Lab Assignment 10: Exploratory Data Analysis, Part 1

DS 6001: Practice and Application of Data Science

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.

In this lab, you will be working with the 2018 [General Social Survey (GSS) (http://www.gss.norc.org/)](http://www.gss.norc.org/). The GSS is a sociological survey created and regularly collected since 1972 by the National Opinion Research Center at the University of Chicago. It is funded by the National Science Foundation. The GSS collects information and keeps a historical record of the concerns, experiences, attitudes, and practices of residents of the United States, and it is one of the most important data sources for the social sciences.

The data includes features that measure concepts that are notoriously difficult to ask about directly, such as religion, racism, and sexism. The data also include many different metrics of how successful a person is in his or her profession, including income, socioeconomic status, and occupational prestige. These occupational prestige scores are coded separately by the GSS. The full description of their methodology for measuring prestige is available here: [http://gss.norc.org/Documents/reports/methodological-reports/MR122%20Occupational%20Prestige.pdf (http://gss.norc.org/Documents/reports/methodological-reports/MR122%20Occupational%20Prestige.pdf)](http://gss.norc.org/Documents/reports/methodological-reports/MR122%20Occupational%20Prestige.pdf) Here's a quote to give you an idea about how these scores are calculated:

> Respondents then were given small cards which each had a single occupational titles listed on it. Cards were in English or Spanish. They were given one card at a time in the preordained order. The interviewer then asked the respondent to "please put the card in the box at the top of the ladder if you think that occupation has the highest possible social standing. Put it in the box of the bottom of the ladder if you think it has the lowest possible social standing. If it belongs somewhere in between, just put it in the box that matches the social standing of the occupation."

The prestige scores are calculated from the aggregated rankings according to the method described above.

Problem 0

Import the following packages:

In [1]:
```
import numpy as np
import pandas as pd
import sidetable
import weighted # this is a module of wquantiles, so type pip install wq
uantiles or conda install wquantiles to get access to it
from scipy import stats
from sklearn import manifold
from sklearn import metrics
import prince
from pandas_profiling import ProfileReport
pd.options.display.max_columns = None
```

Then load the GSS data with the following code:

In [2]:
```
%%capture
gss = pd.read_csv("https://github.com/jkropko/DS-6001/raw/master/localda
ta/gss2018.csv",
                  encoding='cp1252', na_values=['IAP','IAP,DK,NA,uncodeab
le', 'NOT SURE',
                                                 'DK', 'IAP, DK, NA, uncod
eable', '.a', "CAN'T CHOOSE"])
```

Problem 1

Drop all columns except for the following:

- `id` - a numeric unique ID for each person who responded to the survey
- `wtss` - survey sample weights
- `sex` - male or female
- `educ` - years of formal education
- `region` - region of the country where the respondent lives
- `age` - age
- `coninc` - the respondent's personal annual income
- `prestg10` - the respondent's occupational prestige score, as measured by the GSS using the methodology described above
- `mapres10` - the respondent's mother's occupational prestige score, as measured by the GSS using the methodology described above
- `papres10` -the respondent's father's occupational prestige score, as measured by the GSS using the methodology described above
- `sei10` - an index measuring the respondent's socioeconomic status
- `satjob` - responses to "On the whole, how satisfied are you with the work you do?"
- `fechld` - 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."
- `fefam` - 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."
- `fepol` - agree or disagree with: "Most men are better suited emotionally for politics than are most women."
- `fepresch` - agree or disagree with: "A preschool child is likely to suffer if his or her mother works."
- `meovrwrk` - agree or disagree with: "Family life often suffers because men concentrate too much on their work."

Then rename any columns with names that are non-intuitive to you to more intuitive and descriptive ones. Finally, replace the "89 or older" values of `age` with 89, and convert `age` to a float data type. [1 point]

In [5]:
```
# Create a list to hold selected features names -- total 17 of them:
selected_features = ['id', 'wtss', 'sex', 'educ',
                     'region', 'age', 'coninc', 'prestg10',
                     'mapres10', 'papres10', 'sei10', 'satjob',
                     'fechld', 'fefam', 'fepol', 'fepresch', 'meovrwrk']
len(selected_features)
```

Out[5]:
```
17
```

In [6]:
```
# Produce and inspect the reduced dataframe:
gss = gss[selected_features]
gss.head()
```

Out[6]:

	id	wtss	sex	educ	region	age	coninc	prestg10	mapres10	papres10	sei10
0	1	2.357493	male	14.0	new england	43	NaN	47.0	31.0	45.0	65.3
1	2	0.942997	female	10.0	new england	74	22782.5000	22.0	32.0	39.0	14.8
2	3	0.942997	male	16.0	new england	42	112160.0000	61.0	32.0	72.0	83.4
3	4	0.942997	female	16.0	new england	63	158201.8412	59.0	NaN	39.0	69.3
4	5	0.942997	male	18.0	new england	71	158201.8412	53.0	35.0	45.0	68.6

Problem 2

Part a

Use the `ProfileReport()` function to generate and embed an HTML formatted exploratory data analysis report in your notebook. Make sure that it includes a "Correlations" report along with "Overview" and "Variables". [1 point]

In [7]:
```
profile = ProfileReport(gss,
                        title="Report",
                        html ={'style':{'full_width':True}},
                        minimal=False)
profile.to_notebook_iframe()
```

Number of variables	17
Number of observations	2348
Missing cells	6276
Missing cells (%)	15.7%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	312.0 KiB
Average record size in memory	136.1 B

Variable types

Numeric	8
Categorical	9

Alerts

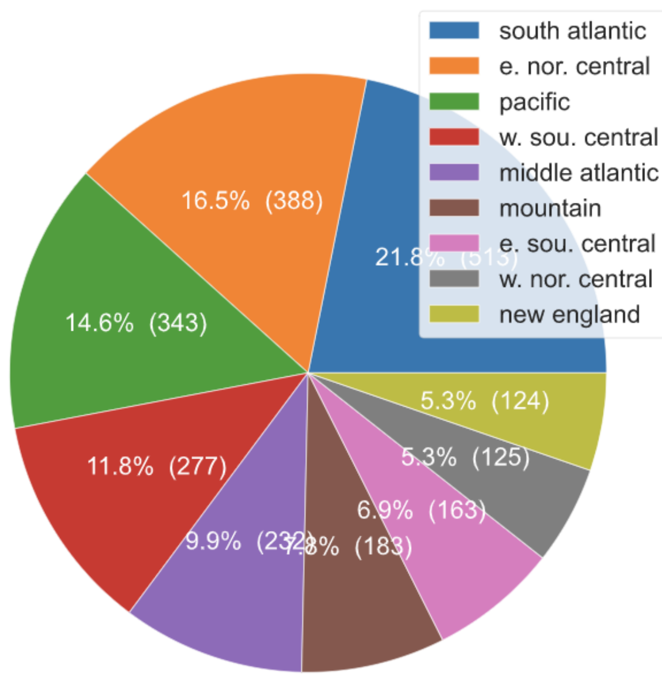
`age` has a high cardinality: 72 distinct values	**High cardinality**
`educ` is highly correlated with `sei10`	**High correlation**
`prestg10` is highly correlated with `sei10`	**High correlation**
`sei10` is highly correlated with `educ` and 1 other fields (educ, prestg10)	**High correlation**
`educ` is highly correlated with `sei10`	**High correlation**
`prestg10` is highly correlated with `sei10`	**High correlation**
`sei10` is highly correlated with `educ` and 1 other fields (educ, prestg10)	**High correlation**
`prestg10` is highly correlated with `sei10`	**High correlation**
`sei10` is highly correlated with `prestg10`	**High correlation**
`id` is highly correlated with `region`	**High correlation**
`educ` is highly correlated with `prestg10` and 1 other fields	**High correlation**

Part b

Looking through the HTML report you displayed in part a, how many people in the data are from New England? [1 point]

From the pie chart below, 124 counts are from New England:

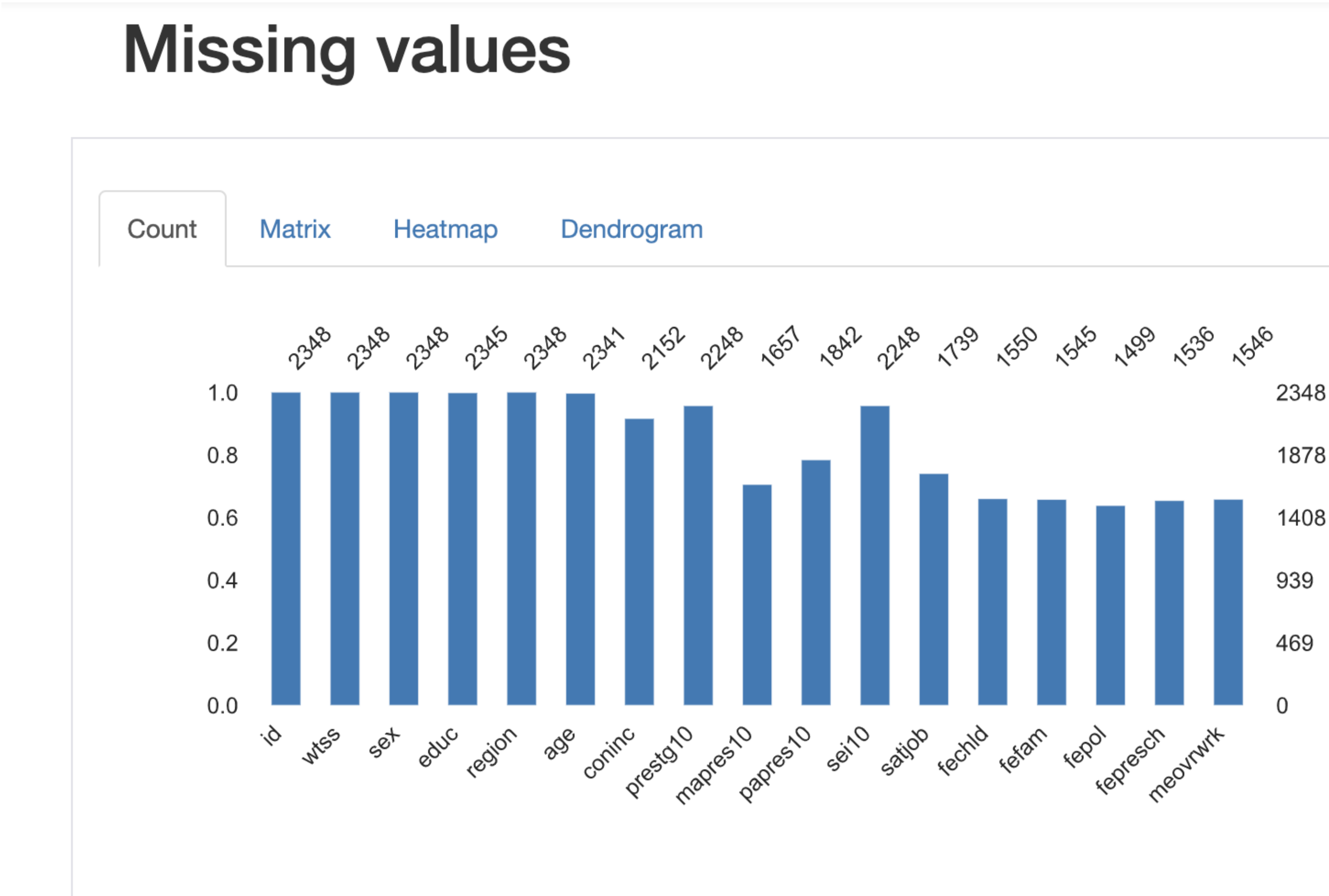
Pie chart



Part c

Looking through the HTML report you displayed in part a, which feature in the data has the highest number of missing values, and what percent of the values are missing for this feature? [1 point]

Feature `fepol` seems to have the highest number of missing values:



Overall, up to 15.7% of sells are missing:

Overview

Overview | Alerts 30 | Reproduction

Dataset statistics

Number of variables	17
Number of observations	2348
Missing cells	6276
Missing cells (%)	15.7%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	312.0 KiB
Average record size in memory	136.1 B

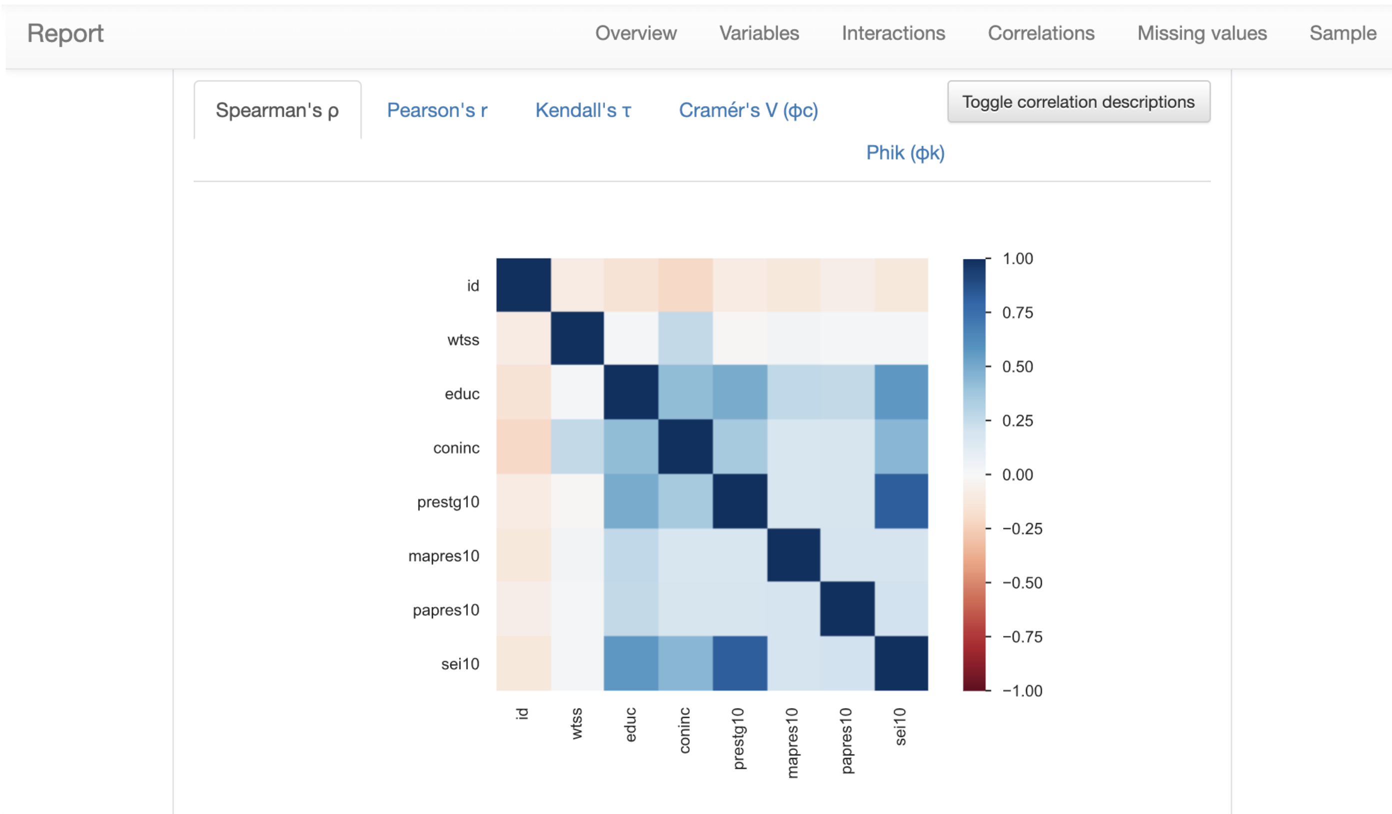
Variable types

Numeric	8
Categorical	9

Part d

Looking through the HTML report you displayed in part a, which two distinct features in the data have the highest correlation? [1 point]

Correlations report included below, `sei10` and `prestg10` have high positive correlation:



Problem 3

On a primetime show on a 24-hour cable news network, two unpleasant-looking men in suits sit across a table from each other, scowling. One says "This economy is failing the middle-class. The average American today is making less than $48,000 a year." The other screams "Fake news! The typical American makes more than $55,000 a year!" Explain, using words and code, how the data can support both of their arguments. Use the sample weights to calculate descriptive statistics that are more representative of the American adult population as a whole. [1 point]

Below are descriptive statistics and histogram for the feature `coninc`, proxy for earned income. The distribution is not normal, as there are a lot of outliers in the right tail, small amount of high earning individuals:

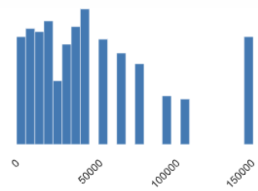
coninc

Real number ($\mathbb{R}_{\geq 0}$)

MISSING

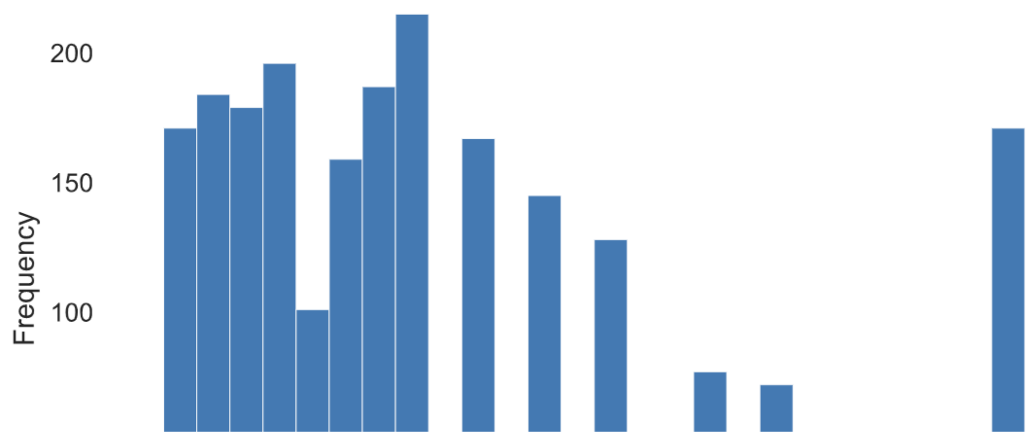
Distinct	26
Distinct (%)	1.2%
Missing	196
Missing (%)	8.3%
Infinite	0
Infinite (%)	0.0%
Mean	49973.96078

Minimum	350.5
Maximum	158201.8412
Zeros	0
Zeros (%)	0.0%
Negative	0
Negative (%)	0.0%
Memory size	18.5 KiB



Toggle details

Statistics | Histogram | Common values | Extreme values



This is obvious when compared the mean and and the median as shown below. On average, expected income is almost 50K, but this observation may not even exist in the sample. The true 50% cut off is at about 39K only as measured by the median.

In [8]:
```
round(np.mean(gss['coninc']), 2)
```

Out[8]:
```
49973.96
```

In [9]:
```
round(np.nanmedian(gss['coninc']), 2)
```

Out[9]:
```
38555.0
```

Problem 4

For each of the following parts,

- generate a table that provides evidence about the relationship between the two features in the data that are relevant to each question,
- interpret the table in words,
- use a hypothesis test to assess the strength of the evidence in the table,
- and provide a **specific and accurate** intepretation of the p-value associated with this hypothesis test beyond "significant or not".

Part a

Is there a gender wage gap? That is, is there a difference between the average incomes of men and women? [2 points]

In [12]:
```
gss.groupby('sex')['coninc'].mean()
```

Out[12]:
```
sex
female    47191.021452
male      53314.626187
Name: coninc, dtype: float64
```

In short, here we are looking at income by sex and testing if there are statistically significant differences in income level when we split our sample in 2 groups based on gender. In other words, we have 2 samples now, males and females, and we want to test if incomes from 2 samples are different and if this difference can be explained by random chance.

In [13]:
```
coninc_male = gss.query("sex == 'male'")['coninc'].dropna()
coninc_female = gss.query("sex == 'female'")['coninc'].dropna()
stats.ttest_ind(coninc_female, coninc_male, equal_var=False)
```

Out[13]:
```
Ttest_indResult(statistic=-3.332824087618215, pvalue=0.0008749557881530
089)
```

Given such small p-value of almost 0, one can reject H0 and conclude that differences in 2 samples are not due to random variation. Once can accept Ha that there may be a gender gap.

Part b

Are there different average values of occupational prestige for different levels of job satisfaction? [2 points]

ANOVA: comparing prestige for more than 2 groups, i.e. 4 groups of different job satisfaction levels.

In [14]:
```
gss.groupby('satjob')['prestg10'].mean()
```

Out[14]:
```
satjob
a little dissat      40.946429
mod. satisfied       42.589984
very dissatisfied    43.000000
very satisfied       46.189320
Name: prestg10, dtype: float64
```

Is there statistically significant difference in prestige when we split our sample in 4 groups based job satisfaction?

In [15]:
```
stats.f_oneway(gss.query("satjob=='a little dissat'")['prestg10'].dropna
(),
               gss.query("satjob=='mod. satisfied'")['prestg10'].dropna
(),
               gss.query("satjob=='very dissatisfied'")['prestg10'].drop
na(),
               gss.query("satjob=='very satisfied'")['prestg10'].dropna
())
```

Out[15]:
```
F_onewayResult(statistic=12.205403153509732, pvalue=6.676686425029878e-
08)
```

From the output below, p-value is very small and thus we can reject H0, i.e. observed differences are not likely due to the random variation. We can conclude that there are different average values of occupational prestige for different levels of job satisfaction.

Problem 5

Report the Pearson's correlation between years of education, socioeconomic status, income, occupational prestige, and a person's mother's and father's occupational prestige? Then perform a hypothesis test for the correlation between years of education and socioeconomic status and provide a **specific and accurate** intepretation of the p-value associated with this hypothesis test beyond "significant or not". [2 points]

In [16]:
```
gss[['educ', 'sei10', 'coninc', 'prestg10', 'mapres10', 'papres10']].cor
r(
    method='pearson')
```

Out[16]:

	educ	sei10	coninc	prestg10	mapres10	papres10
educ	1.000000	0.558169	0.389245	0.479933	0.269115	0.261417
sei10	0.558169	1.000000	0.417210	0.835515	0.203486	0.210451
coninc	0.389245	0.417210	1.000000	0.340995	0.164881	0.171048
prestg10	0.479933	0.835515	0.340995	1.000000	0.189262	0.192180
mapres10	0.269115	0.203486	0.164881	0.189262	1.000000	0.235750
papres10	0.261417	0.210451	0.171048	0.192180	0.235750	1.000000

In [17]:
```
# There seems to be positive correlation between `educ` and `sei10`:
gss[['educ', 'sei10']].corr()
```

Out[17]:

	educ	sei10
educ	1.000000	0.558169
sei10	0.558169	1.000000

In [18]:
```
# Checking this correlation is significuntly different from zero
gss_corrs = gss[['educ', 'sei10']].dropna()
stats.pearsonr(gss_corrs['educ'], gss_corrs['sei10'])
```

Out[18]: (0.5581686004626784, 3.7194488100181494e-184)

Corresponding p-value is very close to zero, thus one can reject H0 and conclude that there is a non-zero correlation between `educ` and `sei10`.

Problem 6

Create a new categorical feature for age groups, with categories for 18-35, 36-49, 50-69, and 70 and older (see the module 8 notebook for an example of how to do this).

Then create a cross-tabulation in which the rows represent age groups and the columns represent responses to the statement that "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." Rearrange the columns so that they are in the following order: strongly agree, agree, disagree, strongly disagree. Place row percents in the cells of this table.

Finally, use a hypothesis test that can tell use whether there is enough evidence to conclude that these two features have a relationship, and provide a specific and accurate intepretation of the p-value. [2 points]

In [19]:
```python
# First, remove 7 missing values
gss.age.dropna()
```

Out[19]:
```
0       43
1       74
2       42
3       63
4       71
        ..
2343    37
2344    75
2345    67
2346    72
2347    79
Name: age, Length: 2341, dtype: object
```

In [20]:
```python
# Second, remap '89 and older' to a string 90 that could be converted to int later
gss.age.replace({'89 or older':'90'}, inplace=True)
```

In [21]:
```python
# Convert all ages to int
gss.age = pd.to_numeric(gss['age'])
```

In [22]:
```python
# Create bins and labels
bins = [0, 35, 49, 69, 90]
labels = ['18-35', '36-49', '50-69', '70 and older']
```

In [23]:
```python
# Cut continious age into categorical age groups
gss['age_group'] = pd.cut(x=gss['age'], bins=bins, labels=labels)
```

In [24]:
```python
gss.head()
```

Out[24]:

	id	wtss	sex	educ	region	age	coninc	prestg10	mapres10	papres10	sei10
0	1	2.357493	male	14.0	new england	43.0	NaN	47.0	31.0	45.0	65.3
1	2	0.942997	female	10.0	new england	74.0	22782.5000	22.0	32.0	39.0	14.8
2	3	0.942997	male	16.0	new england	42.0	112160.0000	61.0	32.0	72.0	83.4
3	4	0.942997	female	16.0	new england	63.0	158201.8412	59.0	NaN	39.0	69.3
4	5	0.942997	male	18.0	new england	71.0	158201.8412	53.0	35.0	45.0	68.6

In [25]:
```
crosstab = pd.crosstab(gss.fefam, gss.age_group, normalize='index')
stats.chi2_contingency(crosstab.values)
```

Out[25]:
```
(0.26483272811022085,
 0.9999980816990571,
 9,
 array([[0.24055218, 0.22080774, 0.32258349, 0.21605659],
        [0.24055218, 0.22080774, 0.32258349, 0.21605659],
        [0.24055218, 0.22080774, 0.32258349, 0.21605659],
        [0.24055218, 0.22080774, 0.32258349, 0.21605659]]))
```

Chi2 test of association shows very large p-value and thus one can't reject H0 that there are no differences between the groups. There is no statistically significant relationship between the groups.

Problem 7

For this problem, you will conduct and interpret a correspondence analysis on the categorical features that ask respondents to state the extent to which they agree or disagree with the statements:

- "A working mother can establish just as warm and secure a relationship with her children as a mother who does not work."
- "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."
- "Most men are better suited emotionally for politics than are most women."
- "A preschool child is likely to suffer if his or her mother works."
- "Family life often suffers because men concentrate too much on their work."

- `fechld` - 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."
- `fefam` - 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."
- `fepol` - agree or disagree with: "Most men are better suited emotionally for politics than are most women."
- `fepresch` - agree or disagree with: "A preschool child is likely to suffer if his or her mother works."
- `meovrwrk` - agree or disagree with: "Family life often suffers because men concentrate too much on their work."

Part a

Conduct a correspondence analysis using the observed features listed above that measures two latent features. Plot the two latent categories for each category in each of the features used in the analysis. [2 points]

In [26]:
```
# Group together the desired features
selection = ['fechld', 'fefam', 'fepol', 'fepresch', 'meovrwrk']
```

In [68]:

```
# Take a copy of slice of data with selected features
gss.dropna(subset=selection, axis=0, inplace=True)
```

In [74]:

```
# Check if there are any missing values
gss[selection].isnull().sum()
```

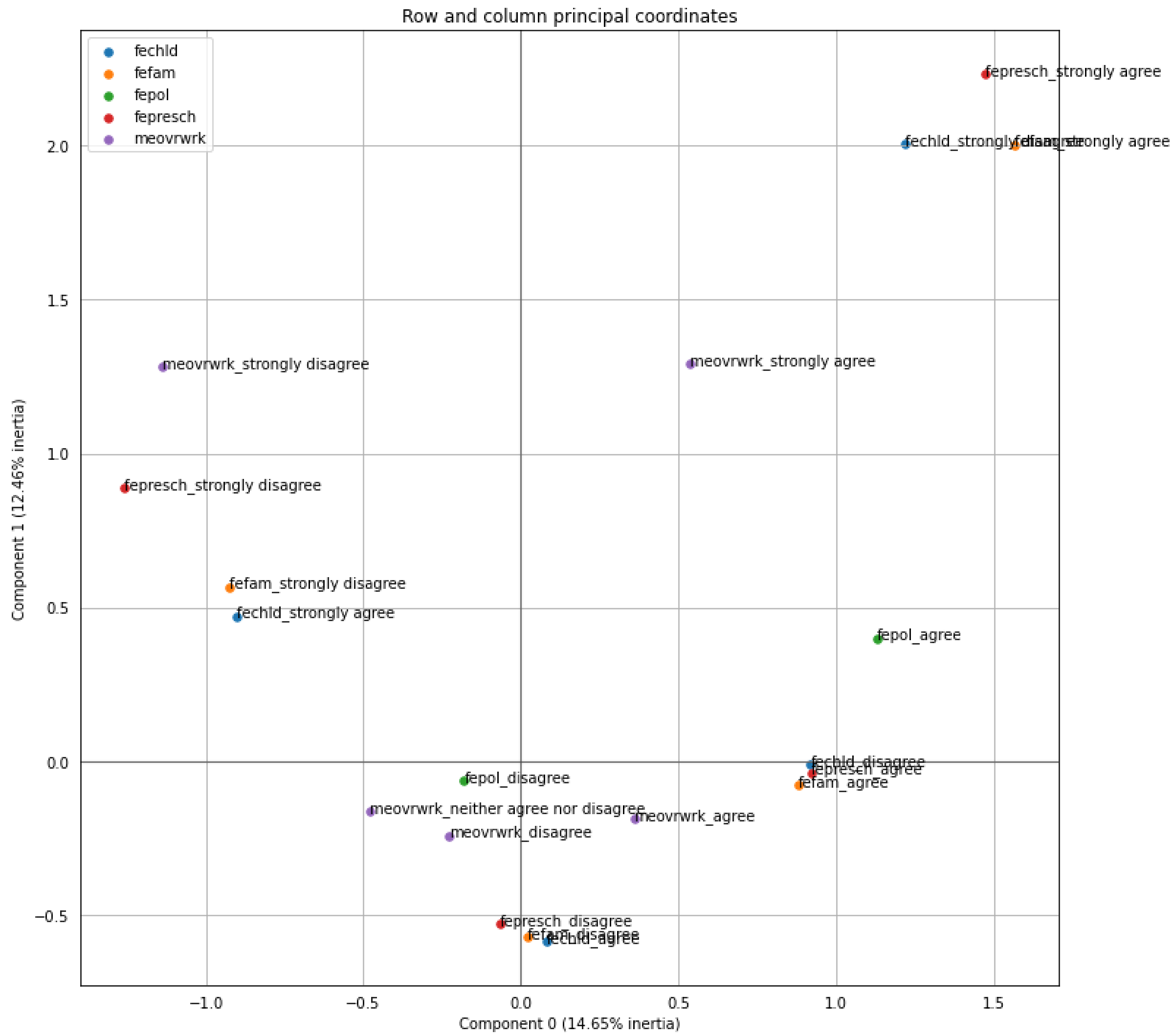
Out[74]:

```
fechld      0
fefam       0
fepol       0
fepresch    0
meovrwrk    0
dtype: int64
```

In [75]:

```
# Instanciate MCA for 2 latent features
mca = prince.MCA(n_components=2)
# Fit the estimator
mca = mca.fit(gss[selection])
```

In [77]:
```python
import matplotlib.pyplot as plt
%matplotlib inline
# Plot coordinates
ax = mca.plot_coordinates(
     X=gss[selection],
     ax=None,
     figsize=(12, 12),
     show_row_points=False,
     row_points_size=10,
     show_row_labels=False,
     show_column_points=True,
     column_points_size=30,
     show_column_labels=True,
     legend_n_cols=1)
```



Part b

Display the latent features for every category in the observed features, sorted by the first latent feature. Describe in words what concept this feature is attempting to measure, and give the feature a name. [2 points]

In [78]:
```
mca.column_coordinates(gss[selection]).sort_values(0, ascending=False)
```

Out[78]:

	0	1
fefam_strongly agree	1.564729	2.002646
fepresch_strongly agree	1.474167	2.234067
fechld_strongly disagree	1.218713	2.005353
fepol_agree	1.131106	0.399629
fepresch_agree	0.919992	-0.036427
fechld_disagree	0.918042	-0.010334
fefam_agree	0.878982	-0.076575
meovrwrk_strongly agree	0.536783	1.291980
meovrwrk_agree	0.358280	-0.187028
fechld_agree	0.080483	-0.586388
fefam_disagree	0.022158	-0.572454
fepresch_disagree	-0.067884	-0.529276
fepol_disagree	-0.180400	-0.063737
meovrwrk_disagree	-0.228691	-0.242578
meovrwrk_neither agree nor disagree	-0.480747	-0.163822
fechld_strongly agree	-0.901120	0.472187
fefam_strongly disagree	-0.922032	0.566789
meovrwrk_strongly disagree	-1.135405	1.283844
fepresch_strongly disagree	-1.258061	0.886712

From the MCA above it seems that the most dominant feature associated with the first latent variable is:

- fefam - 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."

Part c

We can use the results of the MCA model to conduct some cool EDA. For one example, follow these steps:

1. Use the `.row_coordinates()` method to calculate values of the latent feature for every row in the data you passed to the MCA in part a. Extract the first column and store it in its own dataframe.
2. To join it with the full, cleaned GSS data based on row numbers (instead of on a primary key), use the `.join()` method. For example, if we named the cleaned GSS data `gss_clean` and if we named the dataframe in step 1 `latentfeature`, we can type

```
gss_clean = gss_clean.join(latentfeature, how="outer")
```

3. Create a cross-tabuation with age categories (that you constructed in problem 5) in the rows and sex in the columns. Instead of a frequency, place the mean value of the latent feature in the cells.

What does this table tell you about the relationship between sex, age, and the latent feature? [2 points]

In [79]:
```
latentfeature = mca.row_coordinates(gss[selection])[0]
latentfeature
```

Out[79]:
```
0       -0.202210
2       -0.423361
3       -0.195576
5       -0.240092
8        0.341541
           ...
2341     1.219022
2343    -0.521776
2344    -0.423361
2346     1.076896
2347     1.440616
Name: 0, Length: 1454, dtype: float64
```

In [80]:
```
gss = gss.join(latentfeature, how="outer")
```

In [83]:
```
gss.columns
```

Out[83]:
```
Index([       'id',      'wtss',       'sex',      'educ',    'region',
             'age',    'coninc',  'prestg10',  'mapres10',  'papres10',
           'sei10',    'satjob',    'fechld',     'fefam',     'fepol',
        'fepresch',  'meovrwrk', 'age_group',           0],
      dtype='object')
```

In [84]:
```
pd.crosstab(gss.sex, gss.age_group, normalize='index')
```

Out[84]:

age_group	18-35	36-49	50-69	70 and older
sex				
female	0.279609	0.242979	0.312576	0.164835
male	0.248013	0.232114	0.367250	0.152623

From the cross-tabulation above, it seems the sex-based split is pretty even, but the feature importance differs by the age groups, increasing sharply at 50-69, but falling for '70 and older'. This is not conclusive though because age groups were assigned without any prior knowledge or theoretical justification.

In [ ]: