**Question 1**

import pandas as pd

import statsmodels.api as sm

import statsmodels.formula.api as smf

import matplotlib.pyplot as plt

import seaborn as sns

from scipy import stats # Add this import

# Load the dataset

df = pd.read\_csv('individual\_data.csv')

variables= ['Voted\_Yabloko\_1995', 'Voted\_KPRF\_1995', 'Voted\_reported\_1995',

'age1995', 'educ1995', 'married1995',

'vote\_Unity', 'vote\_KPRF', 'vote\_Yabloko', 'vote\_reported','NTV\_received', 'Watches\_NTV\_1999',

'NTV1999', 'age', 'male', 'educ1',

'married', 'NewspapersPolitics', 'RadioPolitics']

df\_subset = df[variables]

summary\_stats = pd.DataFrame(columns=['count','mean','std\_dev','min','max','t-stat','p-value'])

test\_value =0

for column in variables:

mean = df\_subset[column].mean()

std\_dev = df\_subset[column].std()

count = df\_subset[column].count()

min = df\_subset[column].min()

max = df\_subset[column].max()

t\_stat,p\_value = stats.ttest\_1samp(df\_subset[column].dropna(),test\_value)

summary\_stats.loc[column]=[count, mean, std\_dev, min, max,t\_stat, p\_value]

summary\_stats\_combined = summary\_stats.round(2)

summary\_stats\_combined.to\_excel("summary\_statistics.xlsx", index=True)

# Display the summary table

summary\_stats\_combined

plt.figure(figsize=(10,6))

parties = ['vote\_Unity', 'vote\_KPRF', 'vote\_Yabloko']

for party in parties:

means = df.groupby('Watches\_NTV\_1999')[party].mean()

plt.bar(party + '\_No\_NTV', means[0], label='No NTV')

plt.bar(party + '\_NTV', means[1], label='Has NTV')

plt.title('Voting Patterns by NTV Access')

plt.ylabel('Vote Share')

plt.xticks(rotation=45)

plt.legend()

plt.show()

from matplotlib.backends.backend\_pdf import PdfPages

with PdfPages("variable\_distributions.pdf") as pdf:

for var in variables:

plt.figure()

sns.histplot(df\_subset[var].dropna(), kde=True)

plt.title(f'Distribution of {var}')

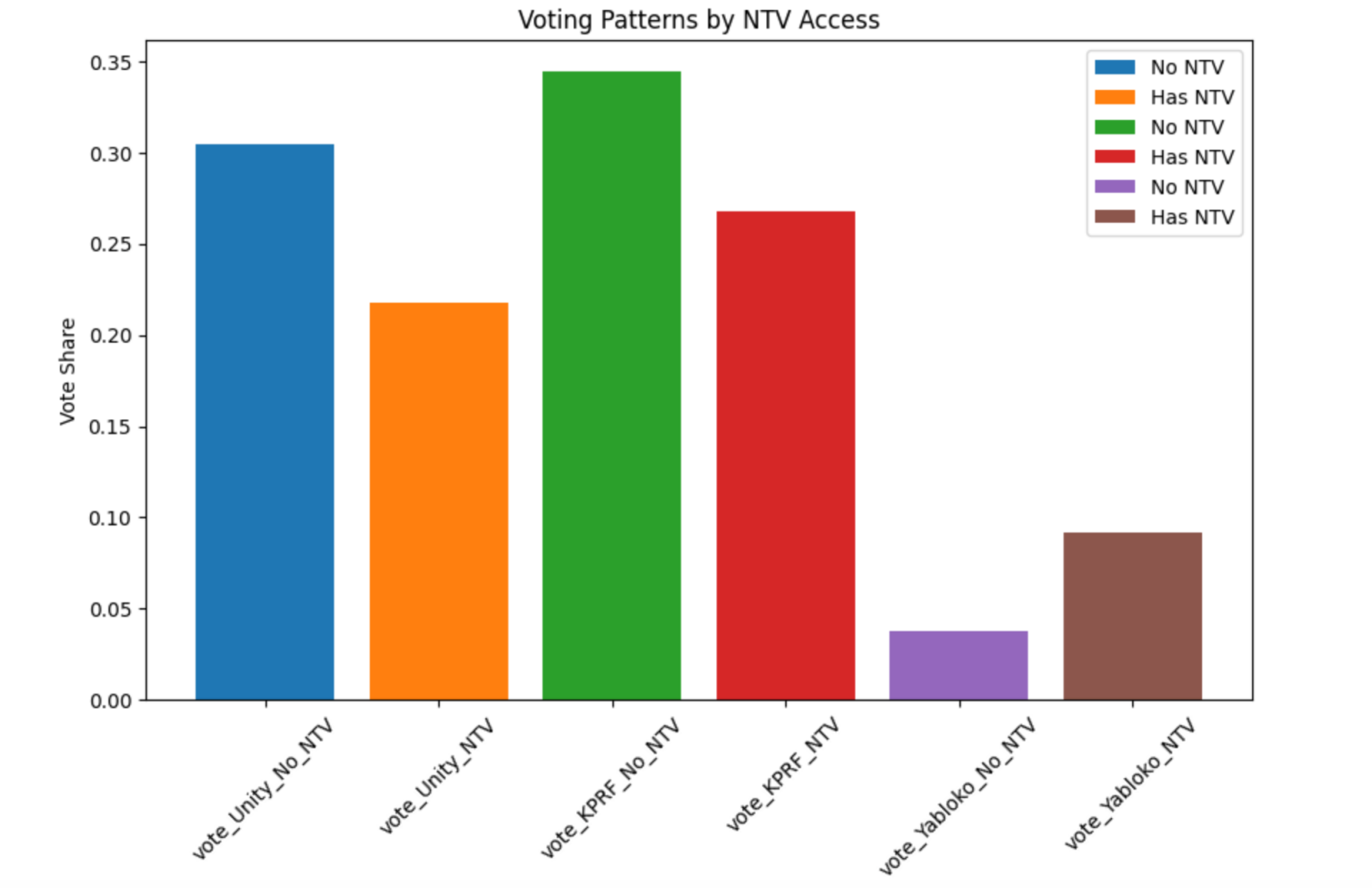
plt.xlabel(var)

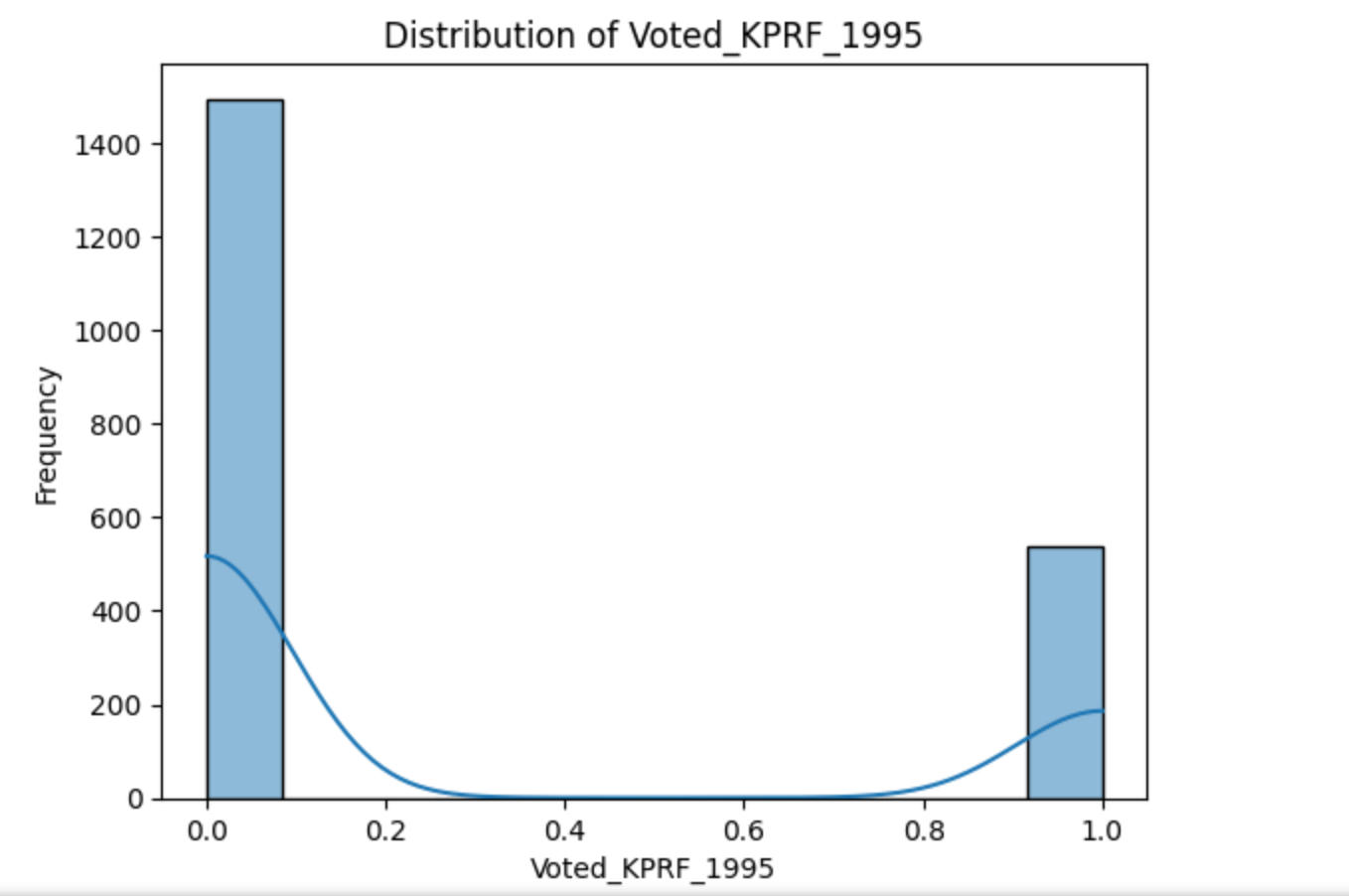
plt.ylabel('Frequency')

pdf.savefig() # Save the current figure to the PDF

plt.show()

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **count** | **mean** | **std\_dev** | **min** | **max** | **t-stat** | **p-value** |
| **Voted\_Yabloko\_1995** | 2034 | 0.1 | 0.3 | 0 | 1 | 14.77 | 0 |
| **Voted\_KPRF\_1995** | 2034 | 0.26 | 0.44 | 0 | 1 | 27.04 | 0 |
| **Voted\_reported\_1995** | 2532 | 0.81 | 0.39 | 0 | 1 | 104.29 | 0 |
| **age1995** | 2594 | 28.51 | 16.59 | -21 | 75 | 87.55 | 0 |
| **educ1995** | 2591 | 0.72 | 0.45 | 0 | 1 | 82.36 | 0 |
| **married1995** | 2587 | 0.61 | 0.49 | 0 | 1 | 64.11 | 0 |
| **vote\_Unity** | 1311 | 0.24 | 0.43 | 0 | 1 | 20.48 | 0 |
| **vote\_KPRF** | 1311 | 0.3 | 0.46 | 0 | 1 | 23.68 | 0 |
| **vote\_Yabloko** | 1311 | 0.07 | 0.26 | 0 | 1 | 10.12 | 0 |
| **vote\_reported** | 1708 | 0.79 | 0.41 | 0 | 1 | 80.52 | 0 |
| **NTV\_received** | 1783 | 0.76 | 0.43 | 0 | 1 | 74.88 | 0 |
| **Watches\_NTV\_1999** | 1624 | 0.63 | 0.48 | 0 | 1 | 52.49 | 0 |
| **NTV1999** | 3499 | 0.62 | 0.49 | 0 | 1 | 74.89 | 0 |
| **age** | 1783 | 30.79 | 17.28 | 0 | 71 | 75.23 | 0 |
| **male** | 5958 | 0.11 | 0.31 | 0 | 1 | 26.68 | 0 |
| **educ1** | 1776 | 0.77 | 0.42 | 0 | 1 | 77.15 | 0 |
| **married** | 1778 | 0.56 | 0.5 | 0 | 1 | 47.47 | 0 |
| **NewspapersPolitics** | 1783 | 0.28 | 0.45 | 0 | 1 | 26.39 | 0 |
| **RadioPolitics** | 1783 | 0.36 | 0.48 | 0 | 1 | 31.55 | 0 |





ps. The rest of the distribution plots are in the pdf file named ‘Appendix.pdf’.

**Explanation of Variable Selection:**

The selected variables are chosen to represent a mix of demographic, economic, and political indicators. They include:

Demographics: age1995, educ1995, male, married1995, age.

Political-engagement: Voted\_Yabloko\_1995, Voted\_KPRF\_1995, Voted\_reported\_1995, vote\_Unity, vote\_KPRF, vote\_Yabloko, vote\_reported.

Media access and exposure: NTV\_received, Watches\_NTV\_1999, NTV1999.

Political media consumption: NewspapersPolitics, RadioPolitics.

These variables allow us to investigate relationships between media exposure, demographic factors, and political outcomes such as voting patterns.

**Explanation of Methods:**

Descriptive Statistics:

Summary statistics were calculated for each variable (count, mean, standard deviation, min, max).

One-sample t-tests were conducted to assess whether the means of these variables differ significantly from a hypothesized value (test\_value = 0).

Visualization:

Histograms were generated to examine the distributions of all variables. This helps in identifying skewness, outliers, or multimodal distributions.

Bar plots were created to compare voting patterns between individuals with and without access to NTV, illustrating the potential influence of media on political preferences.

**Observed Patterns:**

Summary Statistics:

Variables such as age1995 and age are normally distributed with meaningful ranges and standard deviations.

Political variables like vote\_Yabloko have means that suggest relatively lower levels of support, as indicated by their mean values.

Voting Patterns by Media Access:

The bar plot comparing voting shares by NTV access reveals trends suggesting that NTV exposure might correlate with higher support for certain parties.

This pattern underscores the hypothesis that media access impacts voter preferences.

Variable Distributions:

Binary variables show clear categorical splits, as expected.

**Question 2**

import statsmodels.formula.api as smf

import numpy as np

# First, filter the data to include only those who reported voting

question2\_vars = ['vote\_Unity', 'vote\_KPRF', 'vote\_Yabloko', 'NTV1999', 'NTV\_received',

'Watches\_NTV\_1999', 'male', 'age', 'educ1', 'married', 'vote\_reported']

# Create a copy of the data with only the variables we need

question2\_data = df[question2\_vars].copy()

# Clean the data

question2\_data.replace([np.inf, -np.inf], np.nan, inplace=True)

question2\_data.dropna(inplace=True)

# Define OLS regression for Unity

model\_ols\_formula\_Unity = 'vote\_Unity ~ Watches\_NTV\_1999 + age + educ1 + male + married'

model\_ols\_Unity = smf.ols(formula=model\_ols\_formula\_Unity, data=question2\_data).fit(cov\_type='HC3')

# Define OLS regression for KPRF

model\_ols\_formula\_KPRF = 'vote\_KPRF ~ Watches\_NTV\_1999 + age + educ1 + male + married'

model\_ols\_KPRF = smf.ols(formula=model\_ols\_formula\_KPRF, data=question2\_data).fit(cov\_type='HC3')

# Define OLS regression for Yabloko

model\_ols\_formula\_Yabloko = 'vote\_Yabloko ~ Watches\_NTV\_1999 + age + educ1 + male + married'

model\_ols\_Yabloko = smf.ols(formula=model\_ols\_formula\_Yabloko, data=question2\_data).fit(cov\_type='HC3')

# Display the results for each party

print("Unity Party OLS Regression Results:")

print(model\_ols\_Unity.summary())

print("KPRF Party OLS Regression Results:")

print(model\_ols\_KPRF.summary())

print("Yabloko Party OLS Regression Results:")

print(model\_ols\_Yabloko.summary())

def plot\_ols\_coefficients(models, titles):

# Create a figure with three subplots side by side

fig, axes = plt.subplots(1, 3, figsize=(20, 6))

for idx, (model, title) in enumerate(zip(models, titles)):

# Get coefficients and their standard errors (excluding intercept)

coef = model.params[1:] # Exclude intercept

std\_err = model.bse[1:] # Exclude intercept

# Create y-axis labels (variable names)

labels = coef.index

# Plot points and error bars

y\_pos = np.arange(len(labels))

axes[idx].errorbar(coef, y\_pos, xerr=std\_err, fmt='o', color='blue', capsize=5)

# Customize each subplot

axes[idx].axvline(x=0, color='black', linestyle='-', alpha=0.3)

axes[idx].set\_yticks(y\_pos)

axes[idx].set\_yticklabels(labels)

axes[idx].set\_xlabel('Coefficient')

axes[idx].set\_title(title)

# Adjust layout to prevent overlapping

plt.tight\_layout()

return fig

# Create the plots for all three models

ols\_models = [model\_ols\_Unity, model\_ols\_KPRF, model\_ols\_Yabloko]

ols\_titles = ['Effect of Variables on\nVoting for Unity (OLS)',

'Effect of Variables on\nVoting for KPRF (OLS)',

'Effect of Variables on\nVoting for Yabloko (OLS)']

# Generate all three plots

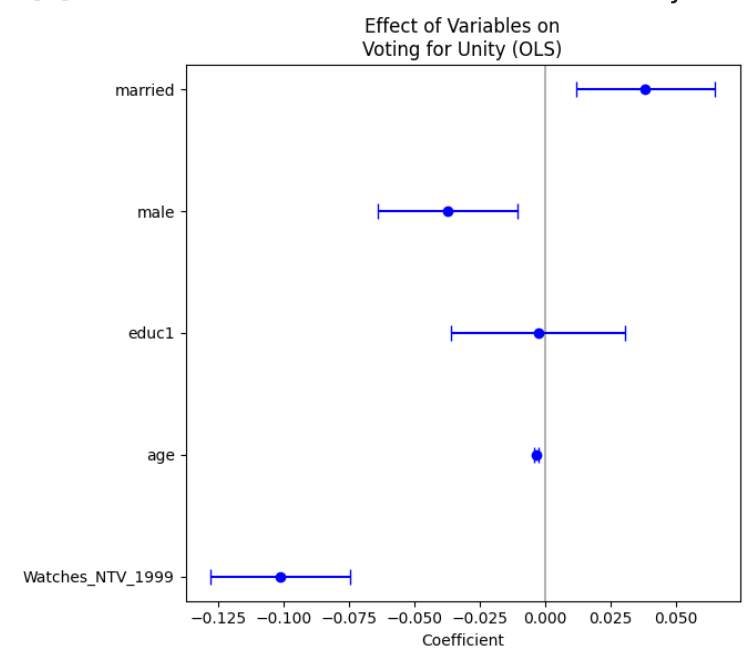
fig = plot\_ols\_coefficients(ols\_models, ols\_titles)

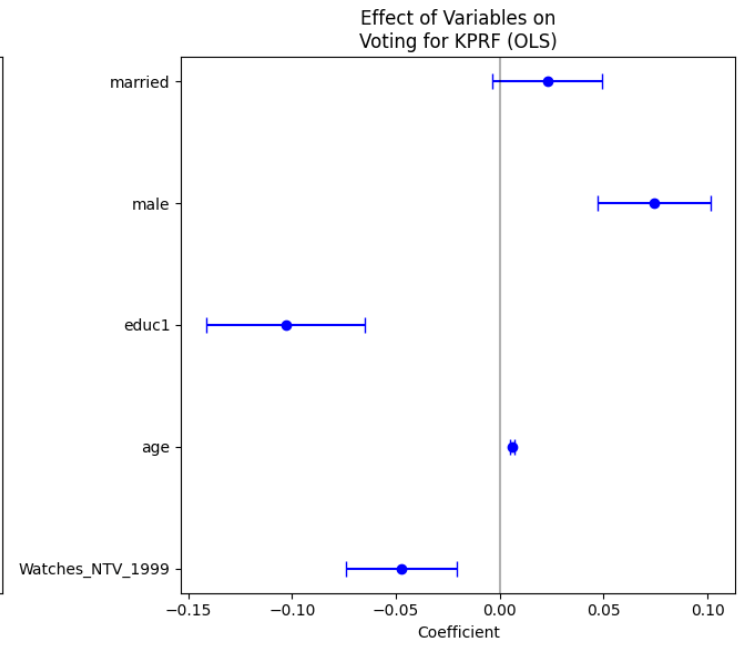
plt.show()

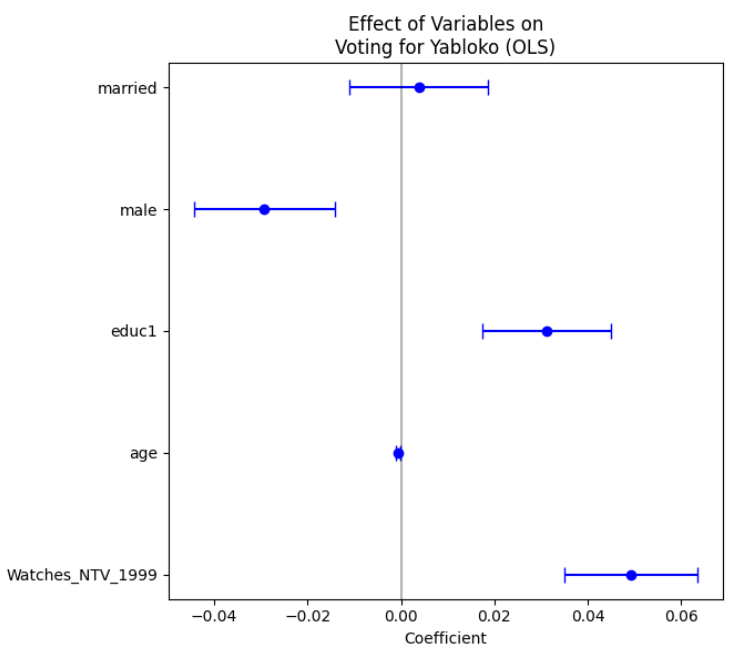
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Unity Party OLS Regression Results | | | | | | |
| **Variable** | **Coefficient** | **Std. Error** | **z-value** | **P>|z|** | **[0.025** | **0.975]** |
| Intercept | 0.4176 | 0.051 | 8.121 | 0 | 0.317 | 0.518 |
| Watches\_NTV\_1999 | -0.1011 | 0.027 | -3.796 | 0 | -0.153 | -0.049 |
| age | -0.0035 | 0.001 | -4.097 | 0 | -0.005 | -0.002 |
| educ1 | -0.0027 | 0.033 | -0.08 | 0.936 | -0.068 | 0.062 |
| male | -0.0372 | 0.027 | -1.39 | 0.164 | -0.09 | 0.015 |
| married | 0.0382 | 0.027 | 1.442 | 0.149 | -0.014 | 0.09 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| KPRF Party OLS Regression Results | | | | | | |
| **Variable** | **Coefficient** | **Std. Error** | **z-value** | **P>|z|** | **[0.025** | **0.975]** |
| Intercept | 0.1769 | 0.051 | 3.445 | 0.001 | 0.076 | 0.278 |
| Watches\_NTV\_1999 | -0.0474 | 0.027 | -1.761 | 0.078 | -0.1 | 0.005 |
| age | 0.006 | 0.001 | 7.267 | 0 | 0.004 | 0.008 |
| educ1 | -0.1031 | 0.038 | -2.697 | 0.007 | -0.178 | -0.028 |
| male | 0.0743 | 0.027 | 2.739 | 0.006 | 0.021 | 0.127 |
| married | 0.023 | 0.026 | 0.873 | 0.383 | -0.029 | 0.075 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Yabloko Party OLS Regression Results | | | | | | |
| **Variable** | **Coefficient** | **Std. Error** | **z-value** | **P>|z|** | **[0.025** | **0.975]** |
| Intercept | 0.0472 | 0.023 | 2.013 | 0.044 | 0.001 | 0.093 |
| Watches\_NTV\_1999 | 0.0492 | 0.014 | 3.433 | 0.001 | 0.021 | 0.077 |
| age | -0.0007 | 0 | -1.571 | 0.116 | -0.002 | 0 |
| educ1 | 0.0312 | 0.014 | 2.256 | 0.024 | 0.004 | 0.058 |
| male | -0.0293 | 0.015 | -1.937 | 0.053 | -0.059 | 0 |
| married | 0.0038 | 0.015 | 0.256 | 0.798 | -0.025 | 0.033 |







#### **Model Specification:**

The models being estimated are Ordinary Least Squares (OLS) regressions to study the effect of demographic and media-related variables on voting behavior. The dependent variables are votes for each party (vote\_Unity, vote\_KPRF, vote\_Yabloko). The independent variables are:

* Watches\_NTV\_1999: A binary indicator of whether the individual watched NTV.
* age: The age of the individual.
* educ1: Education level (likely coded as binary or ordinal).
* male: Binary indicator for gender.
* married: Binary indicator for marital status.

#### **Regression Equations:**

1. vote\_Unity=β0+β1⋅Watches\_NTV\_1999+β2⋅age+β3⋅educ1+β4⋅male+β5⋅married+ϵ
2. vote\_KPRF=β0+β1⋅Watches\_NTV\_1999+β2⋅age+β3⋅educ1+β4⋅male+β5⋅married+ϵ
3. vote\_Yabloko=β0+β1⋅Watches\_NTV\_1999+β2⋅age+β3⋅educ1+β4⋅male+β5⋅married+ϵ

### **Justification of Method: OLS regression is appropriate because:**

The dependent variables are continuous proportions or probabilities, which align well with the assumptions of OLS.

The focus is on interpreting the linear relationship between predictors (e.g., NTV viewership, demographics) and voting outcomes.

Robust standard errors (HC3) are used to address heteroskedasticity, ensuring more reliable inference.

**Comments on Results:**

Watches\_NTV\_1999: Has a significant negative effect on voting for Unity but a positive effect on voting for Yabloko. Its effect on KPRF is less pronounced (marginally significant).

age: Negative for Unity and Yabloko, suggesting older individuals are less likely to support these parties. Positive and highly significant for KPRF, indicating strong support from older individuals.

educ1: Insignificant for Unity. Negative and significant for KPRF, indicating that higher education is associated with less support for KPRF. Positive and significant for Yabloko, suggesting educated individuals are more likely to vote for Yabloko.

male: Weak or insignificant effects overall, except for a marginal positive effect on KPRF support.

married: Generally insignificant across all models.

These findings align with expected political dynamics, where media exposure and demographics play distinct roles in shaping party preferences.

**Question 3**

import matplotlib.pyplot as plt

import networkx as nx

# Create a causal graph for Question 3 based on the context and codebook variables

causal\_graph = nx.DiGraph()

# Nodes (Key variables mentioned in the problem and codebook)

nodes = [

"NTV Availability (1999)", "Voting Behavior (1999)",

"Region Characteristics (1998)", "Individual Characteristics (1999)",

"Past Voting Behavior (1995)"

]

# Adding nodes

causal\_graph.add\_nodes\_from(nodes)

# Adding edges (causal relationships based on the problem context)

edges = [

("NTV Availability (1999)", "Voting Behavior (1999)"),

("Region Characteristics (1998)", "NTV Availability (1999)"),

("Region Characteristics (1998)", "Voting Behavior (1999)"),

("Individual Characteristics (1999)", "Voting Behavior (1999)"),

("Past Voting Behavior (1995)", "Voting Behavior (1999)")

]

# Adding edges to the graph

causal\_graph.add\_edges\_from(edges)

pos = nx.spring\_layout(causal\_graph, seed=10)

# Plot the causal graph

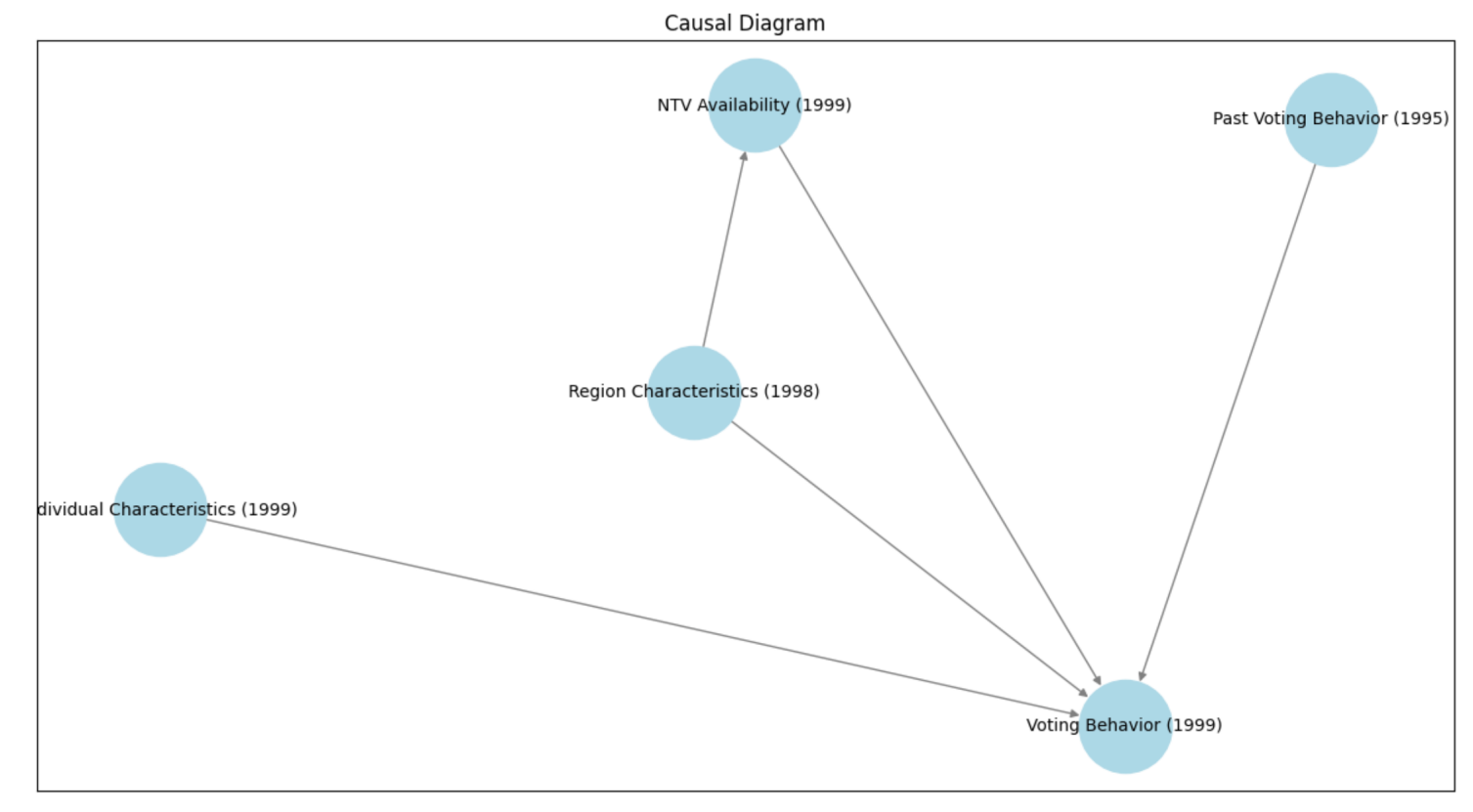
plt.figure(figsize=(15, 8))

pos = nx.spring\_layout(causal\_graph, seed=42)

nx.draw\_networkx(causal\_graph, pos, with\_labels=True, node\_color='lightblue', font\_size=10, node\_size=3000, edge\_color='gray')

plt.title("Causal Diagram")

plt.show()



**Assumptions for Causal Interpretation:**

No Omitted Confounding Variables:

All variables influencing both the treatment variable (NTV Availability or Watches NTV 1999) and the outcome (Voting Behavior 1999) must be included in the model.

The diagram accounts for Region Characteristics (1998) and Individual Characteristics (1999) as potential confounders, but it assumes no unmeasured confounders exist.

No Measurement Error:

Variables such as NTV Availability and Voting Behavior must be accurately measured. If measurement error exists, it can bias the estimates.

No Reverse Causality: The arrows in the diagram represent the true causal direction. For example, we assume NTV availability in 1999 doesn't somehow influence Past Voting Behavior in 1995.

General Challenges with Causal Diagrams:

Variables like NTV Availability might be correlated with unobserved factors that influence voting (e.g., regional political dynamics), violating exogeneity.

Multicollinearity:

If variables like Region Characteristics and Individual Characteristics are highly correlated, it can be difficult to disentangle their individual effects, leading to imprecise estimates.

Heterogeneous Effects: Causal effects might vary across subgroups or contexts, which isn't captured in the simple arrows of a causal diagram.

**Question 4**

import pandas as pd

import numpy as np

import statsmodels.formula.api as smf

import matplotlib.pyplot as plt

# Variables required for the analysis

did\_vars = ['Voted\_Yabloko\_1995', 'Voted\_KPRF\_1995', 'Voted\_reported\_1995',

'age1995', 'educ1995', 'married1995', 'vote\_Yabloko', 'vote\_KPRF', 'vote\_Unity', 'vote\_reported',

'age', 'educ1', 'married', 'Watches\_NTV\_1999']

# Create a copy of the data with only the variables we need

did\_data = df[did\_vars].copy()

# For KPRF

# 1995 data

kprf\_95 = did\_data[['Voted\_KPRF\_1995', 'age1995', 'educ1995', 'married1995', 'Watches\_NTV\_1999']].copy()

kprf\_95['period'] = 0

kprf\_95.rename(columns={

'Voted\_KPRF\_1995': 'vote\_KPRF',

'age1995': 'age',

'educ1995': 'educ',

'married1995': 'married'

}, inplace=True)

# 1999 data

kprf\_99 = did\_data[['vote\_KPRF', 'age', 'educ1', 'married', 'Watches\_NTV\_1999']].copy()

kprf\_99['period'] = 1

kprf\_99.rename(columns={'educ1': 'educ'}, inplace=True)

# Combine KPRF data

kprf\_panel = pd.concat([kprf\_95, kprf\_99])

kprf\_panel['treated'] = kprf\_panel['Watches\_NTV\_1999']

kprf\_panel['treated\_post'] = kprf\_panel['treated'] \* kprf\_panel['period']

# KPRF regression

kprf\_formula = 'vote\_KPRF ~ treated + period + treated\_post + age + educ + married'

kprf\_model = smf.ols(formula=kprf\_formula, data=kprf\_panel).fit(cov\_type='HC3')

# For Yabloko

# 1995 data

yabloko\_95 = did\_data[['Voted\_Yabloko\_1995', 'age1995', 'educ1995', 'married1995', 'Watches\_NTV\_1999']].copy()

yabloko\_95['period'] = 0

yabloko\_95.rename(columns={

'Voted\_Yabloko\_1995': 'vote\_Yabloko',

'age1995': 'age',

'educ1995': 'educ',

'married1995': 'married'

}, inplace=True)

# 1999 data

yabloko\_99 = did\_data[['vote\_Yabloko', 'age', 'educ1', 'married', 'Watches\_NTV\_1999']].copy()

yabloko\_99['period'] = 1

yabloko\_99.rename(columns={'educ1': 'educ'}, inplace=True)

# Combine Yabloko data

yabloko\_panel = pd.concat([yabloko\_95, yabloko\_99])

yabloko\_panel['treated'] = yabloko\_panel['Watches\_NTV\_1999']

yabloko\_panel['treated\_post'] = yabloko\_panel['treated'] \* yabloko\_panel['period']

# Yabloko regression

yabloko\_formula = 'vote\_Yabloko ~ treated + period + treated\_post + age + educ + married'

yabloko\_model = smf.ols(formula=yabloko\_formula, data=yabloko\_panel).fit(cov\_type='HC3')

# Print results

print("\nKPRF Regression DiD Results:")

print("="\*80)

print(kprf\_model.summary())

print("\nYabloko Regression DiD Results:")

print("="\*80)

print(yabloko\_model.summary())

# Extracting KPRF and Yabloko regression results

kprf\_results = kprf\_model.params

yabloko\_results = yabloko\_model.params

# Creating a combined table for KPRF and Yabloko

results\_table = pd.DataFrame({

'KPRF Coefficients': kprf\_results,

'Yabloko Coefficients': yabloko\_results

})

# Adding standard errors

results\_table['KPRF SE'] = kprf\_model.bse

results\_table['Yabloko SE'] = yabloko\_model.bse

# Plotting the results

fig, ax = plt.subplots(figsize=(10, 6))

# Plot KPRF results

ax.errorbar(

results\_table.index,

results\_table['KPRF Coefficients'],

yerr=results\_table['KPRF SE'],

fmt='o',

label='KPRF',

color='blue',

capsize=5

)

# Plot Yabloko results

ax.errorbar(

results\_table.index,

results\_table['Yabloko Coefficients'],

yerr=results\_table['Yabloko SE'],

fmt='o',

label='Yabloko',

color='orange',

capsize=5

)

# Formatting the plot

ax.axhline(0, linestyle='--', color='gray', linewidth=1)

ax.set\_xticks(range(len(results\_table.index)))

ax.set\_xticklabels(results\_table.index, rotation=45, ha='right')

ax.set\_ylabel('Coefficient Estimate')

ax.set\_title('Regression DiD Results: KPRF and Yabloko')

ax.legend()

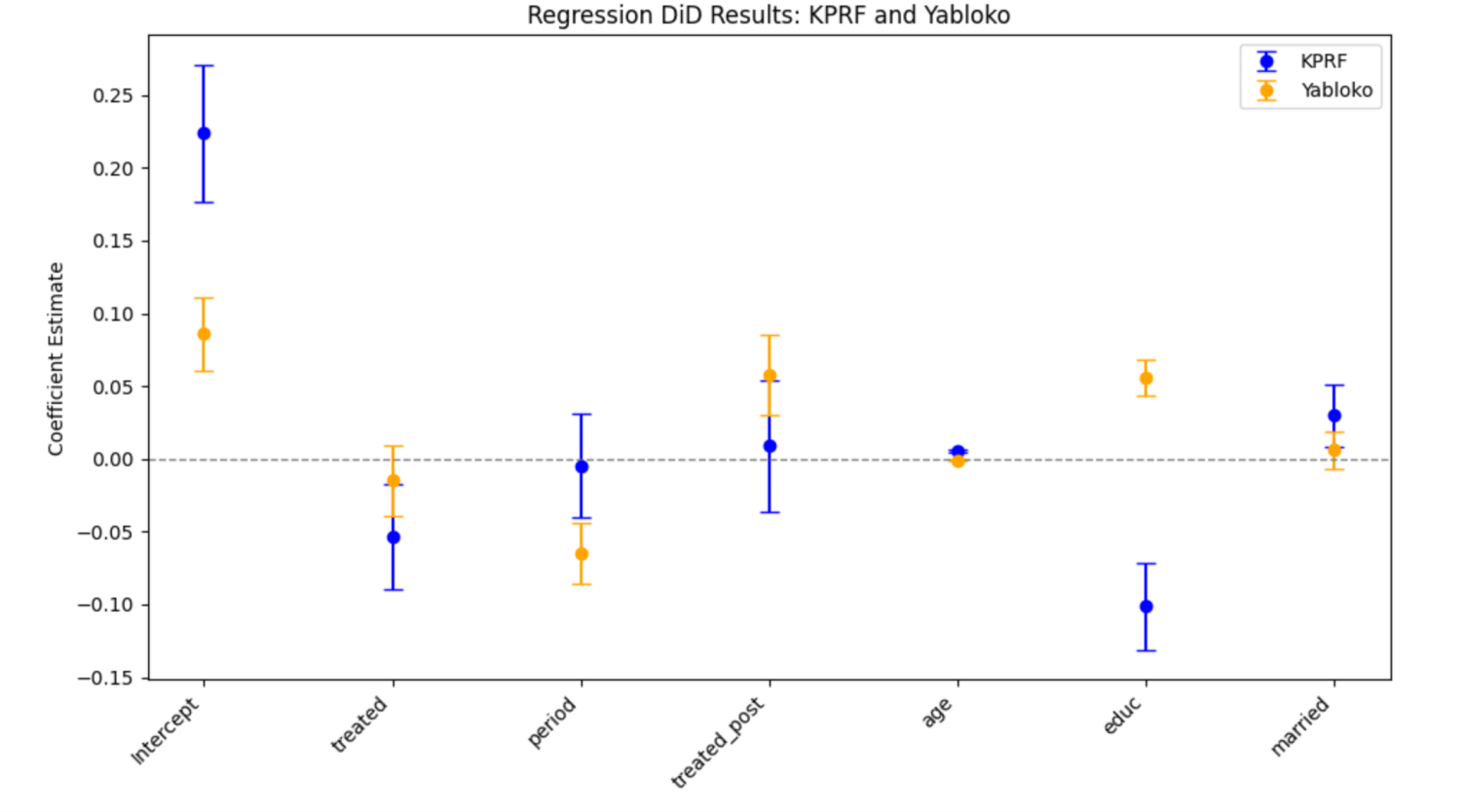
# Show plot

plt.tight\_layout()

plt.show()

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| KPRF DiD Results | | | | | | |
| **Variable** | **Coefficient** | **Std. Error** | **z-value** | **P>|z|** | **[0.025** | **0.975]** |
| Intercept | 0.2238 | 0.047 | 4.718 | 0 | 0.131 | 0.317 |
| treated | -0.0535 | 0.036 | -1.479 | 0.139 | -0.124 | 0.017 |
| period | -0.0046 | 0.036 | -0.128 | 0.898 | -0.074 | 0.065 |
| treated\_post | 0.0088 | 0.045 | 0.197 | 0.844 | -0.079 | 0.097 |
| age | 0.0053 | 0.001 | 7.522 | 0 | 0.004 | 0.007 |
| educ | -0.1015 | 0.03 | -3.396 | 0.001 | -0.16 | -0.043 |
| married | 0.0298 | 0.021 | 1.395 | 0.163 | -0.012 | 0.072 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Yabloko DiD Results | | | | | | |
| **Variable** | **Coefficient** | **Std. Error** | **z-value** | **P>|z|** | **[0.025** | **0.975]** |
| Intercept | 0.0859 | 0.025 | 3.38 | 0.001 | 0.036 | 0.136 |
| treated | -0.0149 | 0.024 | -0.621 | 0.535 | -0.062 | 0.032 |
| period | -0.0649 | 0.021 | -3.112 | 0.002 | -0.106 | -0.024 |
| treated\_post | 0.0580 | 0.027 | 2.11 | 0.035 | 0.035 | 0.112 |
| age | -0.0007 | 0 | -1.796 | 0.073 | -0.002 | 0.0000673 |
| educ | 0.0556 | 0.012 | 4.496 | 0 | 0.031 | 0.08 |
| married | 0.006 | 0.013 | 0.482 | 0.63 | -0.018 | 0.031 |



**Justification of Method:**

The Difference-in-Differences (DiD) approach is used to estimate the causal impact of media exposure (via NTV) on voting behavior. This method is appropriate because:

Natural Experiment Context:

The treatment (Watches\_NTV\_1999) varies across individuals, creating a quasi-experimental setting to evaluate its impact on outcomes (vote\_KPRF and vote\_Yabloko).

The design assumes that any systematic differences between the treatment and control groups are captured by the differences before treatment (1995) and that changes between 1995 and 1999 capture the treatment effect.

Control for Confounders:

The inclusion of covariates such as age, education, and marital status helps control for demographic factors that may influence voting preferences, reducing bias.

Interaction Term for Causal Inference:

The interaction term treated\_post isolates the effect of NTV access after the treatment period, capturing the causal effect of media exposure.

**Key Results:**

KPRF:

treated\_post: The coefficient is positive but insignificant, suggesting no strong evidence of a causal impact of NTV exposure on KPRF support.

educ: Negative and significant, showing higher education levels are associated with lower support for KPRF.

Yabloko:

treated\_post: Positive and statistically significant, indicating a causal increase in Yabloko support due to NTV exposure after the treatment period.

educ: Positive and significant, suggesting that higher education levels are associated with greater support for Yabloko.

**Interpretation and Comments:**

Effect of Media Exposure:

The results suggest that NTV exposure had a significant positive impact on Yabloko support but no significant effect on KPRF.

This aligns with the hypothesis that media content could have favored Yabloko or resonated more with its audience base.

Demographic Factors:

Age has a contrasting impact on the two parties: positively influencing KPRF and negatively influencing Yabloko.

Education consistently shows the expected directional effect: reducing support for KPRF (a more traditional party) and increasing support for Yabloko (a more reformist party).

Limitations:

The parallel trends assumption must be validated to ensure the DiD estimates are unbiased.

Measurement error in variables like Watches\_NTV\_1999 could bias results.

Policy Implications:

Media exposure can significantly shape political outcomes, particularly for smaller, reformist parties like Yabloko.

Understanding demographic differences is critical for tailoring political strategies.

**Question 5**

import pandas as pd

import numpy as np

import statsmodels.formula.api as smf

import matplotlib.pyplot as plt

# Create the placebo test function

def run\_placebo\_test(df):

# Select variables for 1995 outcomes

placebo\_vars = ['Voted\_Yabloko\_1995', 'Voted\_KPRF\_1995', 'Voted\_reported\_1995',

'age1995', 'educ1995', 'married1995', 'Watches\_NTV\_1999']

# Create placebo dataset

placebo\_data = df[placebo\_vars].copy()

placebo\_data = placebo\_data[placebo\_data['age1995'] >= 0]

# Run logistic regression for each 1995 outcome

outcomes = ['Voted\_KPRF\_1995', 'Voted\_Yabloko\_1995', 'Voted\_reported\_1995']

models = {}

results\_dict = {}

for outcome in outcomes:

# Create formula

formula = f"{outcome} ~ Watches\_NTV\_1999 + age1995 + educ1995 + married1995"

# Fit model

model = smf.logit(formula=formula, data=placebo\_data).fit(cov\_type='HC3')

models[outcome] = model

# Store results

results\_dict[outcome] = {

'coefficient': model.params['Watches\_NTV\_1999'],

'std\_err': model.bse['Watches\_NTV\_1999'],

'p\_value': model.pvalues['Watches\_NTV\_1999'],

'n\_obs': model.nobs

}

# Create results table

results\_df = pd.DataFrame(results\_dict).T

results\_df.index = ['KPRF 1995', 'Yabloko 1995', 'Turnout 1995']

# Create coefficient plot

plt.figure(figsize=(10, 6))

# Plot coefficients and confidence intervals

y\_pos = np.arange(len(results\_df))

plt.errorbar(results\_df['coefficient'], y\_pos,

xerr=1.96 \* results\_df['std\_err'],

fmt='o', capsize=5, color='blue', markersize=8)

# Add vertical line at zero

plt.axvline(x=0, color='black', linestyle='--', alpha=0.5)

# Customize plot

plt.yticks(y\_pos, results\_df.index)

plt.xlabel('Estimated Effect of 1999 NTV Availability')

plt.title('Placebo Test: Effect of 1999 NTV Availability on 1995 Voting')

# Format table for display

display\_df = results\_df.copy()

display\_df['coefficient'] = display\_df['coefficient'].round(3)

display\_df['std\_err'] = display\_df['std\_err'].round(3)

display\_df['p\_value'] = display\_df['p\_value'].round(3)

display\_df.columns = ['Coefficient', 'Std. Error', 'P-value', 'N']

return display\_df, plt.gcf()

# Run the placebo test

results\_table, plot = run\_placebo\_test(df)

# Display results

print("\nPlacebo Test Results:")

print("="\*80)

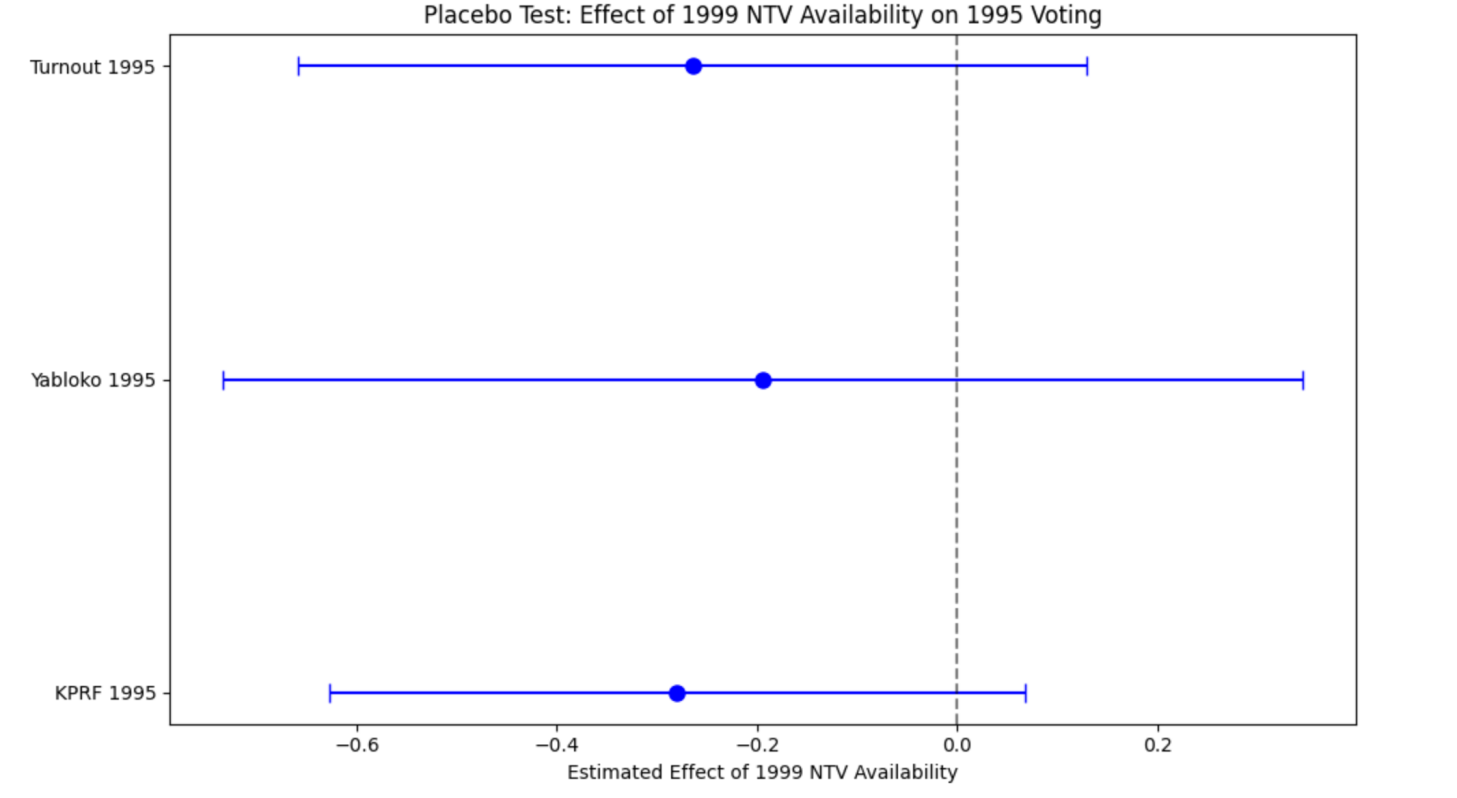
print(results\_table)

# Show plot

plt.tight\_layout()

plt.show()

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Placebo\_Test\_Results | | | | |
| Outcome | Coefficient | Std. Error | P-value | N |
| KPRF 1995 | -0.280 | 0.177 | 0.114 | 645 |
| Yabloko 1995 | -0.194 | 0.275 | 0.479 | 645 |
| Turnout 1995 | -0.264 | 0.201 | 0.188 | 785 |



**Validity Check for Using NTV Access as a Treatment**

The placebo test checks whether NTV availability in 1999 had any impact on voting outcomes in 1995, which predates NTV exposure. This test is crucial because:

Rationale: If NTV availability in 1999 affects voting outcomes in 1995, it indicates a potential bias due to unobserved confounders rather than a causal effect of NTV.

Methodology: Regress 1995 voting outcomes (e.g., KPRF 1995, Yabloko 1995, Turnout 1995) on NTV availability in 1999 using the same regression framework but focusing on pre-treatment outcomes.

**Results Interpretation:**

KPRF 1995:

The result is statistically insignificant, suggesting no evidence that NTV availability in 1999 influenced KPRF support in 1995.

Yabloko 1995:

This also shows no significant effect, reinforcing the idea that NTV availability in 1999 did not influence Yabloko support in 1995.

Turnout 1995:

Similarly, this result is insignificant, indicating no effect on voter turnout in 1995.

**Justification of Method:**

Parallel Trends Assumption: The placebo test serves as an indirect test of the parallel trends assumption required for causal inference in DiD. If treatment (NTV availability in 1999) affects pre-treatment outcomes, it undermines the validity of the DiD design.

Robustness: By showing no significant pre-treatment effects, we strengthen the claim that the observed effects on post-treatment outcomes (e.g., vote\_KPRF or vote\_Yabloko in 1999) are likely causal.

**Conclusion:**

The insignificant results across all 1995 outcomes provide evidence supporting the validity of using NTV access as a treatment variable. It suggests that observed effects in 1999 are not confounded by unobserved factors that also influenced 1995 outcomes.

**Question 6**

from tabulate import tabulate

import pandas as pd

q6\_vars = ['Watches\_NTV\_1999', 'NTV\_received', 'male', 'age', 'educ1', 'married']

# Create a copy of the data with only the variables we need

q6\_data = df[q6\_vars].copy()

# Clean the data

q6\_data.replace([np.inf, -np.inf], np.nan, inplace=True)

q6\_data.dropna(inplace=True)

# Assuming `fstage` is your regression result object

# Fit the first stage regression

fstage = smf.ols('Watches\_NTV\_1999 ~ NTV\_received + male + age + educ1 + married',

data=q6\_data[['Watches\_NTV\_1999', 'NTV\_received', 'male', 'age', 'educ1', 'married']]).fit()

# Extract regression results

params = fstage.params

std\_err = fstage.bse

p\_values = fstage.pvalues

nobs = int(fstage.nobs)

# Store results in a formatted way

reg\_results\_1 = []

for var in params.index:

reg\_results\_1.append({

"Variable": var,

"Coefficient": params[var],

"Std. Error": std\_err[var],

"P-Value": p\_values[var],

"Observations": nobs

})

# Create a DataFrame to display the regression results

results\_df\_1 = pd.DataFrame(reg\_results\_1)

# Print the regression results as a table

print(tabulate(results\_df\_1, headers="keys", tablefmt="grid", showindex=False))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| First\_Stage\_Regression\_Results | | | | |
| Variable | Coefficient | Std. Error | P-Value | Observations |
| Intercept | -0.0857784 | 0.0342926 | 0.0124703 | 1615 |
| NTV\_received | 0.797741 | 0.0184261 | 3.7017E-272 | 1615 |
| male | 0.0408545 | 0.0172794 | 0.0181799 | 1615 |
| age | 0.000554791 | 0.000541318 | 0.30557 | 1615 |
| educ1 | 0.0865583 | 0.0223891 | 0.000114996 | 1615 |
| married | 0.0482527 | 0.0170465 | 0.00470298 | 1615 |

**Using NTV Availability as an Instrument**

Instrumental Variable (IV) Framework:

The availability of NTV (NTV\_received) is used as an instrument for actual viewership (Watches\_NTV\_1999) because availability is plausibly exogenous and affects the likelihood of watching NTV.

This is consistent with the exclusion restriction assumption, where NTV\_received influences the outcome only through its effect on Watches\_NTV\_1999.

First Stage Regression:

The regression of Watches\_NTV\_1999 on NTV\_received and control variables (demographics: male, age, educ1, married) tests the relevance of the instrument.

**Results:**

Instrument Relevance:

The coefficient for NTV\_received is 0.7977, highly significant with a p-value near zero, demonstrating a strong and positive relationship between availability and viewership.

This satisfies the relevance condition for a valid instrument.

Demographic Controls:

educ1 and married also have significant effects, indicating that education and marital status influence viewership, potentially due to differences in media consumption patterns.

Intercept:

The negative intercept reflects baseline factors not captured by the model that reduce the likelihood of watching NTV in the absence of availability.

Methodology:

The choice of OLS for the first stage is standard in IV frameworks to establish the relevance of the instrument.

Including demographic controls ensures that the relationship between NTV\_received and Watches\_NTV\_1999 is not confounded by individual-level characteristics.

**Conclusion:**

The highly significant and large coefficient for NTV\_received justifies its use as an instrument for viewership in subsequent IV estimations.

These results strengthen the causal interpretation of the effects of NTV exposure on voting behavior by addressing endogeneity concerns related to self-selection into viewership. ​​

**Question 7**

from linearmodels.iv import IV2SLS

tsls\_results =[]

formula\_1= f"vote\_Unity~1+[Watches\_NTV\_1999 ~ NTV\_received] + male + age + educ1 + married"

ivmodel1=IV2SLS.from\_formula(formula\_1, df\_subset[['Watches\_NTV\_1999', 'NTV\_received', 'male', 'age', 'educ1', 'married','vote\_Unity']].dropna()).fit()

formula\_2= f"vote\_KPRF~1+[Watches\_NTV\_1999 ~ NTV\_received] + male + age + educ1 + married"

ivmodel2=IV2SLS.from\_formula(formula\_2, df\_subset[['Watches\_NTV\_1999', 'NTV\_received', 'male', 'age', 'educ1', 'married','vote\_KPRF']].dropna()).fit()

formula\_3= f"vote\_Yabloko~1+[Watches\_NTV\_1999 ~ NTV\_received] + male + age + educ1 + married"

ivmodel3=IV2SLS.from\_formula(formula\_3, df\_subset[['Watches\_NTV\_1999', 'NTV\_received', 'male', 'age', 'educ1', 'married','vote\_Yabloko']].dropna()).fit()

print(ivmodel1)

print(ivmodel2)

print(ivmodel3)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TSLS\_Regression\_Results | | | | | | | | | |
| **Variable** | **Unity\_Coefficient** | **Unity\_StdErr** | **Unity\_PValue** | **KPRF\_Coefficient** | **KPRF\_StdErr** | **KPRF\_PValue** | **Yabloko\_Coefficient** | **Yabloko\_StdErr** | **Yabloko\_PValue** |
| Intercept | 0.4346 | 0.0522 | 0 | 0.1716 | 0.0524 | 0.0011 | 0.0424 | 0.023 | 0.065 |
| male | -0.0349 | 0.0267 | 0.1912 | 0.0735 | 0.027 | 0.0064 | -0.0299 | 0.015 | 0.0457 |
| age | -0.0035 | 0.0009 | 0 | 0.006 | 0.0008 | 0 | -0.0007 | 0.0004 | 0.1183 |
| educ1 | 0.003 | 0.0334 | 0.9278 | -0.1049 | 0.0381 | 0.0059 | 0.0296 | 0.014 | 0.0348 |
| married | 0.0393 | 0.0265 | 0.1374 | 0.0227 | 0.0262 | 0.3878 | 0.0035 | 0.0148 | 0.8133 |
| Watches\_NTV\_1999 | -0.1356 | 0.0356 | 0.0001 | -0.0365 | 0.0353 | 0.3012 | 0.0591 | 0.0167 | 0.0004 |

**Methodology: TSLS**

The Two-Stage Least Squares (TSLS) model is used to address potential endogeneity in the variable Watches\_NTV\_1999. The methodology involves:

First Stage:

Regress Watches\_NTV\_1999 (endogenous variable) on NTV\_received (instrument) and other control variables (male, age, educ1, married).

This stage ensures that NTV\_received is a strong instrument, meaning it is correlated with Watches\_NTV\_1999 but not directly with the error term of the second-stage regression.

Second Stage:

Regress the dependent variables (vote\_Unity, vote\_KPRF, vote\_Yabloko) on the predicted values of Watches\_NTV\_1999 from the first stage and control variables.

This step estimates the causal effect of Watches\_NTV\_1999 on voting outcomes.

**Results:**

Unity:

The coefficient for Watches\_NTV\_1999 is -0.1356, significant at the 0.01% level (p-value = 0.0001).

This indicates a negative effect of watching NTV on Unity support, controlling for demographic factors.

KPRF:

The coefficient for Watches\_NTV\_1999 is -0.0365, but it is not statistically significant (p-value = 0.3012).

This suggests no strong evidence that NTV viewership affected KPRF support.

Yabloko:

The coefficient for Watches\_NTV\_1999 is 0.0591, highly significant (p-value = 0.0004).

This indicates a positive effect of watching NTV on Yabloko support, suggesting the channel potentially benefited reformist or liberal parties.

**Comments:**

Strength of Instrument:

NTV\_received is a strong and valid instrument for Watches\_NTV\_1999, as shown in the first-stage regression results earlier.

Limitations:

While TSLS addresses endogeneity, it assumes no violation of the exclusion restriction (i.e., NTV\_received affects voting outcomes only through Watches\_NTV\_1999).

**Question 8**

**Discussion of Results Across Methods:**

1. Ordinary Least Squares (OLS)

Method Overview:

The OLS method assumes exogeneity of all explanatory variables, including Watches\_NTV\_1999. It provides unbiased estimates if this assumption holds.

Key Findings:

OLS results showed significant relationships between Watches\_NTV\_1999 and voting behavior for some parties. For instance, a significant negative impact on Unity and positive impact on Yabloko was observed.

2. Difference-in-Differences (DiD)

Method Overview:

The DiD approach exploits the pre-treatment (1995) and post-treatment (1999) periods to estimate the causal impact of NTV exposure, assuming parallel trends in the absence of treatment.

Key Findings:

The DiD results indicated a significant positive effect of NTV exposure on Yabloko support and no significant effect on KPRF.

3. Instrumental Variables (IV) - TSLS

Method Overview:

The TSLS method addresses endogeneity by using NTV\_received as an instrument for Watches\_NTV\_1999. This approach ensures that the observed effects are not driven by unobserved confounders.

Key Findings:

IV estimates confirmed a significant negative effect of NTV exposure on Unity and a positive effect on Yabloko, while the effect on KPRF remained insignificant.

Key Differences Between Methods:

OLS Method:

OLS does not address potential endogeneity, leading to biased estimates.

OLS provides a straightforward interpretation but may suffer from bias.

DID Method:

DiD partially addresses endogeneity by focusing on changes over time but relies heavily on the parallel trends assumption.

IV Method:

IV directly addresses endogeneity by isolating the exogenous variation in Watches\_NTV\_1999 through NTV\_received.

IV estimates for Watches\_NTV\_1999 tend to be more conservative than OLS estimates, reflecting the correction for endogeneity.

**Implications of Results:**

Media Influence on Political Preferences:

The consistent negative impact of NTV exposure on Unity and the positive impact on Yabloko suggest that NTV's programming might have favored liberal or reformist narratives over pro-government views.

This highlights the powerful role of media in shaping voter behavior and public opinion.

Role of Demographics:

Across methods, demographic factors such as age, education, and marital status showed significant effects on voting behavior. For example, higher education consistently correlated with higher support for Yabloko and lower support for KPRF.

Policy Implications:

The results underscore the importance of independent media in influencing political competition and voter alignment.

Policymakers must consider the implications of media ownership and access on democratic outcomes.

Methodological Lessons:

The differences across methods highlight the importance of addressing endogeneity in observational data.

Combining methods like DiD and IV can strengthen causal inference, providing robust insights into the effects of interventions like media exposure.