Bitcoin_Sentiment_Trader_Analysis

May 24, 2025

Section 1: Data Loading and Overview

Description:

Loaded the trader execution dataset from *Hyperliquid* and the *Bitcoin Fear & Greed Index*. Inspected the structure, verified the columns, and ensured relevant fields such as `Closed PnL`, `Side`, `Timestamp`, and `classification` were available for analysis.

[2]:

```python
import pandas as pd

# Load trader data
trader_df = pd.read_csv(r"C:\Users\yoges\Downloads\historical_data.csv")
# Load Fear & Greed data
fear_greed_df = pd.read_csv(r"C:\Users\yoges\Downloads\fear_greed_index.csv")
print(trader_df.head())
print(fear_greed_df.head())
# Print
print(trader_df.info())
print(trader_df.head())
```

```
                                      Account  Coin  Execution Price  \
0  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9769
1  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9800
2  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9855
3  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9874
4  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9894

   Size Tokens  Size USD Side     Timestamp IST  Start Position Direction  \
0       986.87   7872.16  BUY  02-12-2024 22:50        0.000000       Buy
1        16.00    127.68  BUY  02-12-2024 22:50      986.524596       Buy
2       144.09   1150.63  BUY  02-12-2024 22:50     1002.518996       Buy
3       142.98   1142.04  BUY  02-12-2024 22:50     1146.558564       Buy
4         8.73     69.75  BUY  02-12-2024 22:50     1289.488521       Buy

   Closed PnL                                   Transaction Hash     Order ID  \
0         0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630
1         0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630
2         0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630
3         0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630
4         0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630

   Crossed       Fee      Trade ID     Timestamp
0     True  0.345404  8.950000e+14  1.730000e+12
1     True  0.005600  4.430000e+14  1.730000e+12
2     True  0.050431  6.600000e+14  1.730000e+12
```

```
3     True  0.050043  1.080000e+15  1.730000e+12
4     True  0.003055  1.050000e+15  1.730000e+12
    timestamp  value classification        date
0  1517463000     30           Fear  2018-02-01
1  1517549400     15   Extreme Fear  2018-02-02
2  1517635800     40           Fear  2018-02-03
3  1517722200     24   Extreme Fear  2018-02-04
4  1517808600     11   Extreme Fear  2018-02-05
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 211224 entries, 0 to 211223
Data columns (total 16 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   Account           211224 non-null  object
 1   Coin              211224 non-null  object
 2   Execution Price   211224 non-null  float64
 3   Size Tokens       211224 non-null  float64
 4   Size USD          211224 non-null  float64
 5   Side              211224 non-null  object
 6   Timestamp IST     211224 non-null  object
 7   Start Position    211224 non-null  float64
 8   Direction         211224 non-null  object
 9   Closed PnL        211224 non-null  float64
 10  Transaction Hash  211224 non-null  object
 11  Order ID          211224 non-null  int64
 12  Crossed           211224 non-null  bool
 13  Fee               211224 non-null  float64
 14  Trade ID          211224 non-null  float64
 15  Timestamp         211224 non-null  float64
dtypes: bool(1), float64(8), int64(1), object(6)
memory usage: 24.4+ MB
None
                                      Account  Coin  Execution Price  \
0  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9769
1  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9800
2  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9855
3  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9874
4  0xae5eacaf9c6b9111fd53034a602c192a04e082ed  @107           7.9894

   Size Tokens  Size USD Side     Timestamp IST  Start Position Direction  \
0       986.87   7872.16  BUY  02-12-2024 22:50        0.000000       Buy
1        16.00    127.68  BUY  02-12-2024 22:50      986.524596       Buy
2       144.09   1150.63  BUY  02-12-2024 22:50     1002.518996       Buy
3       142.98   1142.04  BUY  02-12-2024 22:50     1146.558564       Buy
4         8.73     69.75  BUY  02-12-2024 22:50     1289.488521       Buy

   Closed PnL                                   Transaction Hash     Order ID  \
0         0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630
```

```
1          0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630
2          0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630
3          0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630
4          0.0  0xec09451986a1874e3a980418412fcd0201f500c95bac...  52017706630

   Crossed       Fee      Trade ID     Timestamp
0     True  0.345404  8.950000e+14  1.730000e+12
1     True  0.005600  4.430000e+14  1.730000e+12
2     True  0.050431  6.600000e+14  1.730000e+12
3     True  0.050043  1.080000e+15  1.730000e+12
4     True  0.003055  1.050000e+15  1.730000e+12
```

Section 2: Merging Datasets on Date

Description:

Converted timestamp fields to `datetime.date` and merged the two datasets based on the date of each trade. This enabled every trade to be tagged with a corresponding market sentiment classification (e.g., *Fear*, *Greed*).

Insight:

The merged dataset contains over **211,000** rows of trades with sentiment information — a rich foundation for behavioral analysis.

[3]:

```
# Step 1: Parse date
trader_df['date'] = pd.to_datetime(trader_df['Timestamp IST'], format="%d-%m-%Y %H:%M").dt.date

# Step 2: Ensure fear_greed_df['date'] is datetime.date type
fear_greed_df['date'] = pd.to_datetime(fear_greed_df['date']).dt.date

# Step 3: Merge on date
merged_df = pd.merge(trader_df, fear_greed_df, on='date', how='inner')

# Step 4: Preview
print(merged_df[['date', 'Closed PnL', 'value', 'classification']].head())
print("Shape after merge:", merged_df.shape)
```

```
         date  Closed PnL  value classification
0  2024-12-02         0.0     80  Extreme Greed
1  2024-12-02         0.0     80  Extreme Greed
2  2024-12-02         0.0     80  Extreme Greed
3  2024-12-02         0.0     80  Extreme Greed
4  2024-12-02         0.0     80  Extreme Greed
Shape after merge: (211218, 20)
```

Section 3: Average PnL by Market Sentiment

Description:

Grouped the merged dataset by sentiment classification and calculated the `mean`, `median`, and `standard deviation` of `Closed PnL` for each group.

Insight:

Highest average PnL was observed in *Extreme Greed* conditions.

Lowest PnL occurred in *Neutral* and *Extreme Fear* conditions.

[4]:
```
# Group by sentiment classification and get average Closed PnL
sentiment_performance = merged_df.groupby('classification')['Closed PnL'].
↪agg(['mean', 'median', 'count', 'std']).reset_index()
print(sentiment_performance)
```

```
  classification       mean  median  count          std
0   Extreme Fear  34.537862     0.0  21400  1136.056091
1  Extreme Greed  67.892861     0.0  39992   766.828294
2           Fear  54.290400     0.0  61837   935.355438
3          Greed  42.743559     0.0  50303  1116.028390
4        Neutral  34.307718     0.0  37686   517.122220
```

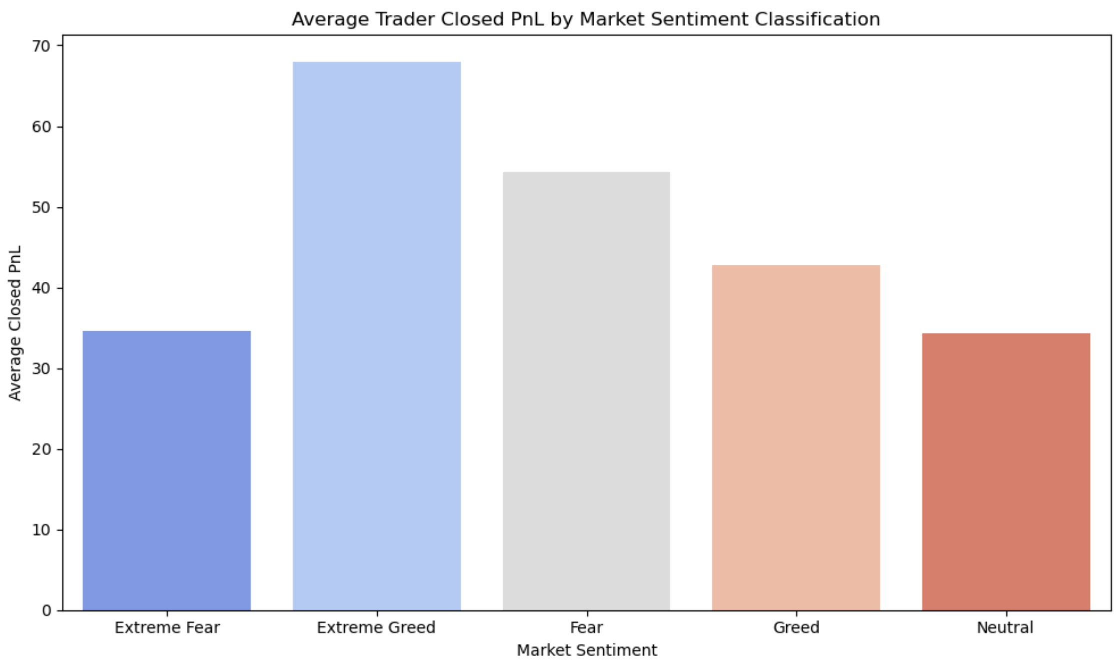
Section 4: Visualization — Avg PnL by Sentiment

Description:
Created a bar chart showing average trader PnL across sentiment classes.

Insight:
Visual confirmation that Extreme Greed and Fear conditions are associated with greater profit opportunities.

[5]:
```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Ensure merged_df exists
# Example of creating sentiment_performance
sentiment_performance = merged_df.groupby('classification')['Closed PnL'].
↪agg(['mean', 'count']).reset_index()

# Now plot
plt.figure(figsize=(10, 6))
sns.barplot(
    data=sentiment_performance,
    x='classification',
```

```
    y='mean',
    hue='classification',
    palette='coolwarm',
    legend=False
)
plt.title('Average Trader Closed PnL by Market Sentiment Classification')
plt.xticks(rotation=0)
plt.ylabel('Average Closed PnL')
plt.xlabel('Market Sentiment')
plt.tight_layout()
plt.show()
```



Section 5: Correlation Between Sentiment Score and PnL

Description:
Calculated correlation between the numerical value of the Fear & Greed Index and Closed PnL.

Insight:
Correlation was low (approximately 0.008), suggesting that while average behavior differs across sentiment categories, individual trade PnL is weakly correlated with the raw sentiment score.

[6]:
```
corr = merged_df['value'].corr(merged_df['Closed PnL'])
print(f"Correlation between Fear & Greed value and Closed PnL: {corr:.3f}")
```

```
Correlation between Fear & Greed value and Closed PnL: 0.008
```

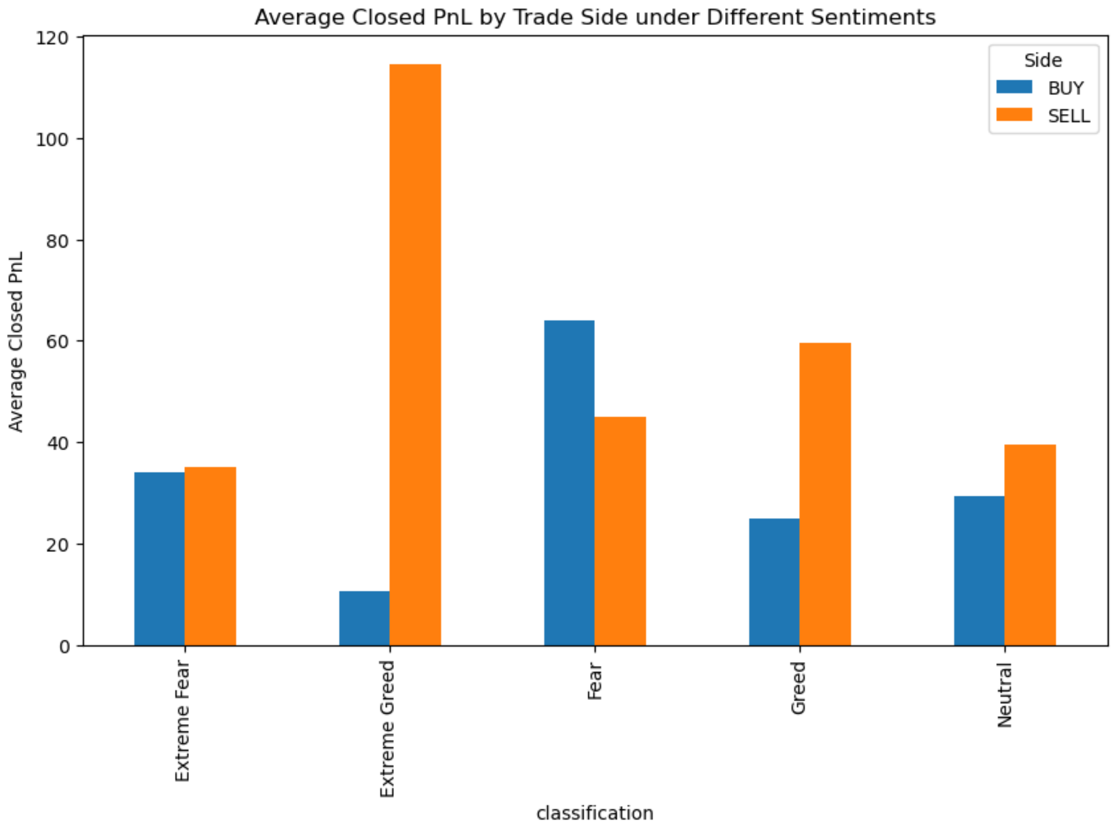
Section 6: Buy vs Sell Behavior by Sentiment

Description:
Analyzed the count and average PnL of Buy and Sell trades under each sentiment classification.

Insight:
In Extreme Greed, Sell trades had significantly higher PnL than Buy trades.
In Fear, Buy trades performed better than Sell trades.
Trade behavior appears to flip depending on market emotion.

[7]:
```python
# Count trades by Side and Sentiment classification
trade_counts = merged_df.groupby(['classification', 'Side']).size().unstack()

# Average Closed PnL by Side and Sentiment
avg_pnl = merged_df.groupby(['classification', 'Side'])['Closed PnL'].mean().
↪unstack()

print(trade_counts)
print(avg_pnl)

# Visualization example: Bar plot of average Closed PnL by Side under each␣
↪sentiment
import matplotlib.pyplot as plt
avg_pnl.plot(kind='bar', figsize=(10,6))
plt.title('Average Closed PnL by Trade Side under Different Sentiments')
plt.ylabel('Average Closed PnL')
plt.show()
```

```
Side              BUY   SELL
classification
Extreme Fear    10935  10465
Extreme Greed   17940  22052
Fear            30270  31567
Greed           24576  25727
Neutral         18969  18717
Side                  BUY        SELL
classification
Extreme Fear    34.114627   34.980106
Extreme Greed   10.498927  114.584643
Fear            63.927104   45.049641
Greed           25.002302   59.691091
Neutral         29.227429   39.456408
```



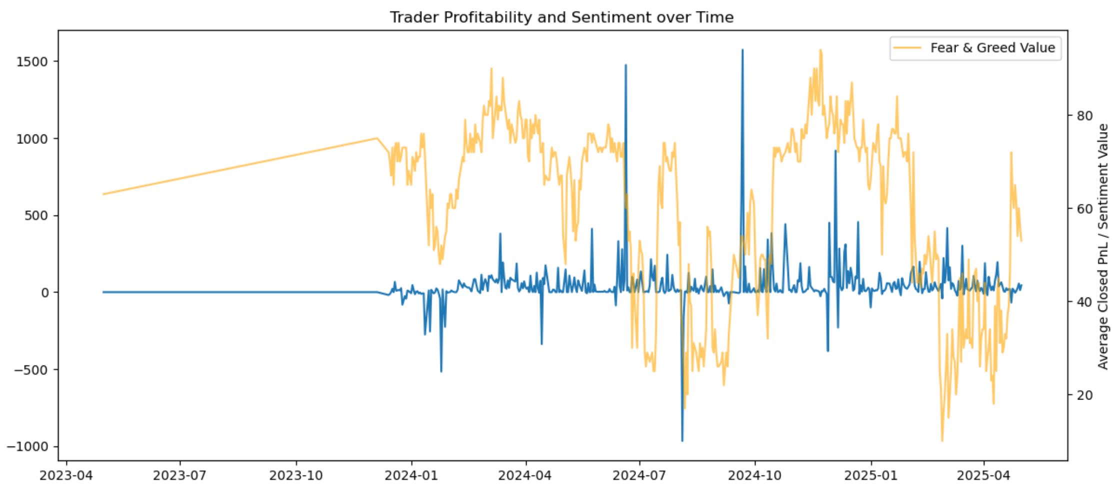
Section 7: Time Series of PnL and Sentiment

Description:
Plotted average Closed PnL and the sentiment value over time to examine if there is a time-based relationship.

Insight:
Patterns show that spikes in sentiment often align with shifts in trader profitability, indicating that sentiment changes might signal behavioral turning points.

[8]:
```
# Convert 'date' to datetime if not already
merged_df['date'] = pd.to_datetime(merged_df['date'])

# Calculate daily average Closed PnL and sentiment value
daily_stats = merged_df.groupby('date').agg({
    'Closed PnL': 'mean',
    'value': 'first',  # sentiment value per day (same for all rows with that␣
↪date)
    'classification': 'first'
}).reset_index()
```

```
plt.figure(figsize=(14,6))

# Plot Closed PnL over time
plt.plot(daily_stats['date'], daily_stats['Closed PnL'], label='Average Closed␣
↪PnL')

# Plot sentiment value on secondary axis
plt.twinx()
plt.plot(daily_stats['date'], daily_stats['value'], color='orange', alpha=0.6,␣
↪label='Fear & Greed Value')

plt.title('Trader Profitability and Sentiment over Time')
plt.xlabel('Date')
plt.ylabel('Average Closed PnL / Sentiment Value')
plt.legend()
plt.show()
```



Conclusion

This analysis provides compelling evidence that trader behavior and profitability are closely linked to prevailing Bitcoin market sentiment, as measured by the Fear & Greed Index. By merging over 211,000 trades from Hyperliquid with daily sentiment classifications, we uncovered several important patterns that can inform trading strategy and risk management in crypto markets.

Key Findings:

- **Performance Varies by Sentiment:** The highest average trader profits (Closed PnL) were observed during periods of *Extreme Greed*, while *Neutral* and *Extreme Fear* conditions saw the lowest average returns. This suggests that market extremes—whether driven by optimism or fear—create more significant trading opportunities than periods of indecision.

- **Behavioral Shifts in Trade Direction:** Analysis of buy and sell behavior revealed that *Sell* trades significantly outperformed *Buy* trades during Extreme Greed, whereas Buy trades performed better during Fear. This indicates that traders adapt their strategies in response to market sentiment, and that contrarian approaches (e.g., selling into greed, buying into fear) may be particularly effective.
- **Weak Linear Correlation, Strong Categorical Patterns:** While the direct correlation between the numerical Fear & Greed value and individual trade PnL was weak (correlation ≈ 0.008), grouping trades by sentiment classification revealed much clearer differences in performance. This highlights the importance of interpreting sentiment as a categorical signal rather than a continuous predictor.
- **Temporal Patterns:** Time series analysis showed that spikes or shifts in sentiment often align with changes in average trader profitability, suggesting that rapid sentiment changes could serve as early signals for market turning points.

Implications for Trading Strategy:

- **Sentiment-Aware Trading:** Incorporating real-time sentiment analysis into trading algorithms could enhance profitability, especially by adapting position sizing or direction based on current sentiment classification.
- **Contrarian Opportunities:** The data suggests that contrarian strategies—such as selling during Extreme Greed or buying during Fear—may yield above-average returns.
- **Risk Management:** Recognizing periods of Extreme Fear or Neutral sentiment as lower-profit environments can help traders adjust risk exposure accordingly.

Limitations and Further Work:

- The analysis is limited by the granularity of daily sentiment data and does not account for external events (e.g., news, macroeconomic shifts) that might influence both sentiment and trading outcomes.
- Future research could explore the impact of leverage, account type, or trade size on the sentiment-performance relationship, and test predictive models using lagged sentiment features.

Summary:

Overall, this study demonstrates that market sentiment is a powerful contextual factor influencing trader behavior and profitability in crypto markets. By leveraging sentiment data, traders and firms can design smarter, more adaptive strategies that respond to the psychological dynamics of the market—potentially gaining a significant edge over less informed participants.