ANL252

ECA

B1972075

Ho Qi-Wei

**1.(a)**

import pandas as pd

########## start of part (a) #############

#1.(a)(i) Read “ship.csv” as pd

ship = pd.read\_csv(*"ship.csv"*, na\_values=[*'.'*])

#1.(a)(ii) Rename the dataframe labels to be more readable

ship.rename(columns={

*'T'*: *'types'*,

*'MS'*: *'s\_months'*,

*'P'*: *'o\_periods'*,

*'A'*: *'c\_years'*,

*'Y'*: *'incidents'*}, inplace=True)

#1.(a)(iii) Generate another dataframe that group by types and o\_periods and compute means of s\_months and incidents, and round to integer

shipgroup = ship.groupby([*'types'*, *'o\_periods'*])[

[*'s\_months'*, *'incidents'*]].mean().round(0).astype(int)

#1.(a)(iv) Replace all NaN for s\_months to be the average of other ships that share same type and o\_periods

for idx in ship[ship[*'s\_months'*].isna()].index:

\_type = ship.loc[idx, *'types'*]

o\_periods = ship.loc[idx, *'o\_periods'*]

ship.loc[idx, *'s\_months'*] = shipgroup.loc[\_type, o\_periods][*'s\_months'*]

#1.(a)(iv) Replace all NaN for incidents to be the average of other ships that share same type and o\_periods

for idx in ship[ship[*'incidents'*].isna()].index:

\_type = ship.loc[idx, *'types'*]

o\_periods = ship.loc[idx, *'o\_periods'*]

ship.loc[idx, *'incidents'*] = shipgroup.loc[\_type, o\_periods][*'incidents'*]

#1.(a)(v) Save target variable "incidents" in a dataframe named Y

Y = ship[*"incidents"*]

########## end of part (a) #############

**1.(b)**

import pandas as pd

import numpy as np

ship = pd.read\_csv(*"ship.csv"*, na\_values=[*'.'*])

########## start of part (a) #############

#1.(a)(i) Read “ship.csv” as pd

ship = pd.read\_csv(*"ship.csv"*, na\_values=[*'.'*])

#1.(a)(ii) Rename the dataframe labels to be more readable

ship.rename(columns={

*'T'*: *'types'*,

*'MS'*: *'s\_months'*,

*'P'*: *'o\_periods'*,

*'A'*: *'c\_years'*,

*'Y'*: *'incidents'*}, inplace=True)

#1.(a)(iii) Generate another dataframe that group by types and o\_periods and compute means of s\_months and incidents, and round to integer

shipgroup = ship.groupby([*'types'*, *'o\_periods'*])[

[*'s\_months'*, *'incidents'*]].mean().round(0).astype(int)

#1.(a)(iv) Replace all NaN for s\_months to be the average of other ships that share same type and o\_periods

for idx in ship[ship[*'s\_months'*].isna()].index:

\_type = ship.loc[idx, *'types'*]

o\_periods = ship.loc[idx, *'o\_periods'*]

ship.loc[idx, *'s\_months'*] = shipgroup.loc[\_type, o\_periods][*'s\_months'*]

#1.(a)(iv) Replace all NaN for incidents to be the average of other ships that share same type and o\_periods

for idx in ship[ship[*'incidents'*].isna()].index:

\_type = ship.loc[idx, *'types'*]

o\_periods = ship.loc[idx, *'o\_periods'*]

ship.loc[idx, *'incidents'*] = shipgroup.loc[\_type, o\_periods][*'incidents'*]

#1.(a)(v) Save target variable "incidents" in a dataframe named Y

Y = ship[*"incidents"*]

########## end of part (a) #############

########## start of part (b) #############

#1.(b)(i) Convert data type to Categorical

ship[*'types'*] = pd.Categorical(ship.types)

ship[*'c\_years'*] = pd.Categorical(ship.c\_years)

ship[*'o\_periods'*] = pd.Categorical(ship.o\_periods)

#1.(b)(ii) Construct dummy variables X

X = pd.get\_dummies(ship[[*'types'*, *'c\_years'*, *'o\_periods'*]])

#1.(b)(iii) log-transform s\_months and append to ship and X dataframe

ship[*'log\_s\_months'*] = np.log2(ship[*'s\_months'*])

X[*'log\_s\_months'*] = np.log2(ship[*'s\_months'*])

########## end of part (b) #############

**1.(c)**

The dataset we have has very limited data. In fact, each row in the dataframe is unique in terms of a combination of factors (types, construction years, operation periods, aggregated months of service). Therefore, it is impossible for us to split the data into train and validation set as by doing so, we will essentially lose some of the unique data that carries unique features. Incomplete training set will then result in the inaccuracy of the model. Therefore, in order to let the model fully learn from all kinds of scenarios, we need to supply the entire dataframe of all unique data as the training set.

**1.(d)**

import pandas as pd

import numpy as np

import sqlite3

########## start of part (a) #############

#1.(a)(i) Read “ship.csv” as pd

ship = pd.read\_csv(*"ship.csv"*, na\_values=[*'.'*])

#1.(a)(ii) Rename the dataframe labels to be more readable

ship.rename(columns={

*'T'*: *'types'*,

*'MS'*: *'s\_months'*,

*'P'*: *'o\_periods'*,

*'A'*: *'c\_years'*,

*'Y'*: *'incidents'*}, inplace=True)

#1.(a)(iii) Generate another dataframe that group by types and o\_periods and compute means of s\_months and incidents, and round to integer

shipgroup = ship.groupby([*'types'*, *'o\_periods'*])[

[*'s\_months'*, *'incidents'*]].mean().round(0).astype(int)

#1.(a)(iv) Replace all NaN for s\_months to be the average of other ships that share same type and o\_periods

for idx in ship[ship[*'s\_months'*].isna()].index:

\_type = ship.loc[idx, *'types'*]

o\_periods = ship.loc[idx, *'o\_periods'*]

ship.loc[idx, *'s\_months'*] = shipgroup.loc[\_type, o\_periods][*'s\_months'*]

#1.(a)(iv) Replace all NaN for incidents to be the average of other ships that share same type and o\_periods

for idx in ship[ship[*'incidents'*].isna()].index:

\_type = ship.loc[idx, *'types'*]

o\_periods = ship.loc[idx, *'o\_periods'*]

ship.loc[idx, *'incidents'*] = shipgroup.loc[\_type, o\_periods][*'incidents'*]

#1.(a)(v) Save target variable "incidents" in a dataframe named Y

Y = ship[*"incidents"*]

########## end of part (a) #############

########## start of part (b) #############

#1.(b)(i) Convert data type to Categorical

ship[*'types'*] = pd.Categorical(ship.types)

ship[*'c\_years'*] = pd.Categorical(ship.c\_years)

ship[*'o\_periods'*] = pd.Categorical(ship.o\_periods)

#1.(b)(ii) Construct dummy variables X

X = pd.get\_dummies(ship[[*'types'*, *'c\_years'*, *'o\_periods'*]])

#1.(b)(iii) log-transform s\_months and append to ship and X dataframe

ship[*'log\_s\_months'*] = np.log2(ship[*'s\_months'*])

X[*'log\_s\_months'*] = np.log2(ship[*'s\_months'*])

########## end of part (b) #############

########## start of part (d) #############

# Save to CSV

ship.to\_csv(*'ship\_prepared.csv'*)

# Save to sqlite db

db = *'ship.db'*

conn = sqlite3.connect(db)

ship.to\_sql(db, conn)

########## end of part (d) #############

**2.(a)**

<https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.PoissonRegressor.html>

The above link is the corresponding module in sklearn. The Poisson Regressor model module in sklearn is an estimator that, given our observation data X and Y, it picks the best possible poisson regression model to fit the data and use it for prediction. The fit function put our input X and Y values into the model and try to find the best-fit Poisson regression line that fits the data which results in the least possible error. The prediction function then uses this best-fit model to give a predicted Y value based on the input X value on the regression line. The parameters of the fit function mainly take in both the training data X as well the expected target values Y. The predict function only takes in the samples X and will put the samples on the best-fit regression line to get the corresponding Y value as predicted results.

**2.(b)**

import pandas as pd

import numpy as np

from sklearn import linear\_model

def get\_train\_test\_set():

# From Question 1

#1.(a)(i) Read “ship.csv” as pd

ship = pd.read\_csv("ship.csv", na\_values=['.'])

#1.(a)(ii) Rename the dataframe labels to be more readable

ship.rename(columns={

'T': 'types',

'MS': 's\_months',

'P': 'o\_periods',

'A': 'c\_years',

'Y': 'incidents'}, inplace=True)

#1.(a)(iii) Generate another dataframe that group by types and o\_periods and compute means of s\_months and incidents, and round to integer

shipgroup = ship.groupby(['types', 'o\_periods'])[

['s\_months', 'incidents']].mean().round(0).astype(int)

#1.(a)(iv) Replace all NaN for s\_months to be the average of other ships that share same type and o\_periods

for idx in ship[ship['s\_months'].isna()].index:

\_type = ship.loc[idx, 'types']

o\_periods = ship.loc[idx, 'o\_periods']

ship.loc[idx, 's\_months'] = shipgroup.loc[\_type, o\_periods]['s\_months']

#1.(a)(iv) Replace all NaN for incidents to be the average of other ships that share same type and o\_periods

for idx in ship[ship['incidents'].isna()].index:

\_type = ship.loc[idx, 'types']

o\_periods = ship.loc[idx, 'o\_periods']

ship.loc[idx, 'incidents'] = shipgroup.loc[\_type, o\_periods]['incidents']

#1.(a)(v) Save target variable "incidents" in a dataframe named Y

Y = ship["incidents"]

#1.(b)(i) Convert data type to Categorical

ship['types'] = pd.Categorical(ship.types)

ship['c\_years'] = pd.Categorical(ship.c\_years)

ship['o\_periods'] = pd.Categorical(ship.o\_periods)

#1.(b)(ii) Construct dummy variables X

X = pd.get\_dummies(ship[['types', 'c\_years', 'o\_periods']])

#1.(b)(iii) log-transform s\_months and append to ship and X dataframe

ship['log\_s\_months'] = np.log2(ship['s\_months'])

X['log\_s\_months'] = np.log2(ship['s\_months'])

return X, Y

def main():

X, y = get\_train\_test\_set()

model = linear\_model.PoissonRegressor()

model.fit(X, y)

score = model.score(X, y)

coefficients = pd.DataFrame(np.transpose(model.coef\_))

coefficients.set\_index(X.columns, inplace=True)

coefficients.columns = ['coefficient']

print("[Model Report]")

print("Model accuracy score: {:.2f}".format(score))

print("Model coefficients:")

print(coefficients)

main()

Timeline

Description automatically generated with medium confidence

**2.(c)**

import pandas as pd

import numpy as np

from sklearn import linear\_model

def get\_train\_test\_set():

# From Question 1

#1.(a)(i) Read “ship.csv” as pd

ship = pd.read\_csv("ship.csv", na\_values=['.'])

#1.(a)(ii) Rename the dataframe labels to be more readable

ship.rename(columns={

'T': 'types',

'MS': 's\_months',

'P': 'o\_periods',

'A': 'c\_years',

'Y': 'incidents'}, inplace=True)

#1.(a)(iii) Generate another dataframe that group by types and o\_periods and compute means of s\_months and incidents, and round to integer

shipgroup = ship.groupby(['types', 'o\_periods'])[

['s\_months', 'incidents']].mean().round(0).astype(int)

#1.(a)(iv) Replace all NaN for s\_months to be the average of other ships that share same type and o\_periods

for idx in ship[ship['s\_months'].isna()].index:

\_type = ship.loc[idx, 'types']

o\_periods = ship.loc[idx, 'o\_periods']

ship.loc[idx, 's\_months'] = shipgroup.loc[\_type, o\_periods]['s\_months']

#1.(a)(iv) Replace all NaN for incidents to be the average of other ships that share same type and o\_periods

for idx in ship[ship['incidents'].isna()].index:

\_type = ship.loc[idx, 'types']

o\_periods = ship.loc[idx, 'o\_periods']

ship.loc[idx, 'incidents'] = shipgroup.loc[\_type, o\_periods]['incidents']

#1.(a)(v) Save target variable "incidents" in a dataframe named Y

Y = ship["incidents"]

#1.(b)(i) Convert data type to Categorical

ship['types'] = pd.Categorical(ship.types)

ship['c\_years'] = pd.Categorical(ship.c\_years)

ship['o\_periods'] = pd.Categorical(ship.o\_periods)

#1.(b)(ii) Construct dummy variables X

X = pd.get\_dummies(ship[['types', 'c\_years', 'o\_periods']])

#1.(b)(iii) log-transform s\_months and append to ship and X dataframe

ship['log\_s\_months'] = np.log2(ship['s\_months'])

X['log\_s\_months'] = np.log2(ship['s\_months'])

return X, Y

# For Question 2.(c)

def get\_deviance(Y, expectedY):

"""Get the deviance of Y and its expected value E(Y)

Args:

Y (list): Y list

expectedY (list): expected E(Y) list

Returns:

float: a deviance score

"""

idx = 0

\_d = 0

while idx < len(Y):

log\_term = Y[idx] \* \

np.log((Y[idx] / expectedY[idx])) if Y[idx] != 0 else 0

exp\_term = Y[idx] - expectedY[idx]

\_d += (log\_term - exp\_term)

idx += 1

D = 2 \* \_d

return D

def main():

X, y = get\_train\_test\_set()

model = linear\_model.PoissonRegressor()

model = model.fit(X, y)

expectedY = model.predict(X)

D = get\_deviance(y.to\_list(), expectedY)

print("Deviance: {}".format(D))

main()



∴ D is 50.44 (2 decimal places).