GROUP WORK PROJECT # 1
GROUP NUMBER: 9644

MScFE 610: FINANCIAL ECONOMETRICS

FULL LEGAL NAME	LOCATION (COUNTRY)	EMAIL ADDRESS	MARK X FOR ANY NON-CONTRIBUTING MEMBER
Nilay Jayantibhai Ganvit	India	nilayganvit252@gmail.com	
Kelvin Theophilus	Indonesia	theophilus.kelvin@gmail.com	
Junchao Huang	US	junchao@huangs.io	

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	**Nilay Jayantibhai Ganvit**
Team member 2	Kelvin Theophilus
Team member 3	Junchao Huang

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.

MScFE 610 Financial Econometrics
Group Work Project # 1

Problem 1

1a)

No, μ(i) does not necessarily satisfy the standard assumptions. μ(i) = δz(i) + ε(i). If z(i) is correlated with x(i) or w(i), then μ(i) is also correlated with those regressors, violating the exogeneity assumption which required for OLS to be unbiased. Even if ε(i) meets the assumptions, μ(i) likely won't due to this correlation.

Reference:

Wooldridge, J. M. (2010). *Econometric Analysis of Cross Section and Panel Data (2nd ed.)*. MIT Press.

1b)

The estimates of α, β, and γ from model (2) are biased and inconsistent if z(i) is correlated with x(i) or w(i). This is due to omitted variable bias—part of z(i)'s effect is wrongly attributed to x(i) and w(i), distorting their estimated coefficients.

Reference:

Stock, J. H., & Watson, M. W. (2015). *Introduction to Econometrics (3rd ed.)*. Pearson.

1c)

The estimates from model (2) will match model (1) only if z(i) is uncorrelated with x(i) and w(i). In that case, δz(i) doesn't bias the estimates, and μ(i) behaves like a proper error term.

Reference:

Gujarati, D. N., & Porter, D. C. (2009). *Basic Econometrics (5th ed.)*. McGraw-Hill.

1d)

Steps:

1. Generate data:
   - Let $Z = 0.5X + \eta$ (so X and Z are correlated)
   - $Y = a + bX + cZ + \varepsilon$
2. Estimate:
   - Full model: Y ~ X + Z
   - Misspecified model: Y ~ X (omit Z)
3. Compare coefficients for X:
   - In the misspecified model, X's coefficient is biased because it captures part of Z's effect.
   - Increasing sample size doesn't fix the bias—it persists because it's structural, not random.

Reference:

Angrist, J. D., & Pischke, J.-S. (2009). *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton University Press.

Problem 2

2a)

Regression models, especially ordinary least squares (OLS) linear regression, are highly sensitive to outliers because they minimize the sum of squared residuals. Squaring residuals gives more weight to large errors, so outliers can disproportionately influence the estimated coefficients. Outliers can skew results by pulling the regression line toward themselves, causing biased parameter estimates and misleading inferences.
Outliers may occur due to measurement errors or rare but real events. Points with extreme predictor values (leverage points) can exert especially strong influence. Robust regression methods and diagnostic tools (e.g., Cook's distance) help detect and reduce the impact of outliers. Addressing outliers ensures more reliable and interpretable regression models. In financial datasets, such as returns or asset prices, outliers may reflect real market shocks. Therefore, careful evaluation is needed before removing them.

References:

Montgomery, D. C., et al. (2012). *Introduction to Linear Regression Analysis*. Wiley.
Rousseeuw, P. J., & Leroy, A. M. (1987). *Robust Regression and Outlier Detection*. Wiley.

2b)

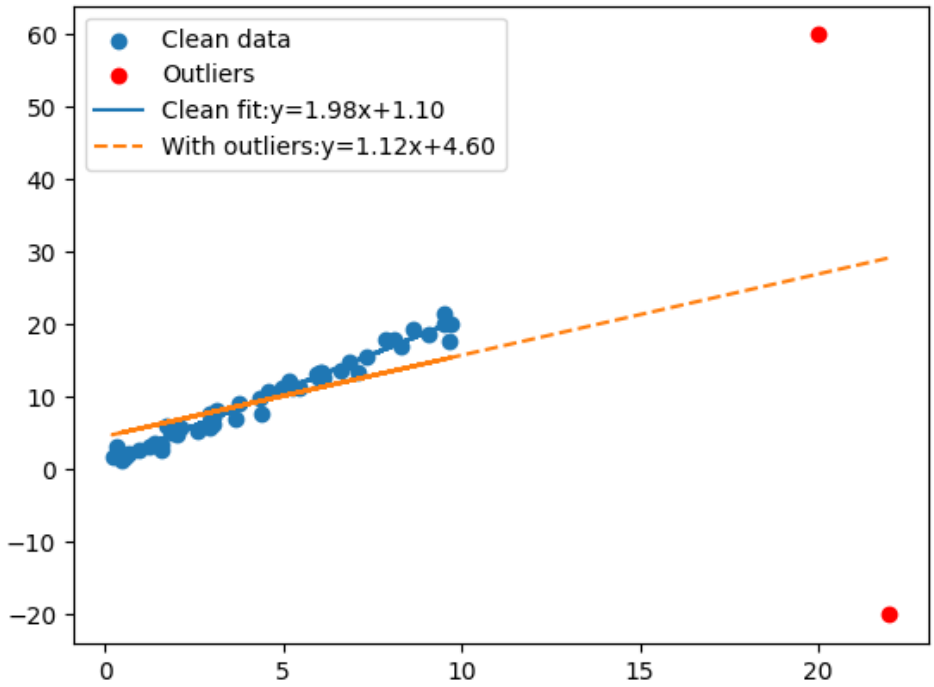
The following simulation demonstrates that the presence of just two outliers dramatically changes slope and intercept, illustrating how sensitive OLS regression is to extreme points.

```
X = np.random.uniform(0, 10, 50).reshape(-1,1)
y = 2 * X.flatten() + 1 + np.random.normal(0, 1, 50)

model_clean = LinearRegression().fit(X, y)

X_out = np.append(X, [[20], [22]], axis=0)
y_out = np.append(y, [60, -20])

model_out = LinearRegression().fit(X_out, y_out)
```



The regression line with outliers has a noticeably steeper or flatter slope and a different intercept, confirming how just a few extreme points can distort parameter estimation.

Problem 3

3a)

To find the best regression model for predicting Y using the independent variables Z1–Z5, we apply two model selection techniques: Forward Selection using Adjusted R^2 and Backward Elimination using BIC. Each approach offers a balance between model fit and complexity, avoiding overfitting.

```
# Forward Selection
def forward_selection(X, y):
    remaining = list(X.columns)
    selected = []
    current_score, best_new_score = 0, 0

    while remaining:
        scores_with_candidates = []
        for candidate in remaining:
            model = sm.OLS(y, sm.add_constant(df[selected + [candidate]])).fit()
            scores_with_candidates.append((model.rsquared_adj, candidate))
        scores_with_candidates.sort(reverse=True)
        best_new_score, best_candidate = scores_with_candidates[0]
        if best_new_score > current_score:
            remaining.remove(best_candidate)
            selected.append(best_candidate)
            current_score = best_new_score
        else:
            break
    return selected, current_score

# Backward Elimination using BIC
def backward_elimination(X, y):
    selected = list(X.columns)
    current_model = sm.OLS(y, sm.add_constant(X[selected])).fit()
    current_bic = current_model.bic

    while True:
        bics = []
        for candidate in selected:
            reduced = selected.copy()
            reduced.remove(candidate)
            model = sm.OLS(y, sm.add_constant(X[reduced])).fit()
            bics.append((model.bic, candidate))
        min_bic, worst = min(bics)
        if min_bic < current_bic:
            selected.remove(worst)
            current_bic = min_bic
        else:
            break
    return selected, current_bic
```

```
Forward Selection (Adj. R²): ['X4', 'X3', 'X2', 'X5'] | Score: 0.634
Backward Elimination (BIC): ['X2', 'X3', 'X4', 'X5'] | BIC: 273.64
```

Adjusted R^2 penalizes model complexity, making it more suitable than plain R^2. BIC applies a stronger penalty than AIC for the number of parameters, favoring simpler models.

References:
Draper, N. R., & Smith, H. (1998). *Applied Regression Analysis (3rd ed.)*. Wiley.
Schwarz, G. (1978). *Estimating the dimension of a model*. The Annals of Statistics.

Problem 4

4a)

(a) Not constant, varies with x and y.
(b) Elasticity = 0.4, constant.
(c) Elasticity = 0.25 (coefficient of ln (x)), this is a log-log model, ideal for elasticity.
(d) Depends on y, not constant

- Elasticity = (dy/dx) * (x/y) = 1.2 * (1/y), which varies depending on y.

4b)

Model (c) is the correct one. In the log-log specification, the coefficient b directly represents the elasticity of y with respect to x and is constant.

Reference:
Wooldridge, J. M. (2012). *Introductory Econometrics: A Modern Approach (5th ed.).* Cengage Learning.

Problem 5

5a)

Stationarity means that a time series' statistical properties (mean, variance, autocorrelation) are constant over time. A unit root indicates non-stationarity, often leading to spurious regression results.

To test:

- Use Augmented Dickey-Fuller (ADF) or Phillips-Perron tests.
- If the test fails to reject the null hypothesis, the series has a unit root (non-stationary).

If unit root is present:

- First-difference the data: $\Delta y(t) = y(t) - y(t-1)$
- If differenced data is stationary, proceed with analysis using the differenced series.

Differencing removes the unit root, converting a non-stationary series into a stationary one suitable for regression and forecasting.

Reference:
Enders, W. (2014). *Applied Econometric Time Series (4th ed.)*. Wiley.

5b)

Use historical stock price data (e.g., S&P 500 closing prices). Run the ADF test on the log of prices:

- Result: Prices typically fail the unit root test → non-stationary.
- Apply first differences: Δln (Pt) -> log returns.
- The differenced data usually passes the test, confirming stationarity in returns.

Reference:
Tsay, R. S. (2010). *Analysis of Financial Time Series (3rd ed.)*. Wiley.

5c)
A unit root implies a random walk: shocks have permanent effects. Over time, variance grows without bound, making predictions unreliable. A root of 1.5 is explosive: the series diverges even faster, but this is easier to detect and typically not seen in practice.

Why economists care about a unit root:
- Difficult to forecast.
- Non-stationary models lead to misleading inference unless differenced.

Reference:
Hamilton, J. D. (1994). *Time Series Analysis*. Princeton University Press.

Problem 6
6a)
To check whether the relationship between Y and X changes after time t = 10, we can use a dummy variable in a single regression instead of splitting the data.
Steps:
1. Create a dummy variable that equals 0 for t = 1 to 10, and 1 for t = 11 to 20. This helps us separate the two time periods.
2. Create another variable by multiplying this dummy variable with X. This new variable allows the slope of X to change after t = 10.
3. Run a regression that includes:
    - The original X variable
    - The dummy variable × X interaction

If the interaction term is statistically significant, it means the slope of X changes after t = 10. This change suggests a structural break—in other words, the effect of X on Y is different before and after t = 10.
This approach lets us test for changes in just one regression, which is simple and efficient.

```
# Create time, X, and error terms
t = np.arange(1, 21)
X = np.random.normal(0, 1, 20)
error = np.random.normal(0, 0.5, 20)

# Create Y with a slope change at t = 10
Y = np.where(t <= 10, 1.0 * X, 2.0 * X) + error

# Create dummy variable: 0 for t ≤ 10, 1 for t > 10
D = (t > 10).astype(int)

# Interaction term: dummy × X
X_interaction = D * X

# Build regression data
df = pd.DataFrame({'Y': Y, 'X': X, 'X_interaction': X_interaction})
X_model = sm.add_constant(df[['X', 'X_interaction']])
model = sm.OLS(df['Y'], X_model).fit()
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      Y   R-squared:                       0.906
Model:                            OLS   Adj. R-squared:                  0.895
Method:                 Least Squares   F-statistic:                     82.16
Date:                Sun, 15 Jun 2025   Prob (F-statistic):           1.83e-09
Time:                        21:38:37   Log-Likelihood:                -13.083
No. Observations:                  20   AIC:                             32.17
Df Residuals:                      17   BIC:                             35.15
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -0.1292      0.149     -0.869      0.397      -0.443       0.185
X              0.8873      0.219      4.057      0.001       0.426       1.349
X_interaction  1.0583      0.318      3.324      0.004       0.386       1.730
==============================================================================
Omnibus:                        1.143   Durbin-Watson:                   2.142
Prob(Omnibus):                  0.565   Jarque-Bera (JB):                0.678
Skew:                           0.445   Prob(JB):                        0.713
Kurtosis:                       2.854   Cond. No.                         4.28
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

This approach is known as the **Chow test** variant using interaction terms, often used to test structural breaks.

Reference:

Gujarati, D. N., & Porter, D. C. (2009). *Basic Econometrics (5th ed.).* McGraw-Hill.