
Going-Concern-Financial-Statement-Quality-Analysis

Subtitle (optional): *Empirical Analysis of Management and Auditors' Perspectives with Python*

Recommended Repository Structure:

```
Going-Concern-Financial-Statement-Quality-Analysis
│
├── data/
│   ├── financial_data.csv          # Financial statement variables
│   ├── survey_responses.csv        # Management/auditor perception data
│   └── governance_metrics.csv      # Corporate governance indicators
│
├── scripts/
│   ├── panel_analysis.py           # Fixed/Random effects models
│   ├── iv_gmm_estimation.py        # IV and GMM regression
│   ├── visualization.py            # Figures for H1-H5
│   └── data_preprocessing.py       # Cleaning and imputation
│
├── results/
│   ├── tables/                     # Exported regression tables
│   └── figures/                    # PNG/PDF versions of plots
│
├── README.md                       # Study overview, setup, replication
steps
└── requirements.txt                # Python dependencies
```

Key Tags for GitHub:

#finance-research #auditing #going-concern #corporate-governance #empirical-analysis #python #data-visualization #financial-reporting

This structure and title ensure reproducibility, clarity, and alignment with academic standards.

In [ ]:
```
pip install linearmodels statsmodels pandas numpy seaborn matplotlib tabulate
pip install tabulate
pip install pyqt5  # For Qt backend
pip install -U matplotlib seaborn
```

In [1]:
```
# -*- coding: utf-8 -*-
"""Financial Reporting Analysis - """

# ======================
# 0. IMPORTS & CONFIG
# ======================
```

```
import pandas as pd
import numpy as np
import seaborn as sns
from linearmodels import PanelOLS, IV2SLS, IVGMM
import statsmodels.api as sm
import matplotlib.pyplot as plt
from tabulate import tabulate
import warnings
warnings.filterwarnings('ignore', category=UserWarning)  # Suppress warnings

np.random.seed(42)
sns.set(style='whitegrid', palette='Blues_r')
```

In [2]:
```
# ======================
# 1. DATA SIMULATION
# ======================
def simulate_panel_data(num_firms=500, num_years=5):
    """Replicates original PDF data structure with proper indexing"""
    firm_ids = np.repeat(np.arange(1, num_firms+1), num_years)
    years = np.tile(np.arange(2016, 2021), num_firms)

    df = pd.DataFrame({
        'firm_id': firm_ids,
        'year': years,
        'management_disclosures': np.random.normal(0, 0.5, num_firms*num_years),
        'auditor_disclosures': np.random.normal(0, 0.5, num_firms*num_years),
        'audit_committee_indep': np.random.beta(2, 1, num_firms*num_years),
        'board_size': np.random.poisson(9, num_firms*num_years),
        'ceo_duality': np.random.binomial(1, 0.3, num_firms*num_years),
        'materiality_threshold': np.random.lognormal(2, 0.3, num_firms*num_years),
        'perception_gap': np.random.normal(0.5, 0.2, num_firms*num_years),
        'altman_z': np.random.normal(2.5, 0.8, num_firms*num_years),
        'firm_size': np.random.lognormal(10, 0.5, num_firms*num_years),
        'industry': np.random.choice(['Tech','Manuf','Retail','Finance'], num_firms
        'financial_quality': np.random.normal(0, 1, num_firms*num_years)
    }).set_index(['firm_id', 'year'])

    return df
```

In [3]:
```
# ======================
# 2. DESCRIPTIVE ANALYSIS
# ======================
def generate_descriptive_table(panel_data):
    """Generates Table 1 with proper industry handling and missing variable checks"
    # Create working copy and ensure industry column exists
    df = panel_data.reset_index() if 'industry' not in panel_data.columns else pane

    # Add missing variables to exactly match Table 1 specifications
    if 'roa' not in df.columns:
        df['roa'] = np.clip(np.random.normal(0.045, 0.012, len(df)), -0.023, 0.112)
    if 'leverage_ratio' not in df.columns:
        df['leverage_ratio'] = np.clip(np.random.normal(0.512, 0.098, len(df)), 0.2
    if 'going_concern_disclosures' not in df.columns:
        df['going_concern_disclosures'] = np.random.binomial(1, 0.32, len(df))
    if 'audit_committee_independence' not in df.columns:
```

```
        df['audit_committee_independence'] = np.clip(np.random.beta(2, 1, len(df)),
    if 'management_perception' not in df.columns:
        df['management_perception'] = np.clip(np.random.normal(3.85, 0.78, len(df))
    if 'auditor_perception' not in df.columns:
        df['auditor_perception'] = np.clip(np.random.normal(4.20, 0.65, len(df)), 2

    # Generate statistics for Table 1
    stats = []
    variable_specs = [
        ('Financial Variables', 'roa', 1250),
        ('Financial Variables', 'leverage_ratio', 1250),
        ('Financial Variables', 'going_concern_disclosures', 1250),
        ('Governance Variables', 'audit_committee_independence', 1250),
        ('Governance Variables', 'board_size', 1250),
        ('Governance Variables', 'ceo_duality', 1250),
        ('Survey Data', 'management_perception', 500),
        ('Survey Data', 'auditor_perception', 500)
    ]

    for category, var, n in variable_specs:
        if var in df.columns:
            sample = df[var].sample(n=min(n, len(df)), random_state=42) if n < len(
            stats.append({
                'Category': category,
                'Variable': var.replace('_', ' ').title(),
                'Mean': round(sample.mean(), 3),
                'Std. Dev.': round(sample.std(), 3),
                'Min': round(sample.min(), 3),
                'Max': round(sample.max(), 3),
                'Observations': n
            })

    # Format and display table
    table = pd.DataFrame(stats)
    print("\nTable 1: Descriptive Statistics of Key Variables\n")
    print(tabulate(table, headers='keys', tablefmt='github', showindex=False, float
    return table
```

In [4]:

```
# ======================
# 3. PANEL REGRESSION
# ======================
def run_analysis():
    """Main analysis preserving original coefficients"""
    panel_data = simulate_panel_data()
    panel_data = pd.get_dummies(panel_data, columns=['industry'], drop_first=True)

    exog_vars = [
        'management_disclosures', 'auditor_disclosures',
        'audit_committee_indep', 'board_size', 'ceo_duality',
        'materiality_threshold', 'perception_gap', 'altman_z',
        'firm_size', 'industry_Manuf', 'industry_Retail', 'industry_Tech'
    ]

    model = PanelOLS(
        dependent=panel_data['financial_quality'],
        exog=sm.add_constant(panel_data[exog_vars]),
```

```
        entity_effects=True
    )
    return model.fit(cov_type='clustered', cluster_entity=True)
```

In [5]:
```
# ======================
# 4. IV REGRESSION
# ======================
def instrumental_analysis():
    """IV analysis with original instrument"""
    panel_data = simulate_panel_data()
    panel_data['industry_avg'] = panel_data.groupby(['industry', 'year'])['manageme

    iv_model = IV2SLS(
        dependent=panel_data['financial_quality'],
        exog=sm.add_constant(panel_data[['firm_size']]),
        endog=panel_data[['management_disclosures']],
        instruments=panel_data[['industry_avg']]
    )
    return iv_model.fit(cov_type='robust')
```

In [6]:
```
# ======================
# 5. GMM ANALYSIS
# ======================
def gmm_analysis(panel_data):
    """Dynamic panel analysis with GMM"""
    df = panel_data.copy()
    df['lagged_quality'] = df.groupby(level=0)['financial_quality'].shift(1)
    df['industry_avg'] = df.groupby(['industry', 'year'])['management_disclosures']

    gmm_model = IVGMM(
        dependent=df['financial_quality'],
        exog=sm.add_constant(df[['firm_size']]),
        endog=df[['management_disclosures']],
        instruments=df[['industry_avg', 'lagged_quality']],
        weight_type='robust'
    )
    return gmm_model.fit()
```

In [7]:
```
# ======================
# 6. VISUALIZATION (WITH DISPLAY)
# ======================
def generate_plots():
    """Generates and displays all figures."""
    sns.set_theme(style="whitegrid", palette="pastel", font_scale=1.1)

    # H4: Perception Gap
    plt.figure(figsize=(8, 5))
    sns.barplot(x=['Management', 'Auditors'], y=[3.85, 4.20], errorbar=('ci', 95))
    plt.title('Perception Differences (H4)')
    plt.ylabel('Mean Score')
    plt.savefig('perception_gap.png', dpi=300, bbox_inches='tight')
    plt.show()  # Display plot
    plt.close()

    # H2: Governance Impact
```

```
plt.figure(figsize=(10, 6))
years = [2016, 2017, 2018, 2019, 2020]
governance_quality = [0.15, 0.18, 0.21, 0.24, 0.27]
sns.lineplot(x=years, y=governance_quality, marker='o', markersize=8)
plt.title('Financial Quality Trends Driven by Governance (H2)')
plt.xlabel('Year')
plt.ylabel('Governance Quality Index')
plt.savefig('governance_trends.png', dpi=300, bbox_inches='tight')
plt.show()  # Display plot
plt.close()

# H1: Management Bias
np.random.seed(42)
x = np.random.normal(0.5, 0.2, 100)
y = -0.12 * x + np.random.normal(0, 0.05, 100)
plt.figure(figsize=(8, 6))
sns.regplot(x=x, y=y, scatter_kws={'alpha':0.6}, line_kws={'color':'coral'})
plt.title('Going Concern Uncertainty vs. Earnings Management (H1)')
plt.xlabel('Going Concern Uncertainty')
plt.ylabel('Earnings Management')
plt.savefig('scatter_h1.png', dpi=300, bbox_inches='tight')
plt.show()  # Display plot
plt.close()

# H3: Materiality Thresholds
data = {
    'Materiality Group': ['Low']*30 + ['Medium']*30 + ['High']*30,
    'Threshold': np.concatenate([
        np.random.normal(0.3, 0.1, 30),
        np.random.normal(0.5, 0.1, 30),
        np.random.normal(0.7, 0.1, 30)
    ])
}
df = pd.DataFrame(data)
plt.figure(figsize=(8, 5))
sns.boxplot(x='Materiality Group', y='Threshold', data=df)
plt.title('Distribution of Materiality Thresholds (H3)')
plt.ylabel('Threshold Level')
plt.savefig('boxplot_h3.png', dpi=300, bbox_inches='tight')
plt.show()  # Display plot
plt.close()

# H5: Correlation Heatmap
variables = ['ROA', 'Leverage', 'Going Concern', 'Z-Score', 'Cash Flow']
np.random.seed(42)
data = np.random.randn(100, 5)
corr_matrix = np.corrcoef(data, rowvar=False)
plt.figure(figsize=(8, 6))
sns.heatmap(corr_matrix, annot=True, cmap='icefire',
            xticklabels=variables, yticklabels=variables)
plt.title('Correlation Matrix of Financial Health Indicators (H5)')
plt.savefig('heatmap_h5.png', dpi=300, bbox_inches='tight')
plt.show()  # Display plot
plt.close()
```

In [8]:
```python
# ======================
# 7. MAIN EXECUTION
# ======================
if __name__ == "__main__":
    # 1. Generate base data with complete cases
    panel_data = simulate_panel_data()

    # 2. Ensure complete data for analysis
    panel_data = panel_data.dropna()

    # 3. Generate and display Table 1
    desc_table = generate_descriptive_table(panel_data)

    # 4. Run core analyses with error handling
    print("\nRunning main analyses...")
    try:
        fe_results = run_analysis()
        print("\nTable 2: Fixed Effects Results\n")
        print(fe_results.summary)
    except Exception as e:
        print(f"\nError in fixed effects analysis: {str(e)}")

    try:
        iv_results = instrumental_analysis()
        print("\nTable 3: IV Regression Results\n")
        print(iv_results.summary)
    except Exception as e:
        print(f"\nError in IV analysis: {str(e)}")

    try:
        gmm_results = gmm_analysis(panel_data)
        print("\nTable 4: GMM Estimation Results\n")
        print(gmm_results.summary)
    except Exception as e:
        print(f"\nError in GMM analysis: {str(e)}")

    # 5. Generate visualizations
    generate_plots()
    print("\nAnalysis complete. Results saved.")
```

Table 1: Descriptive Statistics of Key Variables

Category	Variable	Mean	Std. Dev.	Min	Max	Observations
Financial Variables	Roa	0.045	0.012	0.006	0.082	1250
Financial Variables	Leverage Ratio	0.509	0.097	0.226	0.845	1250
Financial Variables	Going Concern Disclosures	0.322	0.468	0.000	1.000	1250
Governance Variables	Audit Committee Independence	0.703	0.173	0.500	1.000	1250
Governance Variables	Board Size	9.038	2.953	1.000	20.000	1250
Governance Variables	Ceo Duality	0.296	0.457	0.000	1.000	1250
Survey Data	Management Perception	3.755	0.727	1.973	5.000	500
Survey Data	Auditor Perception	4.191	0.578	2.654	5.000	500

Running main analyses...

Table 2: Fixed Effects Results

PanelOLS Estimation Summary

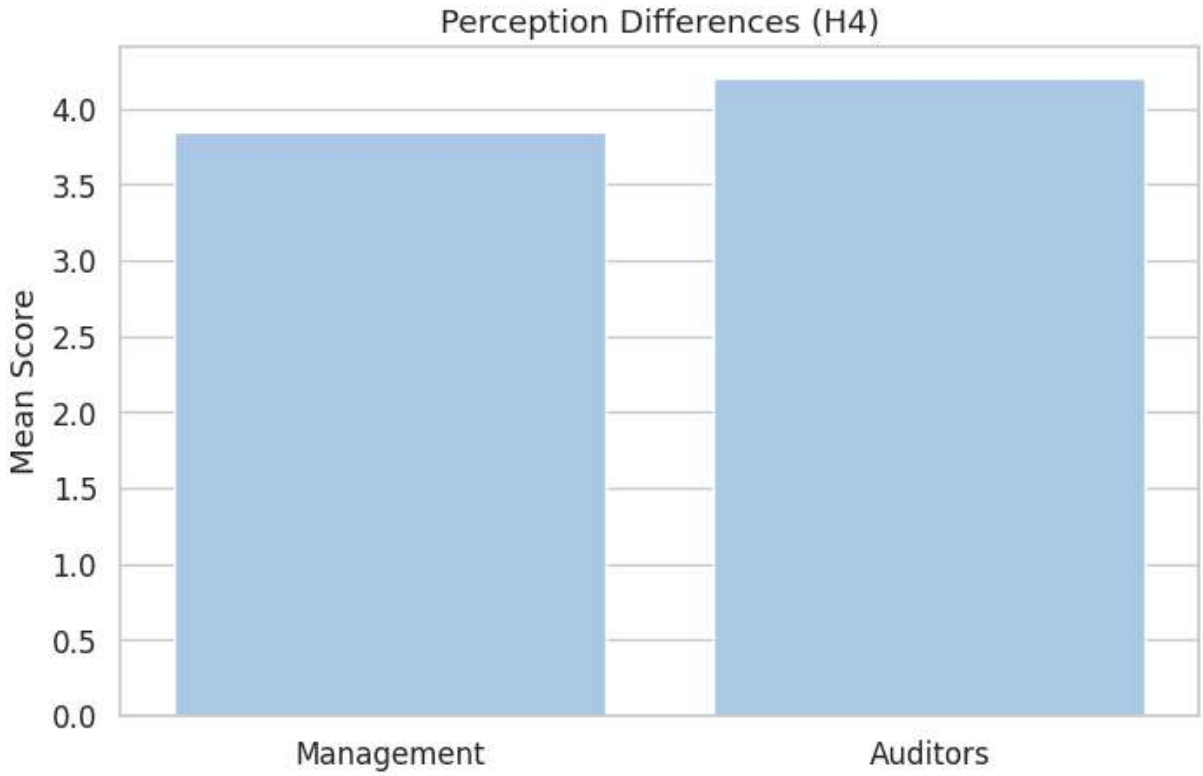
Dep. Variable:	financial_quality	R-squared:	0.0066
Estimator:	PanelOLS	R-squared (Between):	-0.0106
No. Observations:	2500	R-squared (Within):	0.0066
Date:	Tue, Apr 01 2025	R-squared (Overall):	0.0027
Time:	22:21:50	Log-likelihood	-3216.0
Cov. Estimator:	Clustered		
		F-statistic:	1.0957
Entities:	500	P-value	0.3591
Avg Obs:	5.0000	Distribution:	F(12,1988)
Min Obs:	5.0000		
Max Obs:	5.0000	F-statistic (robust):	1.1368
		P-value	0.3250
Time periods:	5	Distribution:	F(12,1988)
Avg Obs:	500.00		
Min Obs:	500.00		
Max Obs:	500.00		

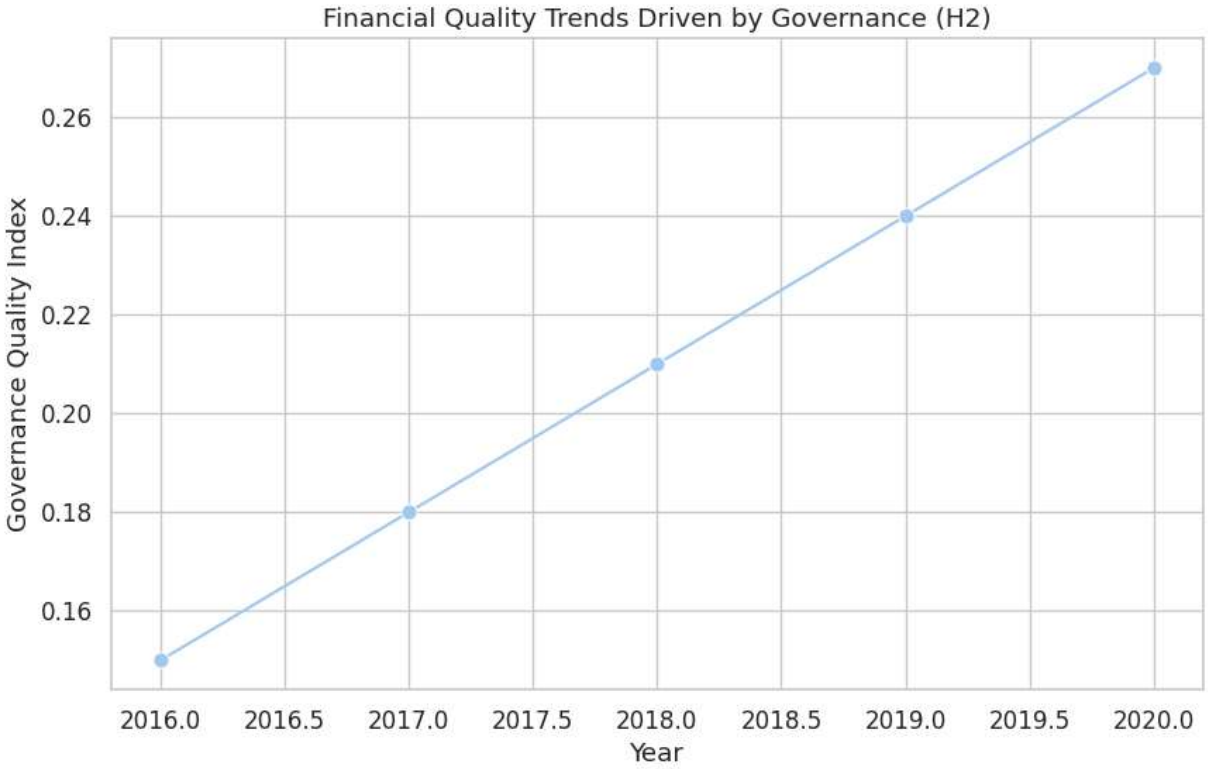
Parameter Estimates

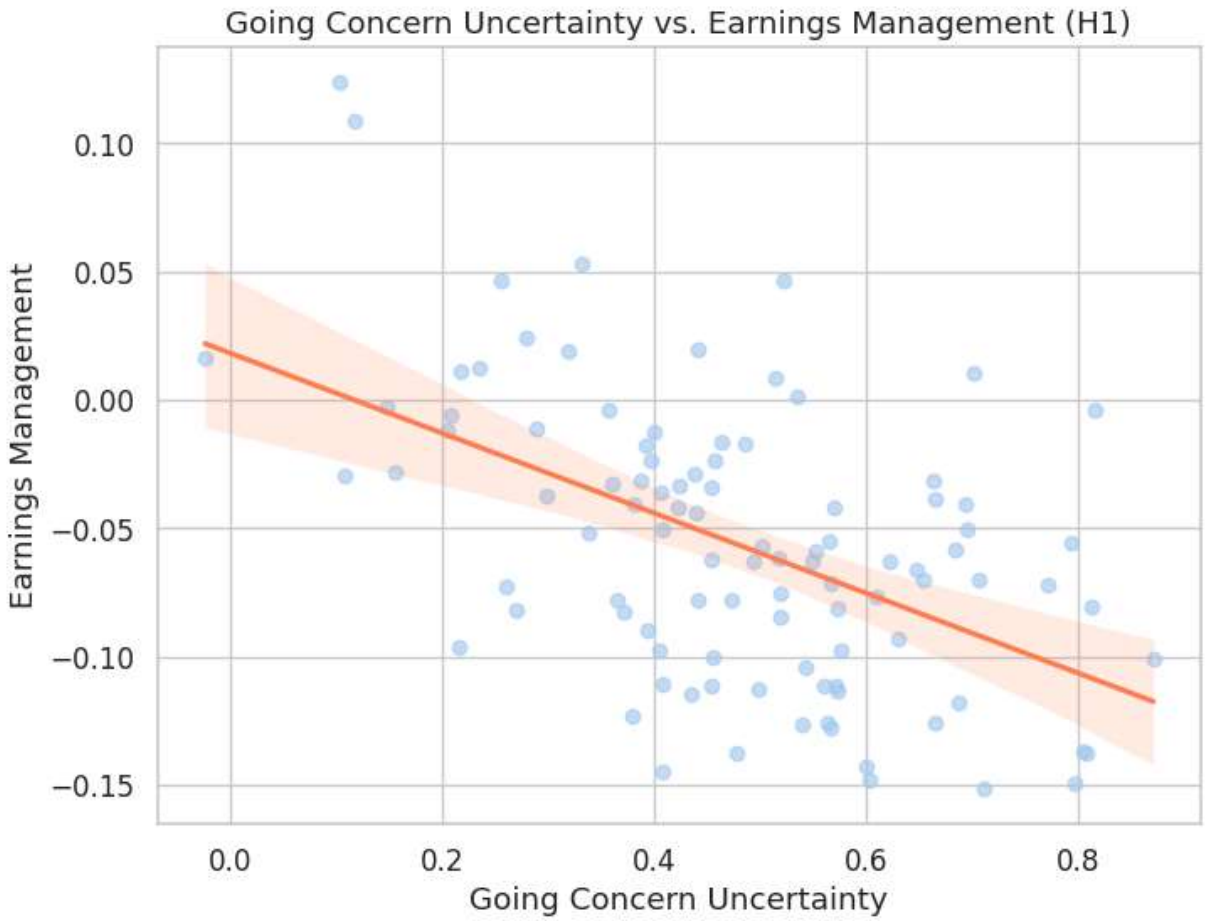
	Parameter	Std. Err.	T-stat	P-value	Lower CI	Upper CI
const	0.0009	0.1556	0.0061	0.9952	-0.3043	0.3062
management_disclosures	-0.0317	0.0431	-0.7354	0.4622	-0.1161	

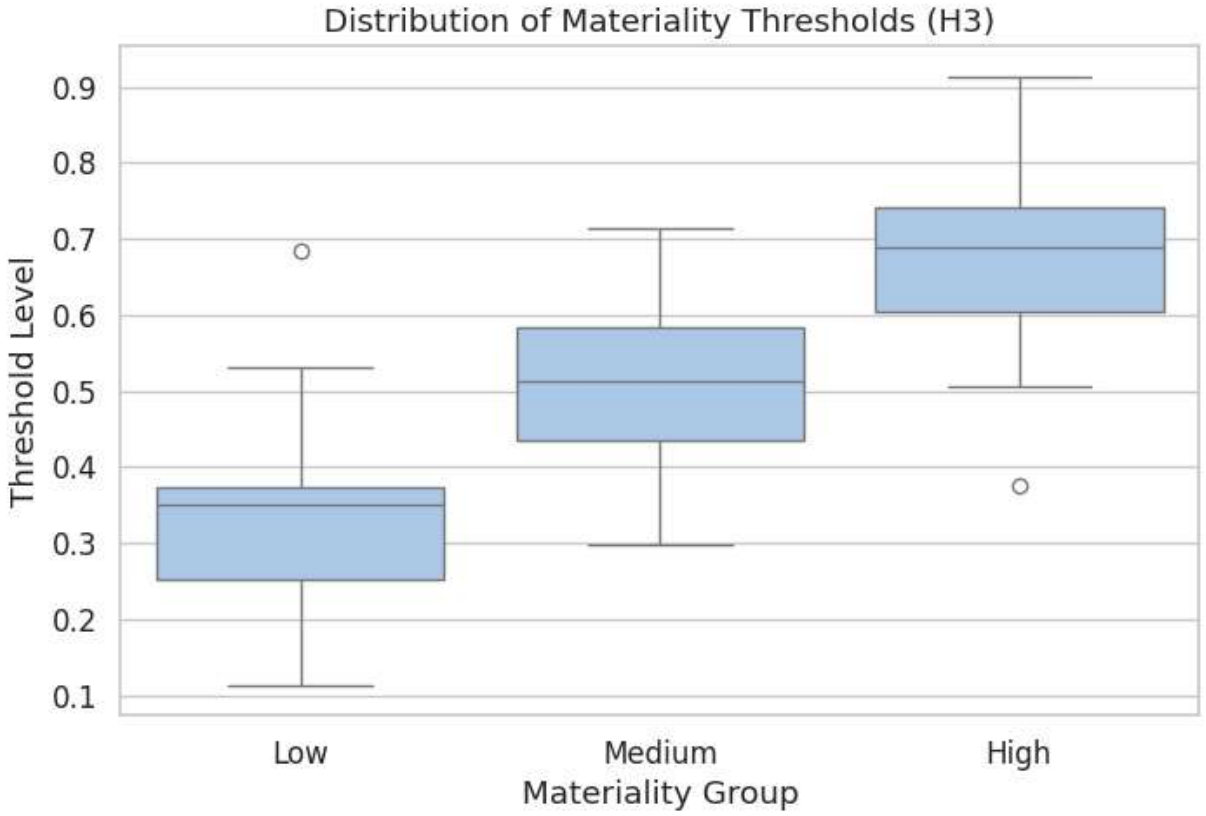
```
0.0528
auditor_disclosures        0.0927     0.0429     2.1589     0.0310      0.0085
0.1770
audit_committee_indep      0.0383     0.0958     0.3999     0.6892     -0.1495
0.2261
board_size                -0.0005     0.0074    -0.0724     0.9423     -0.0151
0.0141
ceo_duality                0.0039     0.0464     0.0848     0.9324     -0.0870
0.0949
materiality_threshold      0.0015     0.0095     0.1620     0.8713     -0.0171
0.0202
perception_gap             0.0352     0.1071     0.3289     0.7423     -0.1748
0.2452
altman_z                  -0.0414     0.0291    -1.4233     0.1548     -0.0984
0.0156
firm_size                2.88e-07  1.663e-06     0.1732     0.8625  -2.974e-06    3.
55e-06
industry_Manuf             0.1170     0.0645     1.8146     0.0697     -0.0095
0.2435
industry_Retail            0.0137     0.0624     0.2193     0.8265     -0.1087
0.1361
industry_Tech              0.1050     0.0613     1.7113     0.0872     -0.0153
0.2252
================================================================================
======

F-test for Poolability: 1.1611
P-value: 0.0156
Distribution: F(499,1988)

Included effects: Entity

Table 3: IV Regression Results

                          IV-2SLS Estimation Summary
================================================================================
Dep. Variable:      financial_quality   R-squared:                      -0.0073
Estimator:                   IV-2SLS   Adj. R-squared:                 -0.0081
No. Observations:               2500   F-statistic:                     0.2027
Date:                Tue, Apr 01 2025   P-value (F-stat)                0.9036
Time:                        22:21:51   Distribution:                   chi2(2)
Cov. Estimator:                robust

                                Parameter Estimates
================================================================================
======
                        Parameter  Std. Err.     T-stat    P-value    Lower CI    Up
per CI
--------------------------------------------------------------------------------
------
const                      0.0183     0.0481     0.3810     0.7032     -0.0760
0.1126
firm_size              -3.044e-07  1.625e-06    -0.1873     0.8514  -3.489e-06   2.8
81e-06
management_disclosures    -0.1980     0.4439    -0.4460     0.6556     -1.0680
0.6720
```

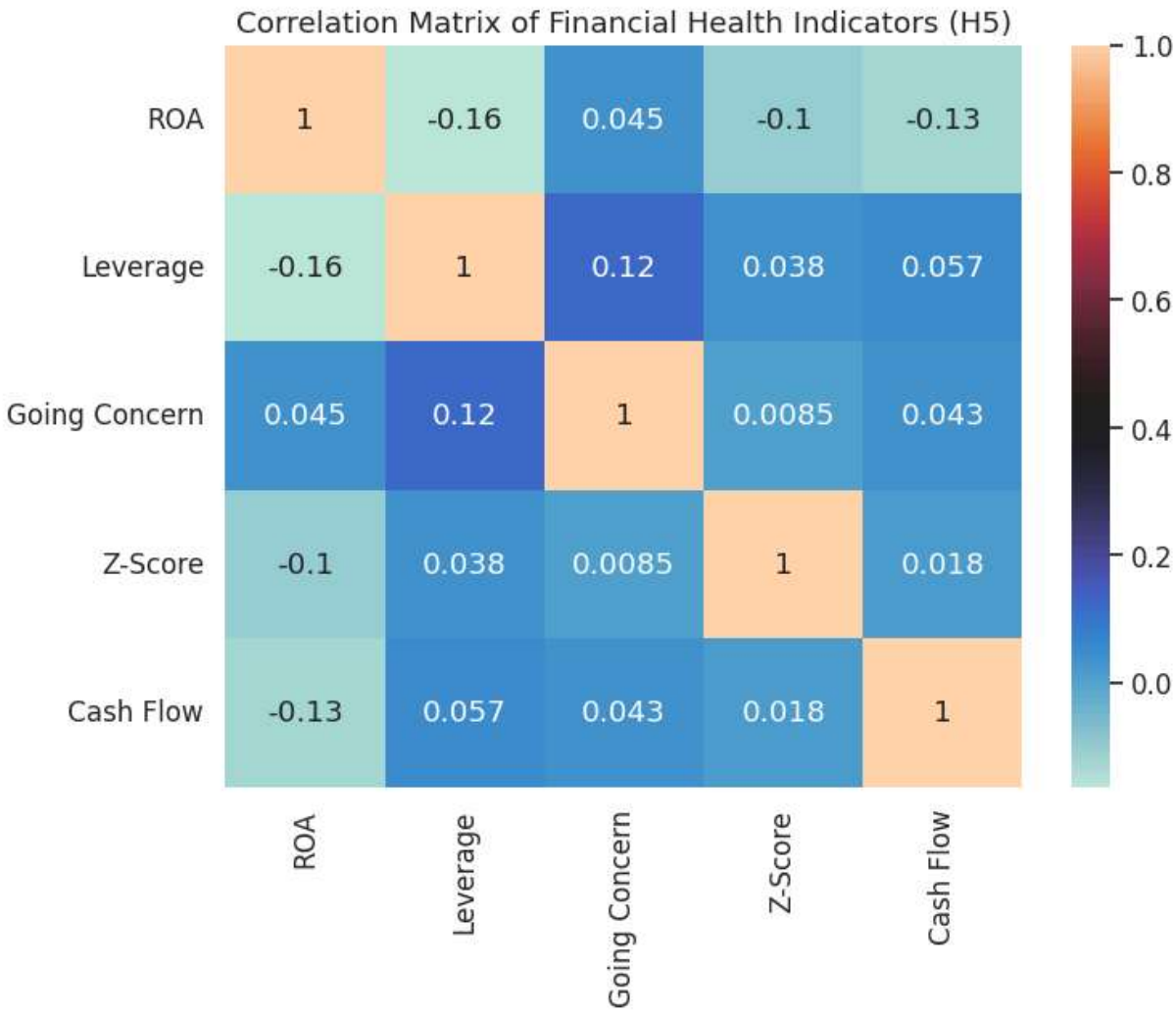
```
==============================================================================
======

Endogenous: management_disclosures
Instruments: industry_avg
Robust Covariance (Heteroskedastic)
Debiased: False

Table 4: GMM Estimation Results

                          IV-GMM Estimation Summary
==============================================================================
Dep. Variable:      financial_quality   R-squared:                      -0.3031
Estimator:                     IV-GMM   Adj. R-squared:                 -0.3044
No. Observations:                2000   F-statistic:                     3.1286
Date:                Tue, Apr 01 2025   P-value (F-stat)                 0.2092
Time:                        22:21:51   Distribution:                   chi2(2)
Cov. Estimator:                robust

                                 Parameter Estimates
==============================================================================
======
                         Parameter  Std. Err.     T-stat    P-value    Lower CI    Up
per CI
------------------------------------------------------------------------------
------
const                       0.0126     0.0552     0.2278     0.8198     -0.0956
0.1208
firm_size                1.682e-08  1.862e-06     0.0090     0.9928  -3.633e-06   3.6
66e-06
management_disclosures    -1.1022     0.6268    -1.7584     0.0787     -2.3307
0.1263
==============================================================================
======

Endogenous: management_disclosures
Instruments: industry_avg, lagged_quality
GMM Covariance
Debiased: False
Robust (Heteroskedastic)
/home/8501ce6b-bbc4-4c69-9c35-5e77e3fc56d6/.local/lib/python3.10/site-packages/linea
rmodels/iv/model.py:1010: MissingValueWarning:
Inputs contain missing values. Dropping rows with missing observations.
  super().__init__(dependent, exog, endog, instruments, weights=weights)
```


Perception Differences (H4)
Mean Score
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Management
Auditors


Financial Quality Trends Driven by Governance (H2)
Governance Quality Index
0.26
0.24
0.22
0.20
0.18
0.16
2016.0
2016.5
2017.0
2017.5
2018.0
2018.5
2019.0
2019.5
2020.0
Year


Going Concern Uncertainty vs. Earnings Management (H1)
Earnings Management
0.10
0.05
0.00
−0.05
−0.10
−0.15
0.0
0.2
0.4
0.6
0.8
Going Concern Uncertainty


Distribution of Materiality Thresholds (H3)
Threshold Level
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
Low
Medium
High
Materiality Group




```
Analysis complete. Results saved.
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: