Lab Assignment 12: Interactive Visualizations

DS 6001: Practice and Application of Data Science

H. Diana McSpadden (hdm5s)

Instructions

Please answer the following questions as completely as possible using text, code, and the results of code as needed. Format your answers in a Jupyter notebook. To receive full credit, make sure you address every part of the problem, and make sure your document is formatted in a clean and professional way.

Problem 0

Import the following libraries:

In [ ]:
```
import numpy as np
import pandas as pd
import plotly.graph_objects as go
import plotly.express as px
import plotly.graph_objects as go
import plotly.figure_factory as ff
import dash
from jupyter_dash import JupyterDash
#import dash_core_components as dcc
from dash import dcc
from dash import html
#import dash_core_components as dcc
#import dash_html_components as html
from dash.dependencies import Input, Output
external_stylesheets = ['https://codepen.io/chriddyp/pen/bWLwgP.css']
```

For this lab, we will be working with the 2019 General Social Survey one last time.

%%capture

In [ ]:
```
gss = pd.read_csv("https://github.com/jkropko/DS-6001/raw/master/localdata/gss2018.
                  encoding='cp1252', na_values=['IAP','IAP,DK,NA,uncodeable', 'NOT S
                                                'DK', 'IAP, DK, NA, uncodeable', '.a
```

```
C:\Users\dianam\AppData\Local\Temp\ipykernel_24412\2595273594.py:1: DtypeWarning:

Columns (23,41,45,46,47,73,91,99,197,263,265,273,351,403,466,471,472,473,474,592,7
28,730,738,745,747,755,757,759,767,781,783,790,791,885,950,986,990,991,992) have m
ixed types. Specify dtype option on import or set low_memory=False.
```

Here is code that cleans the data and gets it ready to be used for data visualizations:

In [ ]:
```
mycols = ['id', 'wtss', 'sex', 'educ', 'region', 'age', 'coninc',
          'prestg10', 'mapres10', 'papres10', 'sei10', 'satjob',
          'fechld', 'fefam', 'fepol', 'fepresch', 'meovrwrk']
gss_clean = gss[mycols]
gss_clean = gss_clean.rename({'wtss':'weight',
                              'educ':'education',
                              'coninc':'income',
                              'prestg10':'job_prestige',
                              'mapres10':'mother_job_prestige',
                              'papres10':'father_job_prestige',
                              'sei10':'socioeconomic_index',
                              'fechld':'relationship',
                              'fefam':'male_breadwinner',
                              'fehire':'hire_women',
                              'fejobaff':'preference_hire_women',
                              'fepol':'men_bettersuited',
                              'fepresch':'child_suffer',
                              'meovrwrk':'men_overwork'},axis=1)
gss_clean.age = gss_clean.age.replace({'89 or older':'89'})
gss_clean.age = gss_clean.age.astype('float')
```

The `gss_clean` dataframe now contains the following features:

- `id` - a numeric unique ID for each person who responded to the survey
- `weight` - survey sample weights
- `sex` - male or female
- `education` - years of formal education
- `region` - region of the country where the respondent lives
- `age` - age
- `income` - the respondent's personal annual income
- `job_prestige` - the respondent's occupational prestige score, as measured by the GSS using the methodology described above
- `mother_job_prestige` - the respondent's mother's occupational prestige score, as measured by the GSS using the methodology described above
- `father_job_prestige` -the respondent's father's occupational prestige score, as measured by the GSS using the methodology described above
- `socioeconomic_index` - an index measuring the respondent's socioeconomic status
- `satjob` - responses to "On the whole, how satisfied are you with the work you do?"
- `relationship` - agree or disagree with: "A working mother can establish just as warm and secure a relationship with her children as a mother who does not work."

- `male_breadwinner` - agree or disagree with: "It is much better for everyone involved if the man is the achiever outside the home and the woman takes care of the home and family."
- `men_bettersuited` - agree or disagree with: "Most men are better suited emotionally for politics than are most women."
- `child_suffer` - agree or disagree with: "A preschool child is likely to suffer if his or her mother works."
- `men_overwork` - agree or disagree with: "Family life often suffers because men concentrate too much on their work."

Problem 1

Our goal in this lab is to build a dashboard that presents our findings from the GSS. A dashboard is meant to be shared with an audience, whether that audience is a manager, a client, a potential employer, or the general public. So we need to provide context for our results. One way to provide context is to write text using markdown code.

Find one or two websites that discuss the gender wage gap, and write a short paragraph in markdown code summarizing what these sources tell us. Include hyperlinks to these websites. Then write another short paragraph describing what the GSS is, what the data contain, how it was collected, and/or other information that you think your audience ought to know. A good starting point for information about the GSS is here: http://www.gss.norc.org/About-The-GSS

Then save the text as a Python string so that you can use the markdown code in your dashboard later.

It should go without saying, but no plagiarization! If you summarize a website, make sure you put the summary in your own words. Anything that is copied and pasted from the GSS webpage, Wikipedia, or another website without attribution will receive no credit.

(Don't spend too much time on this, and you might want to skip it during the Zoom session and return to it later so that you can focus on working on code with your classmates.) [1 point]

Here I am just seeing how the markdown will look:

Examining the gender pay gap in the United States attempts to discern the difference in pay between men and women that is a direct effect of being a women, rather than the effect of job selection, schedule selection, or other contributions. Meara, K., Pastore, F. & Webster, A.'s 'The gender pay gap in the USA: a matching study' from 2020's Journal of Population Economics attempts to use a "matching estimator" to control for the effects of job selection, parenthood, schedule selection, and other contributions. Their work finds that there are interaction effects between gender and other variables such as part-time work. England, P.,

Levine, A., & Mishel, E.'s 2020 article 'Progress toward gender equality in the United States has slowed or stalled' further describes how, even with the reversed gap in higher education attainment by women, the gender wage gap has not closed, and progress has stalled since 2018.

The data used in this dashboard is from the General Social Survey (GSS), a nationally representative survey of adults from the National Opinion Research Center (NORC) at the University of Chicago. For upto 80 years, questions have been asked of representative samples of the US population. Included in the GSS are topics related to psychological well-being, social mobility, and, survery questions related to gender, education, income, and job prestige: the topics of this dashboard.

In [ ]:
```
markdown_text = '''
Examining the gender pay gap in the United States attempts to discern the differenc
schedule selection, or other contributions. Meara, K., Pastore, F. & Webster, A.'s
attempts to use a "matching estimator" to control for the effects of job selection,
Their work finds that there are interaction effects between gender and other variab
even with the reversed gap in higher education attainment by women, the gender wage


The data used in this dashboard is from the [General Social Survey](https://gssdata
For upto 80 years, questions have been asked of representative samples of the US po
'''
```

Problem 2

Generate a table that shows the mean income, occupational prestige, socioeconomic index, and years of education for men and for women. Use a function from a `plotly` module to display a web-enabled version of this table. This table is for presentation purposes, so round every column to two decimal places and use more presentable column names. [3 points]

In [ ]:
```
gss_display = gss_clean.groupby('sex').agg({'income':'mean',
                                            'job_prestige':'mean',
                                            'socioeconomic_index':'mean',
                                            'education':'mean'})
gss_display = gss_display.rename({'income':'Income',
                                  'job_prestige':'Job Prestige',
                                  'socioeconomic_index':'SES Index',
                                  'education':'Yrs Education'}, axis=1)
gss_display = round(gss_display, 2)
gss_display = gss_display.reset_index().rename({'sex':'Sex'}, axis=1)
# Income column is a currency column, so we need to format it as such
gss_display['Income'] = gss_display['Income'].map('${:,.2f}'.format)

gss_display
```

Out[ ]:

	Sex	Income	Job Prestige	SES Index	Yrs Education
0	female	$47,191.02	44.67	46.58	13.76
1	male	$53,314.63	44.70	47.38	13.69

In [ ]:
```
table_p2 = ff.create_table(gss_display, height_constant=15)
table_p2.show()
```

Problem 3

Create an interactive barplot that shows the number of men and women who respond with each level of agreement to `male_breadwinner`. Write presentable labels for the x and y-axes, but don't bother with a title because we will be using a subtitle on the dashboard for this graphic. [3 points]

In [ ]:
```
m_breadwinner = pd.crosstab(gss_clean.sex, gss_clean.male_breadwinner).reset_index(
m_breadwinner = pd.melt(m_breadwinner, id_vars = 'sex', value_vars = ['agree', 'dis
m_breadwinner = m_breadwinner.rename({'value':'count'}, axis=1)
m_breadwinner
```

Out[ ]:

	sex	male_breadwinner	count
0	female	agree	152
1	male	agree	158
2	female	disagree	377
3	male	disagree	337
4	female	strongly agree	48
5	male	strongly agree	40
6	female	strongly disagree	286
7	male	strongly disagree	147

In [ ]:
```
# in the m_breadwinner dataframe make the male_breadwinner column into a categorica
m_breadwinner['male_breadwinner'] = pd.Categorical(m_breadwinner['male_breadwinner'
                      categories=["strongly disagree","disagree","agree","strongly
                      ordered=True)
```

In [ ]:
```
fig_p3 = px.bar(m_breadwinner, x="male_breadwinner", y="count",
             color='sex', barmode='group',
             labels={'male_breadwinner':'It is much better for everyone involved if
             width=900, height=300, )
# set the font size of the x and y axis labels
fig_p3.update_layout(xaxis_title_font_size=12, yaxis_title_font_size=10)
# order the categories in the x-axis
fig_p3.update_xaxes(categoryorder='array', categoryarray=['strongly disagree','disa
# set the x label
```

```
fig_p3.show()
```

Problem 4

Create an interactive scatterplot with `job_prestige` on the x-axis and `income` on the y-axis. Color code the points by `sex` and make sure that the figure includes a legend for these colors. Also include two best-fit lines, one for men and one for women. Finally, include hover data that shows us the values of `education` and `socioeconomic_index` for any point the mouse hovers over. Write presentable labels for the x and y-axes, but don't bother with a title because we will be using a subtitle on the dashboard for this graphic. [3 points]

In [ ]:
```
fig_p4 = px.scatter(gss_clean, x='job_prestige', y='income',
                    color='sex', trendline='lowess',
                    hover_data=['education', 'socioeconomic_index'],
                    labels={'job_prestige':'Job Prestige', 'income':'Income'},
                    width=400, height=400)
fig_p4.show()
```

Problem 5

Create two interactive box plots: one that shows the distribution of `income` for men and for women, and one that shows the distribution of `job_prestige` for men and for women. Write presentable labels for the axis that contains `income` or `job_prestige` and remove the label for `sex`. Also, turn off the legend. Don't bother with titles because we will be using subtitles on the dashboard for these graphics. [3 points]

In [ ]:
```
box_income = px.box(gss_clean, x='sex', y='income',
                    labels={'sex':'', 'income':'Income'}, color='sex', width=400, h
box_income.update_layout(showlegend=False)

box_jobprestige = px.box(gss_clean, x='sex', y='job_prestige',
                         labels={'sex':'', 'job_prestige':'Job Prestige'}, color='s
box_jobprestige.update_layout(showlegend=False)

box_income.show()
box_jobprestige.show()
```

Problem 6

Create a new dataframe that contains only `income`, `sex`, and `job_prestige`. Then create a new feature in this dataframe that breaks `job_prestige` into six categories with equally sized ranges. Finally, drop all rows with any missing values in this dataframe.

Then create a facet grid with three rows and two columns in which each cell contains an interactive box plot comparing the income distributions of men and women for each of these new categories.

(If you want men to be represented by blue and women by red, you can include `color_discrete_map = {'male':'blue', 'female':'red'}` in your plotting function. Or use different colors if you want!) [3 points]

In [ ]:
```
print(gss_clean[['job_prestige']].min())
(gss_clean[['job_prestige']].max() - gss_clean[['job_prestige']].min()) / 6
```

```
job_prestige    16.0
dtype: float64
```

Out[ ]:
```
job_prestige    10.666667
dtype: float64
```

In [ ]:
```
df_p6 = gss_clean[['income','sex','job_prestige']]
# break job prestige into 6 categories  with the cuts at 16, 27, 38, 49, 60, 71, 82
#df_p6['jp_cat'] = pd.cut(df_p6['job_prestige'], bins=[15, 27, 38, 49, 60, 71, 82],
df_p6['jp_cat'] = pd.cut(df_p6['job_prestige'], 6, labels=['1','2','3','4','5','6']
# drop rows with missing values
df_p6 = df_p6.dropna()

# I am leaving the default colors because I think people should be willing to think
```

In [ ]:
```
# make jp_cat columns categorical and ordered categorical
df_p6['jp_cat'] = pd.Categorical(df_p6['jp_cat'], ordered=True, categories=['1','2'
```

In [ ]:
```
df_p6.head()
```

Out[ ]:

	income	sex	job_prestige	jp_cat
1	22782.5000	female	22.0	1
2	112160.0000	male	61.0	5
3	158201.8412	female	59.0	5
4	158201.8412	male	53.0	4
6	13143.7500	female	48.0	3

In [ ]:
```
# sort df_p6 by jp_cat
df_p6 = df_p6.sort_values(by='jp_cat')
```

I am leaving the default colors because I think people should be willing to think about men with violet and women with red.

In [ ]:
```
fig_box = px.box(df_p6, x='sex', y='income', color='sex',
             facet_col='jp_cat', facet_col_wrap=2,
            labels={'sex':'', 'income':'Income', 'jp_cat':'Job Prestige Level'},
            width=800, height=600)
```

```
fig_box.update(layout=dict(title=dict(x=0.5)))
#fig_box.update_layout(showlegend=False)
#fig_box.for_each_annotation(lambda a: a.update(text=a.text.replace("vote=", "")))
fig_box.show()
```

Problem 7

Create a dashboard that displays the following elements:

- A descriptive title
- The markdown text you wrote in problem 1
- The table you made in problem 2
- The barplot you made in problem 3
- The scatterplot you made in problem 4
- The two boxplots you made in problem 5 side-by-side
- The faceted boxplots you made in problem 6
- Subtitles for all of the above elements

Use `JupyterDash` to display this dashboard directly in your Jupyter notebook.

Any working dashboard that displays all of the above elements will receive full credit. [4 points]

In [ ]:
```
the_title = "The Gender Wage Gap in the United States: A Dashboard"

# create the dashboard
app = JupyterDash(__name__, external_stylesheets=external_stylesheets) # __name__ i


app.layout = html.Div(
    [
        html.H1(the_title),

        dcc.Markdown(children = markdown_text),

        html.H4("Income, occupational prestige, socioeconomic index, and years of e
        dcc.Graph(figure=table_p2),


        html.H4("Responses to men as househod earners and women as homemakers"),
        dcc.Graph(figure=fig_p3),

        html.Div([
                html.H5("Income vs. job prestige for men and women"),
```

```
                html.P(
                    children=(
                        "The income for women is lower than for men at every level
                    ),
                    className="header-description",
                ),
                dcc.Graph(figure=fig_p4),

                ], style = {'width':'32%', 'float':'left', 'backgroundColor':'#ffff
                ),
        html.Div([
            html.H5("Income distribution for men and women"),
            html.P(
                    children=(
                        "Income distributions are similar; although men's 75th quan
                    ),
                    className="header-description",
                ),
            dcc.Graph(figure=box_income)], style = {'width':'32%', 'float':'left',
        html.Div([
            html.H5("Job prestige distribution for men and women"),
            html.P(
                    children=(
                        "Job prestige distributions for men and women are similar."
                    ),
                    className="header-description",
                ),
            dcc.Graph(figure=box_jobprestige)
        ], style = {'width':'32%', 'float':'left', 'backgroundColor':'#ffffff'}),
        html.Div([
            html.H4("Income distribution for men and women by job prestige level"),
            dcc.Graph(figure=fig_box),
        ], style = {'width':'45%', 'float':'left', 'backgroundColor':'#ffffff',}),
        #html.Div([
        #    html.P(
        #            children=(
        #                "In the 1940s, my grandmother, preparing to become a teach
        #            ),
         #           className="header-description",
         #       ),
         #   html.Img(src="https://scontent.forf1-4.fna.fbcdn.net/v/t1.6435-9/52080
         #], style = {'width':'32%', 'float':'right', 'backgroundColor':'#f2f2f2',}

    ],style = {'backgroundColor':'#ffffff', 'padding':'5px', 'width':'1600px, color
)

#if __name__ == '__main__':
#     app.run_server(debug=True, port=8060, use_reloader=False) # use_reloader=Fals

if __name__ == '__main__':
    app.run_server(mode='inline', debug=True, port=8060)
```

```
Dash is running on http://127.0.0.1:8060/
```

The Gender Wage Gap in the United States: A Dashboard

Examining the gender pay gap in the United States attempts to discern the difference in pay between men and women that is a direct effect of being a women, rather than the effect of job selection, schedule selection, or other contributions. Meara, K., Pastore, F. & Webster, A.'s 'The gender pay gap in the USA: a matching study' from 2020's Journal of Population Economics attempts to use a "matching estimator" to control for the effects of job selection, parenthood, schedule selection, and other contributions. Their work finds that there are interaction effects between gender and other variables such as part-time work. England, P., Levine, A., & Mishel, E.'s 2020 article 'Progress toward gender equality in the United States has

Extra Credit (up to 10 bonus points)

Dashboards are all about good design, functionality, and accessability. For this extra credit problem, create another version of the dashboard you built for problem 7, but take extra steps to improve the appearance of the dashboard, add user-inputs, and host it on the internet with its own URL.

Challenge 1: Be creative and use a layout that significantly departs from the one used for the ANES data in the module 12 notebook. A good place to look for inspiration is the Dash gallery. We will award up to 3 bonus points for creativity, novelty, and style.

Challenge 2: Alter the barplot from problem 3 to include user inputs. Create two dropdown menus on the dashboard. The first one should allow a user to display bars for the categories of `satjob`, `relationship`, `male_breadwinner`, `men_bettersuited`, `child_suffer`, or `men_overwork`. The second one should allow a user to group the bars by `sex`, `region`, or `education`. After choosing a feature for the bars and one for the grouping, program the barplot to update automatically to display the user-inputted features. One bonus point will be awarded for a good effort, and 3 bonus points will be awarded for a working user-input barplot in the dashboard.

Challenge 3: Follow the steps listed in the module notebook to deploy your dashboard on Heroku. 1 bonus point will be awarded for a Heroku link to an app that isn't working. 4 bonus points will be awarded for a working Heroku link.

In [ ]:
```
the_title = "THE GENDER WAGE GAP IN THE UNITED STATES"

# create the dashboard
app2 = dash.Dash(__name__, external_stylesheets=external_stylesheets) # __name__ is
#app2 = JupyterDash(__name__, external_stylesheets=external_stylesheets) # __name__

app2.layout = html.Div(
    [
        html.Div([
            html.H1(the_title),
            dcc.Markdown(children = markdown_text),
            html.H5("INCOME, JOB PRESTIGE, SOCIOECONOMIC INDEX & YRS EDUCATION"),
            dcc.Graph(figure=table_p2),
            html.P(children=('')),
        ], style = {'background-color':'#0B2447', 'backgroundColor':'#0B2447', 'wid
        html.Div([
            html.Div([
                html.H5("SURVEY RESPONSES (men and women)"),
                html.P(
                    children=(
                        "Responses to the question of whether men belong in the wor
                    ),
                    className="header-description",
                    ),
                dcc.Graph(figure=fig_p3),
                html.H4("INCOME DISTRIBUTION BY PRESTIGE FOR MEN & WOMEN"),
                dcc.Graph(figure=fig_box),

            ], style = {'background-color':'#0B2447', 'backgroundColor':'#0B2447',
            html.Div([
                html.Div([
                    html.H5("INCOME V. JOB PRESTIGE (men and women)"),
                    html.P(
                        children=(
                            "Women's is lower than men's at every level of job pres
                        ),
                        className="header-description",
                    ),
                    dcc.Graph(figure=fig_p4),
```

```
                    ], style = {'backgroundColor':'#0B2447', 'width':'32%', 'float'
                html.Div([
                    html.H5("INCOME DISTRIBUTION (men and women)"),
                    html.P(
                        children=(
                            "Distributions are similar; although men's 75th quantil
                        ),
                        className="header-description",
                    ),
                    dcc.Graph(figure=box_income)], style = {'width':'32%', 'float':
                html.Div([
                    html.H5("PRESTIGE DISTRIBUTION (men and women)"),
                    html.P(
                        children=(
                            "Job prestige distributions are similar."
                        ),
                        className="header-description",
                    ),
                    dcc.Graph(figure=box_jobprestige),
                    html.P('')
                ], style = {'width':'32%', 'float':'left', 'backgroundColor':'#0B24

            ], style = {'backgroundColor':'#0B2447',  'color': '#F5F3C1'}),
        ], style = {'background-color':'#0B2447', 'backgroundColor':'#0B2447', 'wid
    ],style = {'body.color':'#0B2447','background-color':'#0B2447;','backgroundColo
)


if __name__ == '__main__':
     app2.run_server(debug=True, port=8070, use_reloader=False) # use_reloader=Fals

#if __name__ == '__main__':
#    app2.run_server(mode='inline', debug=True, port=8070)
```

```
Dash is running on http://127.0.0.1:8070/
```

Since you cannot see the dashboard in the PDF printout, here is an image of it:

THE GENDER WAGE GAP IN THE UNITED STATES

Examining the gender pay gap in the United States attempts to discern the difference in pay between men and women that is a direct effect of being a women, rather than the effect of job selection, schedule selection, or other contributions. Meara, K., Pastore, F. & Webster, A.'s 'The gender pay gap in the USA: a matching study' from 2020's Journal of Population Economics attempts to use a "matching estimator" to control for the effects of job selection, parenthood, schedule selection, and other contributions. Their work finds that there are interaction effects between gender and other variables such as part-time work. England, P., Levine, A., & Mishel, E.'s 2020 article 'Progress toward gender equality in the United States has slowed or stalled' further describes how, even with the reversed gap in higher education attainment by women, the gender wage gap has not closed, and progress has stalled since 2018.

The data used in this dashboard is from the General Social Survey (GSS), a nationally representative survey of adults from the National Opinion Research Center (NORC) at the University of Chicago. For upto 80 years, questions have been asked of representative samples of the US population. Included in the GSS are topics related to psychological well-being, social mobility, and, survery questions related to gender, education, income, and job prestige: the topics of this dashboard.

INCOME, JOB PRESTIGE, SOCIOECONOMIC INDEX & YRS EDUCATION

Sex	Income	Job Prestige	SES Index	Yrs Education
female	$47,191.02	44.67	46.58	13.76
male	$53,314.63	44.7	47.38	13.69

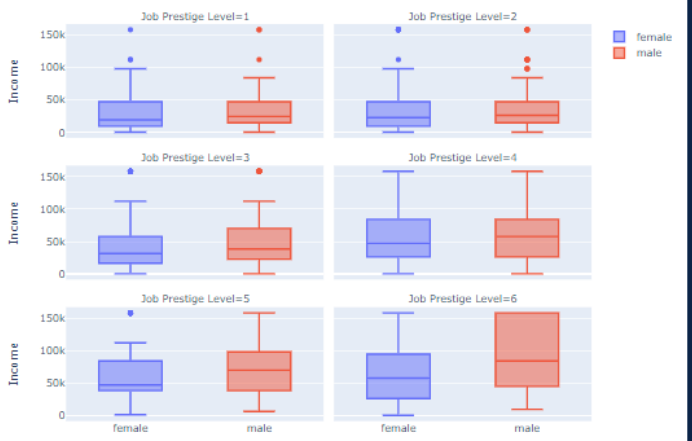
SURVEY RESPONSES (men and women)

Responses to the question of whether men belong in the workplace and women in the home.

INCOME V. JOB PRESTIGE (men and women)

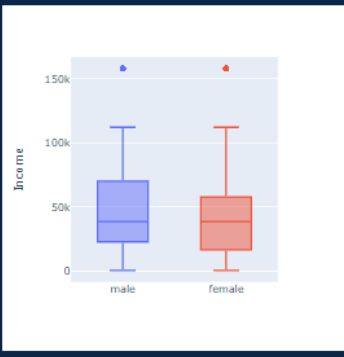
Women's is lower than men's at every level of job prestige.

INCOME DISTRIBUTION BY PRESTIGE FOR MEN & WOMEN



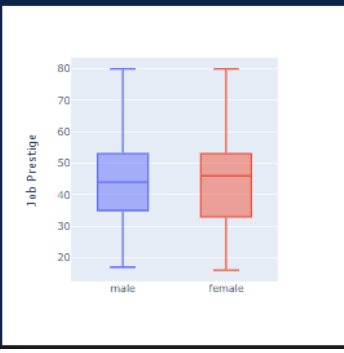
INCOME DISTRIBUTION (men and women)

Distributions are similar; although men's 75th quantile is greater.



PRESTIGE DISTRIBUTION (men and women)

Job prestige distributions are similar.



In [ ]:

In [ ]:

In [ ]: