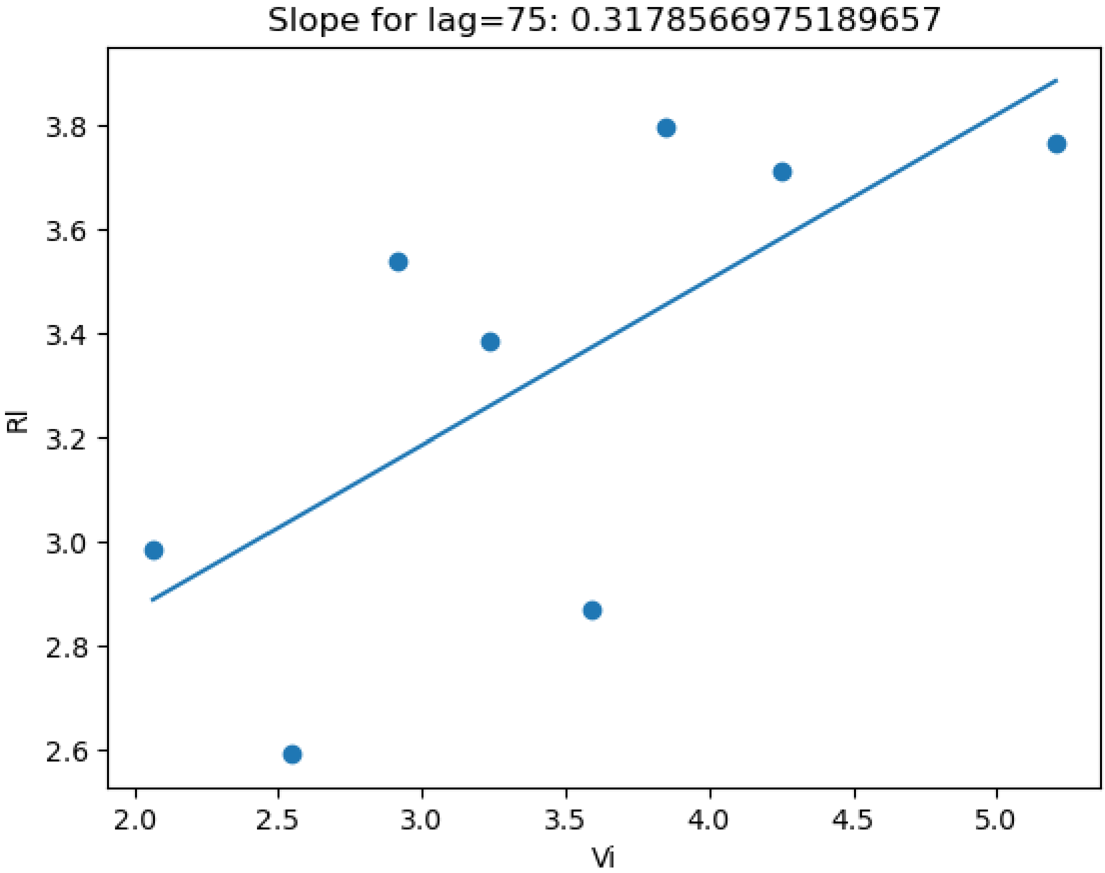


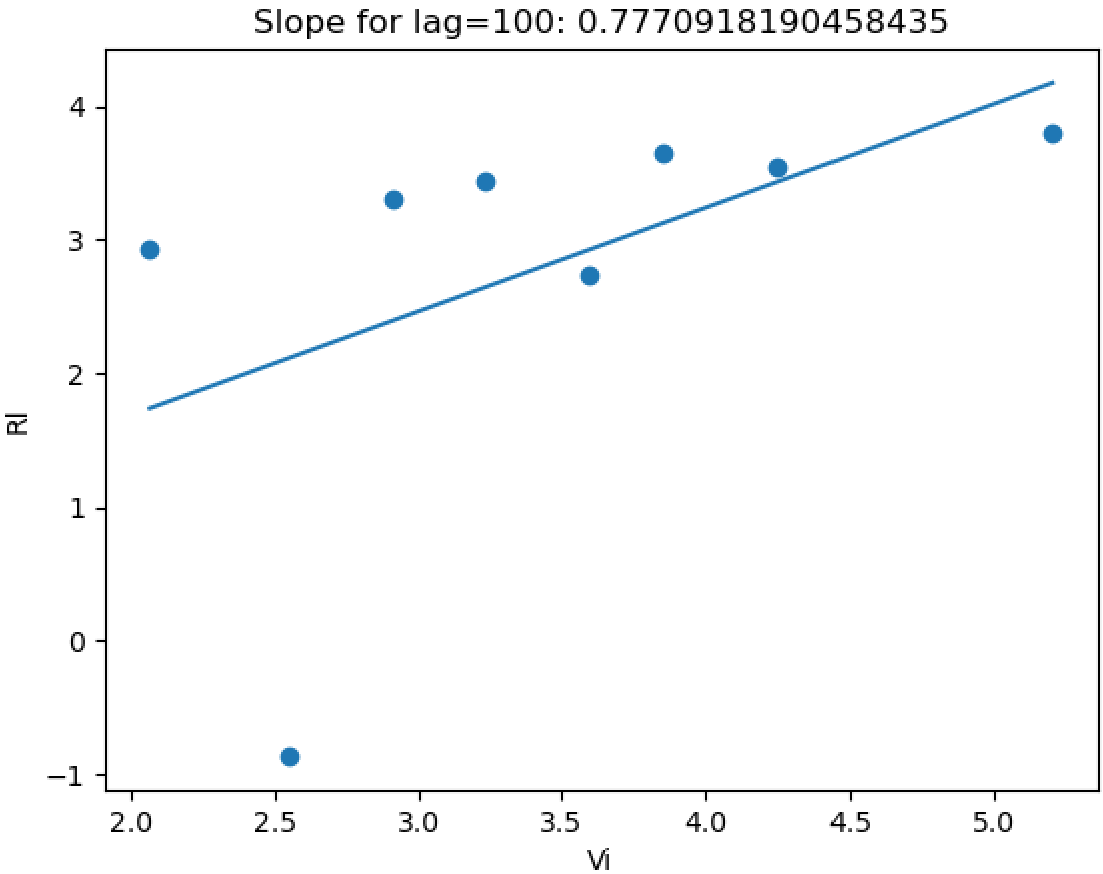

Slope for lag=20: 0.25845436221334717
RI
Vi
3.8
3.6
3.4
3.2
3.0
2.8
2.6
2.4
2.0
2.5
3.0
3.5
4.0
4.5
5.0

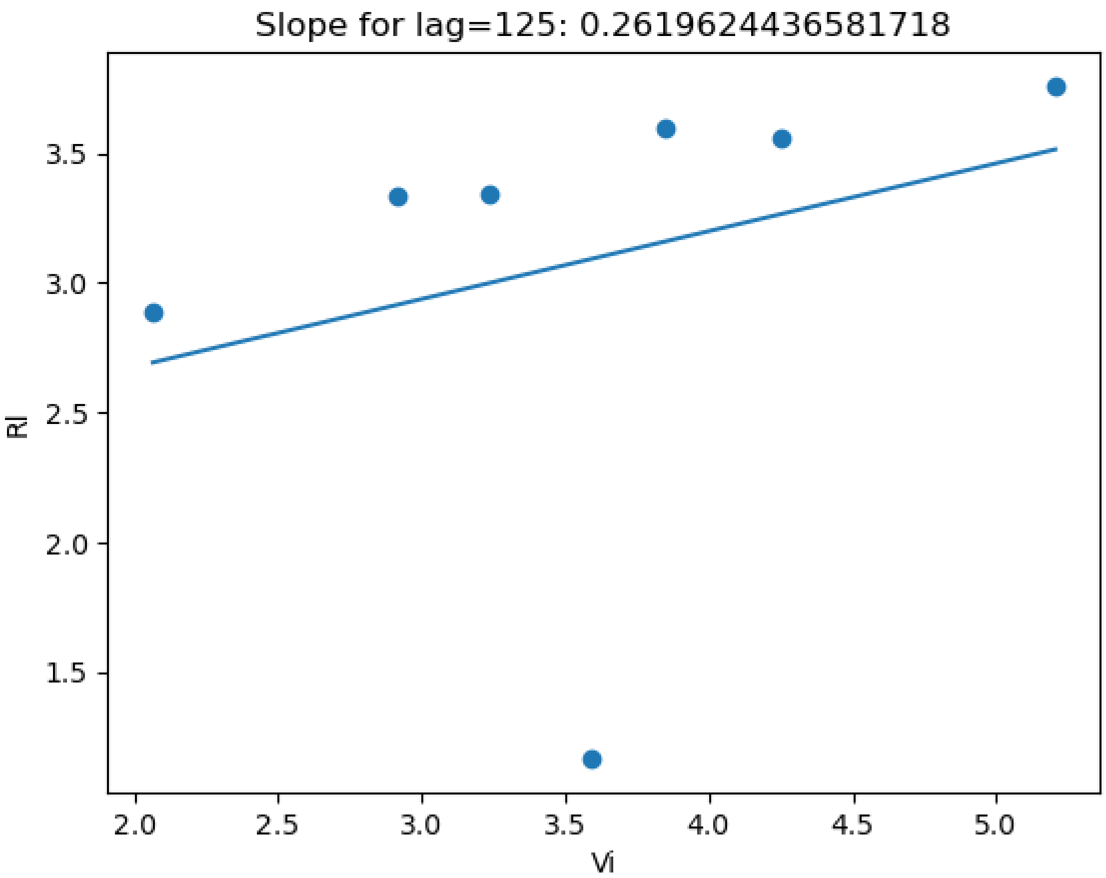

Slope for lag=30: 0.2703923664722671
RI
Vi
3.8
3.6
3.4
3.2
3.0
2.8
2.6
2.0
2.5
3.0
3.5
4.0
4.5
5.0

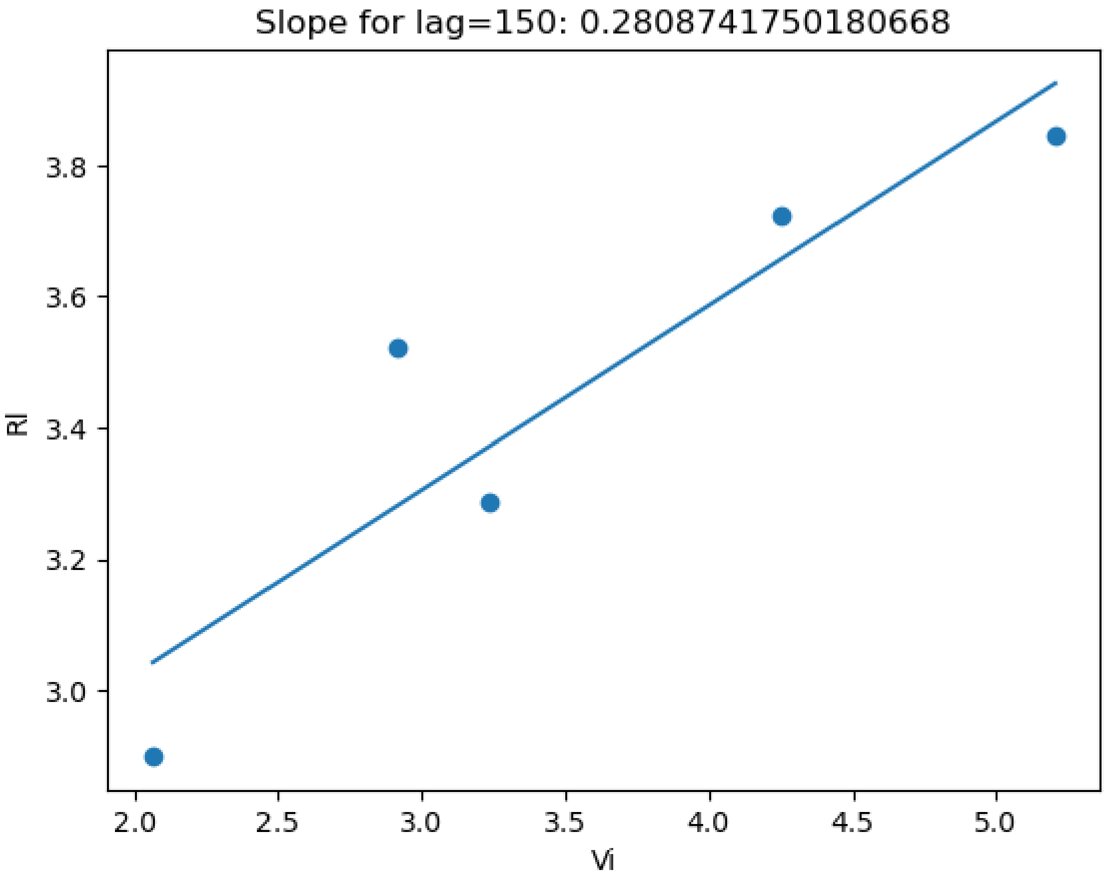

Slope for lag=40: 0.3076059055259549
4.00
3.75
3.50
3.25
3.00
2.75
2.50
2.25
RI
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Vi

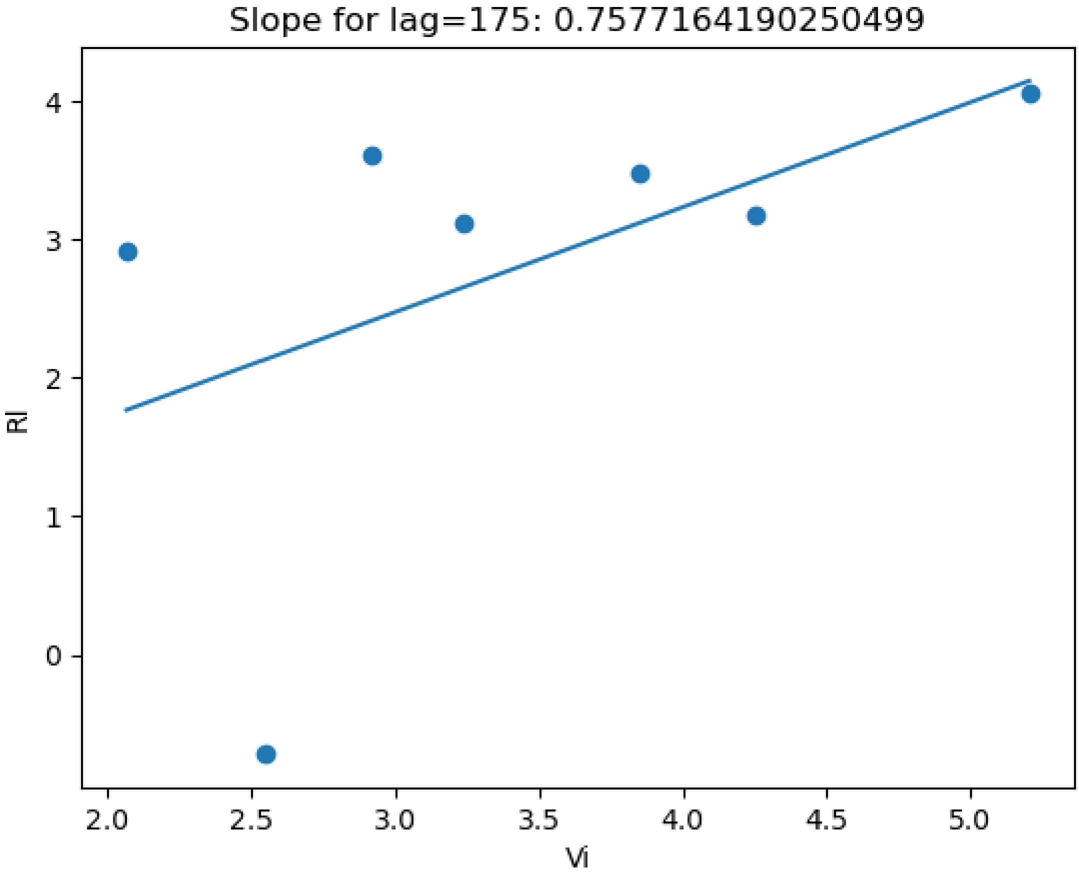

Slope for lag=50: 0.2783724776365134
4.0
3.8
3.6
3.4
3.2
3.0
2.8
RI
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Vi

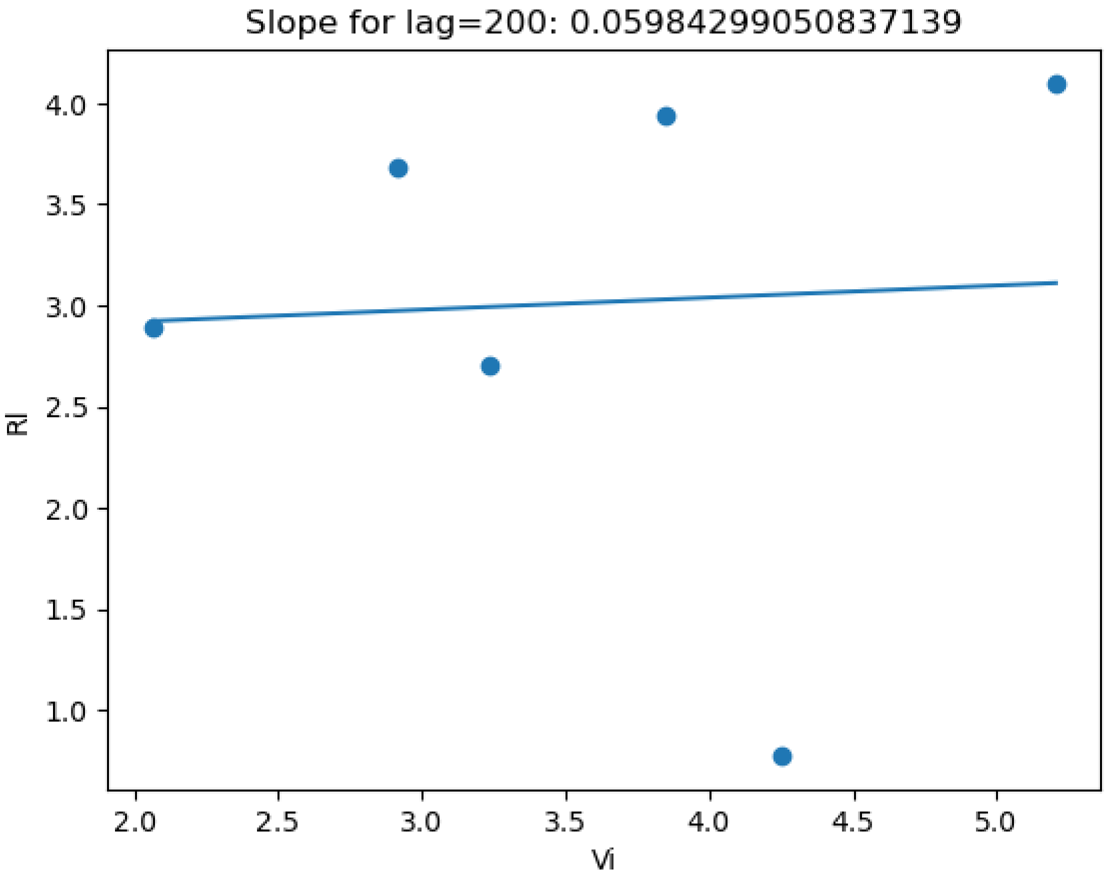

Slope for lag=75: 0.3178566975189657
3.8
3.6
3.4
3.2
3.0
2.8
2.6
RI
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Vi

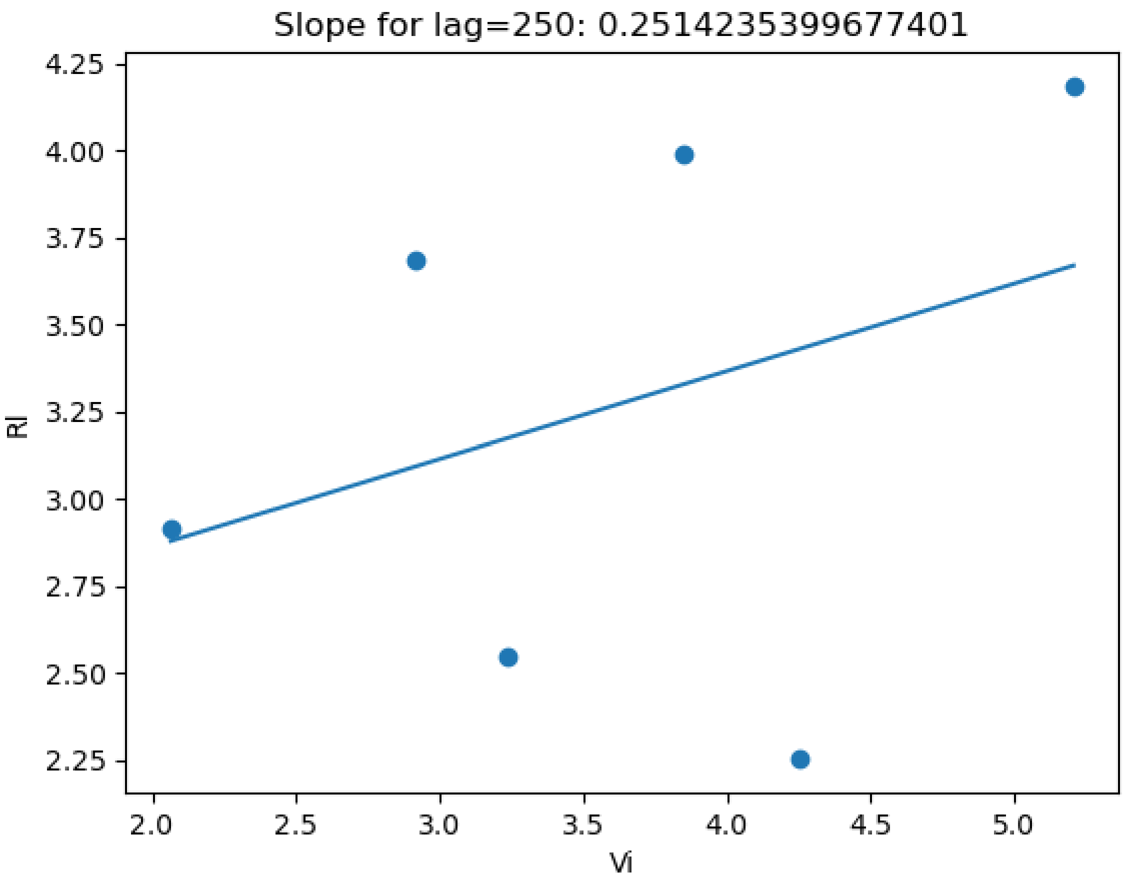

Slope for lag=100: 0.7770918190458435
4
3
2
1
0
−1
RI
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Vi


Slope for lag=125: 0.2619624436581718
RI
Vi
3.5
3.0
2.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0


Slope for lag=150: 0.2808741750180668
RI
Vi
3.8
3.6
3.4
3.2
3.0
2.0
2.5
3.0
3.5
4.0
4.5
5.0


Slope for lag=175: 0.7577164190250499
RI
Vi
0
1
2
3
4
2.0
2.5
3.0
3.5
4.0
4.5
5.0

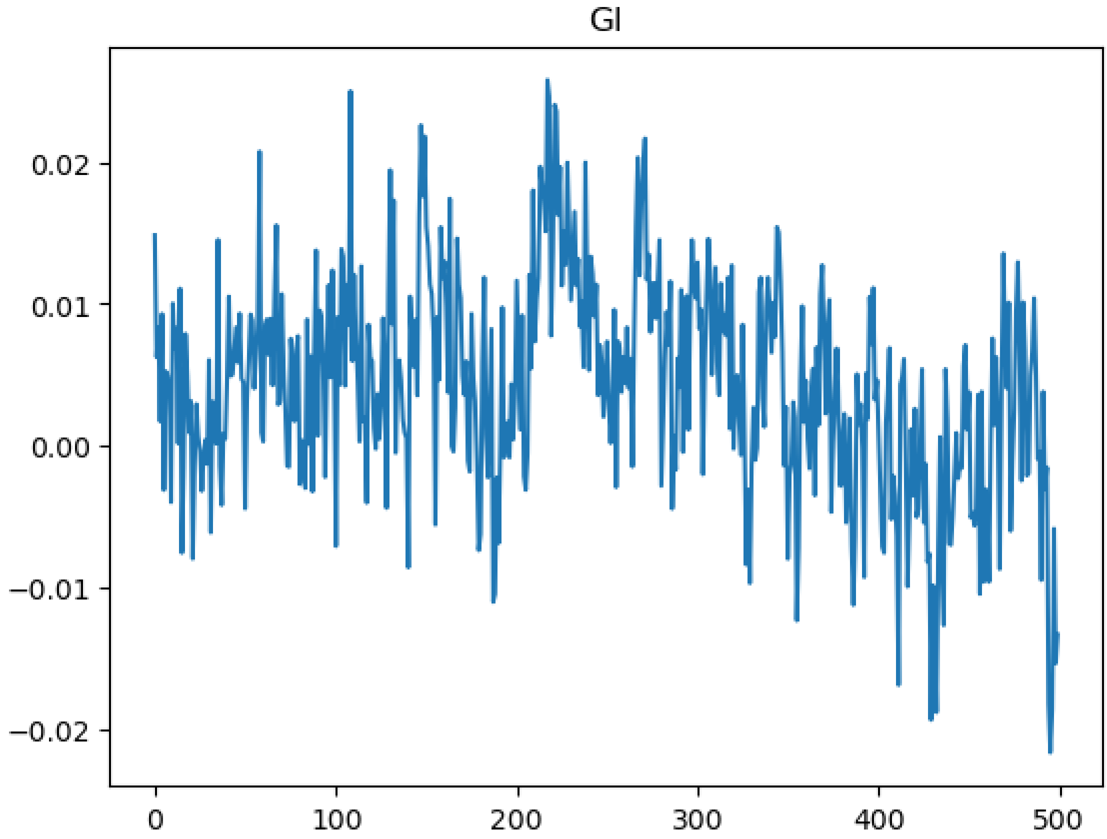

Slope for lag=200: 0.05984299050837139
RI
Vi
1.0
1.5
2.0
2.5
3.0
3.5
4.0
2.0
2.5
3.0
3.5
4.0
4.5
5.0



Question 5

In [5]:

```
log_response_functions=np.log(np.abs(response_function[1:]))
corr_values=np.array([dataset['Sign'].autocorr(lag) for lag in np.arange(1,5
valid_indices=~np.isnan(corr_values)&~np.isnan(log_response_functions)
corr_values=np.array(corr_values)[valid_indices]
log_response_functions=np.array(log_response_functions)[valid_indices]
A_matrix=np.zeros((len(np.arange(1,501)),len(np.arange(1,501))))
for i,lag in enumerate(np.arange(1, 501)):
    A_matrix[i,:lag]=corr_values[-lag:]
    A_matrix[i,lag:]=-corr_values[:500-lag]
model=LinearRegression().fit(A_matrix,np.array(log_response_functions))
plt.plot(model.coef_)
plt.title("Gl")
plt.show()
```


GI
0.02
0.01
0.00
−0.01
−0.02
0
100
200
300
400
500