GROUP WORK PROJECT # 1 M6 MScFE 600: FINANCIAL DATA
GROUP NUMBER: 12698

FULL LEGAL NAME	LOCATION (COUNTRY)	EMAIL ADDRESS	MARK X FOR ANY NON-CONTRIBUTING MEMBER
Gracious Tehillah	Kenya	Gracious Tehillah Kenya gracious.ogita@strathmore.edu	
Sarat Das	India	saratdas67012@gmail.com	
X	X	X	X

Statement of integrity: By typing the names of all group members in the text boxes below, you confirm that the assignment submitted is original work produced by the group (excluding any non-contributing members identified with an "X" above).

Team member 1	**Gracious Tehillah**
Team member 2	**Sarat Das**
Team member 3	**X**

Use the box below to explain any attempts to reach out to a non-contributing member. Type (N/A) if all members contributed.
Note: You may be required to provide proof of your outreach to non-contributing members upon request.

Gemechu Bosho Deressa has not posted any messages or participated in the discussion. We attempted to reach out through the discussion platform, but there was no response. There are no other available contact options, and the member appears to be completely out of communication.

Part 1. Assessing Models (Intelligent Stock Prediction)

Q1: Data Understanding:-

The authors used basic financial data like daily opening, high, low, and closing prices, along with trading volume, to build their stock prediction model. They gathered this data from Yahoo Finance for three ETFs. This raw data was then processed into over 200 technical indicators using a library called PandasTA. These indicators are designed to spot patterns in price and volume that can help predict future movements, like whether a stock might go up or down.

These indicators are essential because they compress a lot of information about past stock behavior into simpler patterns that are easier for a machine learning model to understand. For example, a moving average shows the general trend, while indicators like RSI can signal if a stock is overbought or oversold. By selecting only the most useful indicators, the authors improved the accuracy of the predictions and made the model run faster .

Q2: Security Understanding:-

The ETF selected is ECH (iShares MSCI Chile Capped ETF). This fund is traded on the **NYSE** and includes major **Chilean stocks**. It started in 2007 and had price fluctuations from around $29 to $80 between 2009 and 2020. During this time, the average price was about $50. More recently, the price has been around $45, with a market value of about $1.3 billion. It also pays a modest dividend.

Instead of predicting exact stock prices, the authors chose **to predict the direction of movement**—whether the stock goes up or down. This is a simpler and often more useful goal in finance. They used a binary target: 1 if today's open is higher than yesterday's, and 0 otherwise.

This makes it a classification problem. Other ways to set this up might include using the closing price instead of the open, or adding a third category like "no change" if the price moves less than 1%.

Q3: Methodology Understanding :-

The subcategories of Data (Section 2):-

- ETFs and Timeframe
- Raw Data and Sectors
- Feature Construction
- Class Label Definition

- Data Preprocessing

The subcategories of Methodology (Section 3):-

- Neural Network Model
- LASSO Feature Selection
- Dimensionality Reduction
- Specialized Metrics like Dispersion Ratio
- Validation (10-fold cross-validation)
- Experiment Algorithm (how all pieces are combined)

Descriptive methods like Pearson correlation help understand feature relationships. While, LASSO is a modeling technique that selects features by penalizing unimportant ones. The authors combine several strategies to narrow down the 200+ indicators to the most relevant ~10. This not only improves performance but saves computation time.

Q4: Feature Understanding :-

Features in the paper are the technical indicators and raw data points (Open, Close, etc.) used as inputs to the model. A method is something like LASSO or Pearson correlation that processes features. A model (like a neural net) uses features to make predictions. Features are grouped into types—momentum, volume, trend, etc.—and optimized by ranking them using different selection methods. Then, only the best ones are kept. This reduced set performed better than using all 216 features.

Q5: Optimization Understanding:-

Cross-validation means dividing data into chunks (folds), training on some parts and testing on others, rotating through all chunks. This helps ensure that the model doesn't just memorize one set. In 10-fold CV, the data is split into 10 equal parts.

Jaccard distance is used to compare sets of selected features. If two ETFs choose very similar features, their Jaccard distance is small. For example, the Chile and Brazil ETFs shared many features (low distance), but the U.S. ETF differed more. Other distances like Hamming or Cosine could be used too, but Jaccard works well for comparing sets.

The "best" set of features is the one that gives the highest prediction accuracy with the fewest indicators. The authors tested many combinations and found that using about 10 indicators gave the best result.

Step 1: Financial Problem:-

The authors wanted to solve the problem of predicting whether a stock (ETF) will go up or down tomorrow. This is especially tough in emerging markets like Chile and Brazil because they are more volatile and sensitive to local events. Their model is tuned for such markets, unlike developed markets like the U.S. where conditions are more stable. By customizing the approach to emerging markets, the model can better spot opportunities and manage risks.

Step 2: Application:-

The main result is that using only about 5% of all indicators (roughly 10 out of 200+) gave better predictions than using all of them. For instance, with the Chile ETF, this approach raised accuracy from 77% to 79%. It also sped up training by 85%. Indicators that performed best included volume and momentum-based ones like Balance of Power, Bollinger Bands, and Williams %R. Interestingly, developed-market ETFs needed different features, price-volume interactions.

Step 3: Replication:-

We selected the ECH ETF, focusing on data from 2009–2020. Using Yahoo Finance, we downloaded daily price and volume data.

```
def download_ech_data(start_date, end_date, filename):
    """
    Download historical daily data for ECH from Yahoo Finance and save to CSV.
    - start_date, end_date: strings "YYYY-MM-DD".
    - filename: output CSV file path.
    """
    try:
        # Yahoo Finance end date is exclusive: add 1 day to include final date
        end_inclusive = (datetime.strptime(end_date, '%Y-%m-%d') + timedelta(days=1)).strftime
        # Fetch data (Open, High, Low, Close, Volume, Adj Close):contentReference[oaicite:12]{
        df = yf.download("ECH", start=start_date, end=end_inclusive, progress=False)

        # Check if data was returned
        if df.empty:
            raise ValueError("No data found for ECH between given dates.")

        # Save DataFrame to CSV
        df.to_csv(filename)
        print(f"Successfully saved {len(df)} rows of data to {filename}")

    except Exception as e:
        print(f"Error fetching data: {e}")

#  download data for ECH from Dec 12, 2009 to Jan 1, 2020
download_ech_data("2009-12-12", "2020-01-01", "ECH_2009-2020.csv")
```

```
Successfully saved 2529 rows of data to ECH_2009-2020.csv
```

Data File Link : [Data]

Analysis :-

[7]:
```
# Load your uploaded CSV
df = pd.read_csv("ECH_2009-2020.csv", index_col=0, parse_dates=True)
```

```
C:\Users\Asus\AppData\Local\Temp\ipykernel_20840\1610733760.py:2: UserWarning: Could not infer format, so each element will be parsed individually, fal
ling back to `dateutil`. To ensure parsing is consistent and as-expected, please specify a format.
  df = pd.read_csv("ECH_2009-2020.csv", index_col=0, parse_dates=True)
```

[11]:
```
df.head(5)
```

[11]:

Price	Close	High	Low	Open	Volume
Ticker	ECH	ECH	ECH	ECH	ECH
Date	NaN	NaN	NaN	NaN	NaN
2009-12-14	36.48460388183594	36.63846237194658	36.05647546555473	36.19026368175762	212300
2009-12-15	36.35749816894531	36.51135919678961	36.29060406697977	36.29060406697977	122100
2009-12-16	36.39762496948242	36.79230617415169	36.324039948838816	36.79230617415169	201500

[15]:
```
# Convert Open column to numeric (coerce errors to NaN)
df['Open'] = pd.to_numeric(df['Open'], errors='coerce')
```

[16]:
```

# Drop any rows where Open is missing or non-numeric
df = df.dropna(subset=['Open'])
```

[15]:
```
# Convert Open column to numeric (coerce errors to NaN)
df['Open'] = pd.to_numeric(df['Open'], errors='coerce')
```

[16]:
```

# Drop any rows where Open is missing or non-numeric
df = df.dropna(subset=['Open'])
```

[18]:
```


# Now compute Gamma safely
df['Gamma'] = (df['Open'].diff() > 0).astype(int)
```

[19]:
```
# Drop the first row (has NaN from diff)
df = df.dropna()
```

[21]:
```
# Rolling arithmetic and geometric means over 10 days
df['Open_roll_mean'] = df['Open'].rolling(window=10).mean()
df['Open_roll_geom'] = df['Open'].rolling(window=10).apply(lambda x: np.exp(np.mean(np.log(x))), raw=False)
```

We calculated a basic indicator called the Dispersion Ratio:
DR_{10} = (10-day average) / (10-day geometric mean)

```
# Dispersion Ratio
df['DR_10'] = df['Open_roll_mean'] / df['Open_roll_geom']
```

[23]:

```
# Drop rows with NaN (from rolling)
df = df.dropna(subset=['DR_10'])
```

[24]:

```
# Final feature set for modeling
features = df[['DR_10']]
target = df['Gamma']
```

[25]:

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import cross_val_score
```

[26]:

```
# Logistic regression classifier
clf = LogisticRegression()
```

[27]:

```
# 10-fold cross-validation
scores = cross_val_score(clf, features, target, cv=10, scoring='accuracy')
```

This ratio was computed daily. Then we created a binary target: 1 if today's open is higher than yesterday's, otherwise 0.
A logistic regression model was evaluated using 10-fold cross-validation. The following table summarizes results:

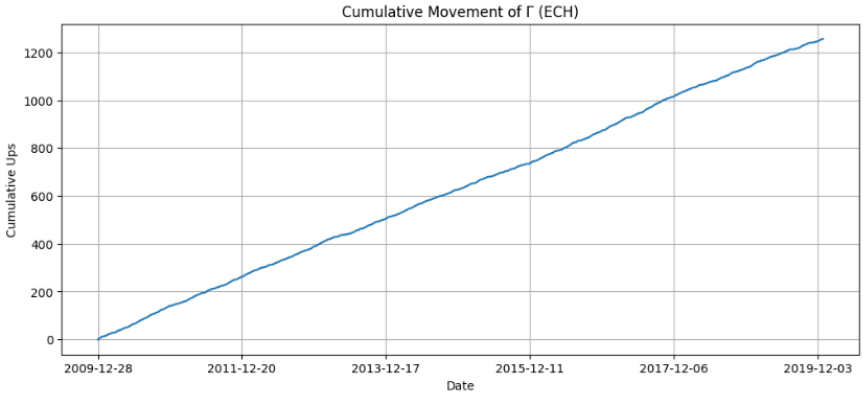
```
# Results
print("Cross-validation accuracy per fold:", scores)
print("Average accuracy:", scores.mean())
```

```
Cross-validation accuracy per fold: [0.50396825 0.50396825 0.50396825 0.50396825 0.5
0.5
 0.5        0.5        0.5        0.5        ]
Average accuracy: 0.5015873015873016
```

The accuracy hovered around 50%—a bit better than random chance. This shows that one indicator alone isn't enough, validating the authors' strategy of combining optimized features.
We also recreated two charts:

- ECH's daily opening price (2009–2020)

- The total count of days where the price went up (cumulative sum of Γ)



References

1)Sagaceta-Mejía, Rodrigo, et al. “An Intelligent Approach for Predicting Stock Market Movements in Emerging Markets Using Optimized Technical Indicators and Neural Networks.” *The Econometrics Journal*, March 25, 2024, https://doi.org/10.1515/econ-2022-0073.
2)Investing.com. “ECH Stock Price | iShares MSCI Chile Capped ETF.” *Investing.com*, https://www.investing.com/etfs/ishares-msci-chile.

Part 2 : EVALUATING ALTERNATIVE DATA: SATELLITE IMAGERY (USER GUIDE)

1.Introduction

Alternative data has become a major channel of help in the financial world, both to finance professionals and investors. It refers to nontraditional data sets that provide timely, detailed and non-official information. Among the categories and sub sequent sub categories mentioned by Sun et al. (2024), satellite imagery has emerged to be one of the most useful sources of alternative data for finance and business.

Satellite imagery data is increasingly used to infer economic activity, monitor assets, forecast revenues, and assess risk in near real time. Applications span equity research, commodities trading, supply-chain monitoring, insurance, and ESG analysis.

This user guide provides a practical and ethical overview of satellite imagery as an alternative data source, focusing on its sources, structure, quality, analytical workflow in Python, and relevance to academic and applied finance research.

2.Sources of Satellite Imagery Data

2.1 Public Sources

Public satellite data is typically free and open-access, though it may have lower spatial or temporal resolution:

NASA (United States), ESA (European Space Agency), USGS Earth Explorer and NOAA

These sources are commonly used in academic research due to transparency and reproducibility.

2.2 Commercial Providers

Commercial satellite firms provide higher-resolution, more frequent imagery and value-added analytics:

- Maxar Technologies
- Planet Labs
- Airbus Defence and Space

- BlackSky
- Satellogic

Though costly, it is highly valuable for financial institutions requiring real-time or intraday signals.

2.3 Data Access Models

- Direct imagery downloads (GeoTIFF, NetCDF)
- API-based access
- Processed indicators (e.g., car counts, night-light intensity)

3.Types of Satellite Imagery Data

Satellite imagery data can be categorized by resolution, spectrum, and analytical output.

3.1 By Spatial and Temporal Resolution

- Low-resolution (250m–1km): Macroeconomic and climate analysis
- Medium-resolution (10–30m): Agricultural output, infrastructure monitoring
- High-resolution (<1m): Vehicle counting, retail traffic, port congestion

Temporal resolution ranges from daily to monthly revisits, depending on satellite constellation size.

3.2 By Spectral Characteristics

- Optical imagery (visible spectrum)
- Infrared and near-infrared (vegetation, heat)
- Radar (SAR) imagery for night and all-weather monitoring

3.3 Derived and Structured Outputs

Rather than raw images, financial users often rely on:

- Night-time light indices (economic activity proxy)
- Object counts (cars, ships, containers)
- Change-detection metrics
- Asset utilization indicators

4. Data Quality Considerations

4.1 Accuracy and Noise

- Cloud cover and atmospheric distortion reduce optical image usability
- Seasonal effects may bias interpretations
- Shadows and occlusion can affect object detection accuracy

4.2 Coverage and Bias

- Uneven geographic coverage (better for developed regions)
- Urban bias due to higher economic density
- Limited historical depth for some commercial datasets

4.3 Validation

Common validation approaches include:

- Cross-referencing with official statistics
- Ground truth data
- Comparison across multiple satellite providers

As Sun et al. (2024) suggests, to have a high-quality research, combine satellite data with traditional datasets.

5. Ethical and Legal Issues

The use of satellite imagery does raise important ethical concerns.

5.1 Privacy

While satellites do not identify individuals directly, high-resolution imagery can indirectly reveal sensitive information about private activities, locations, or communities.

5.2 Market Fairness

- Unequal access due to high costs
- Potential informational advantages for large institutions

5.3 Geopolitical and Security Concerns

- Monitoring critical infrastructure
- Cross-border data usage restriction

That said, best practices include:

- Using aggregated indicators
- Avoiding targeting of individuals or vulnerable populations
- Complying with international data governance standards

6. Python: Importing and Structuring Satellite Data

Below is an illustrative example using night-time light intensity data, a common proxy for economic activity.

```
import pandas as pd
import numpy as np

# Load sample satellite-derived
# Example: night-time light intensity by region and

dat = pd.read_csv("night_lights_sample.csv")

# Inspect
print(data.head())
print(data.info())

# Convert date
data['date'] = pd.to_datetime(data['date'])

# Set multi-index for time series
dat = data.set_index(['region', 'date']).sort_inde

# Summary
summary_stats = data.describe()
print(summary_stats)
```

The resulting structure supports panel data analysis commonly used in finance and econometrics.

7. Exploratory Data Analysis (EDA)

Exploratory data analysis is essential for transforming satellite imagery indicators into investable signals. This section extends basic visualization to include correlation analysis, dimensionality reduction, and panel regressions commonly used in quantitative finance research.

7.1 Correlation Analysis

Satellite-derived indicators are often evaluated against traditional economic or financial variables to assess signal relevance.

Correlation analysis helps identify whether satellite indicators lead, lag, or contemporaneously move with benchmark variables.

7.2 Principal Component Analysis (PCA)

When multiple satellite-derived features are available (e.g., lights, vehicle counts, port congestion), PCA can be used to extract latent economic factors.

The resulting factor can be interpreted as a composite real-activity signal suitable for asset pricing models.

7.3 Panel Regression Framework

Satellite imagery data is naturally suited for panel analysis across regions and time, allowing researchers to control for unobserved heterogeneity.

While pooled OLS is simple, it ignores region-specific and time-specific effects that may bias estimates.

7.4 Fixed Effects Panel Regressions

Fixed effects models control for unobserved, time-invariant characteristics across regions (entity fixed effects) and common macro shocks over time (time fixed effects).

Interpretation

- The coefficient on light intensity measures the within-region effect of changes in satellite derived activity
- Region fixed effects absorb structural differences such as geography or development level
- Time fixed effects control for global shocks (e.g., pandemics, recessions)

7.5 Implications for Quantitative Strategies

- Statistically significant fixed-effects coefficients indicate robust information content
- Signals derived from within-entity variation are less prone to spurious correlations
- Two-way fixed effects models align with best practices in empirical asset pricing and macro-finance research

8. Literature Review and Research Applications

Academic literature strongly supports the use of satellite imagery as a quantitative input in financial decision-making.

8.1 Satellite Data in Asset Pricing and Forecasting

Henderson et al. (2012) establish night-time lights as a robust proxy for economic growth, particularly in regions with weak statistical infrastructure. Babenko et al. (2021) extend this approach by demonstrating that satellite-based urban activity measures predict equity returns and firm fundamentals.

Athey et al. (2020) highlight how machine learning applied to satellite imagery improves measurement accuracy and timeliness relative to traditional datasets. Sun et al. (2024) combine these findings and formally classify satellite imagery as a core alternative data subcategory used by hedge funds, asset managers, and policymakers.

8.2 Implications for Quantitative Finance

- Improved nowcasting of revenues and GDP
- Enhanced cross-sectional stock selection
- Risk monitoring during economic shocks

The literature confirms that satellite imagery contributes statistically and economically significant information beyond standard financial variables.

9. Conclusion

Satellite imagery is one of the most analytically mature and economically meaningful forms of alternative data. Its high frequency, global coverage, and objectivity make it particularly valuable for quantitative finance applications such as nowcasting, factor construction, and risk monitoring. When combined with rigorous validation and econometric techniques, satellite imagery can greatly enhance financial decision-making.

10. References

Athey, S., Levin, J., & Seira, E. (2020). Comparing open and proprietary data sources: Satellite imagery and economic measurement. *Journal of Economic Perspectives*.

Babenko, I., Herskovic, B., Kelly, B., Lustig, H., & Van Nieuwerburgh, S. (2021). Measuring urban economic activity from satellite images. *Journal of Finance*.

Henderson, J. V., Storeygard, A., & Weil, D. N. (2012). Measuring economic growth from outer space. *American Economic Review*.

Sun, Y., et al. (2024). Alternative data in finance and business: emerging applications and theory analysis (review). *Journal of Finance and Data Science*.