НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ УНИВЕРСИТЕТ «МИФИ»

Отчет по лабораторной работе №3

по курсу: «Специальные технологии баз данных»

Выполнил: *студент группы С20-702* *Нуритдинходжаева А.А.*

(подпись) (Фамилия И.О.)

Проверил: *Манаенкова Т.А.*

(оценка) (подпись) (Фамилия И.О.)

Москва 2023 г

Условие задания

Вариант 1

1. Соберите в интернете набор данных о курсах американского доллара, канадского доллара и евро не менее, чем за последние 50 месяцев (как минимум одно наблюдение на месяц). Курсы валют должны быть приведены к одинаковым единицам измерения. Например, количество рублей за единицу валюты или количество фунтов стерлингов за единицу валюты и т.д.

2. Загрузите собранную информацию в набор Pandas DataFrame, назвав переменные USD, CAD, EUR соответственно.

3. Выполните графическое исследование для каждой из переменных на предмет совпадения её распределения с нормальным при помощи графиков P-P и Q-Q. При необходимости выполните нормализацию или стандартизацию данных.

4. Проверьте приблизительное равенство моды, медианы и среднего арифметического значения исследуемого набора данных. Постройте их на гистограмме исследуемого набора данных.

5. Выполните аналитическое исследование для каждой из переменных на предмет совпадения её распределения с нормальным. Обосновано выберите один из методов: Шапиро-Уилка, Андерсона-Дарлинга или Колмоговора-Смирнова.

6. Удалите наиболее заметные выбросы и повторите исследование.

7. Для каждой из пар переменных выполните графическое исследование данной пары на предмет взаимной корреляции.

8. Для каждой из пар переменных выполните оценку с помощью четырёх методов (Пирсона, Спирмена, Кендалла). Объясните, какой из методов было лучше использовать в вашем случае. Дайте интерпретацию полученных результатов.

9. Для каждой из пар переменных постройте график, отображающий все точки наблюдений и функцию линейной регрессии.

Решение

```
import pandas as pd
import statsmodels.api as sm
import matplotlib.pyplot as plt
from sklearn import preprocessing
import scipy
import seaborn as sns

#2
dt = pd.read_csv('курс валют.csv', delimiter = ',')
print(dt)

#3 нормализация и стандартизация
scaler_std = preprocessing.StandardScaler()
scaler_range = preprocessing.MinMaxScaler()
x = scaler_range.fit_transform(dt[['USD']])
dt['USD'] = x[0:]
x = scaler_range.fit_transform(dt[['CAD']])
dt['CAD'] = x[0:]
x = scaler_range.fit_transform(dt[['EUR']])
dt['EUR'] = x[0:]
print(dt)

#USD
probplot = sm.ProbPlot(dt['USD'])
probplot.qqplot(line = 'r')
probplot.ppplot(line = 'r')

plt.show()

#CAD
probplot = sm.ProbPlot(dt['CAD'])
probplot.qqplot(line = 'r')
probplot.ppplot(line = 'r')

plt.show()

#EUR
probplot = sm.ProbPlot(dt['EUR'])
probplot.qqplot(line = 'r')
probplot.ppplot(line = 'r')

plt.show()

dt1 = pd.read_csv('курс валют.csv', delimiter = ',') #Начальные данные

#4
#USD
fig, ax = plt.subplots()
ax.vlines(dt1['USD'].mean(), 0, dt1['USD'].size, colors = "Red")
ax.vlines(dt1['USD'].median(), 0, dt1['USD'].size, colors = "Green")
ax.vlines(dt1['USD'].mode()[0], 0, dt1['USD'].size, colors = "Yellow")
dt1['USD'].plot.hist()
plt.show()

#CAD
```

```
fig, ax = plt.subplots()
ax.vlines(dt1['CAD'].mean(), 0, dt1['CAD'].size, colors = "Red")
ax.vlines(dt1['CAD'].median(), 0, dt1['CAD'].size, colors = "Green")
ax.vlines(dt1['CAD'].mode()[0], 0, dt1['CAD'].size, colors = "Yellow")
dt1['CAD'].plot.hist()
plt.show()

#EUR
fig, ax = plt.subplots()
ax.vlines(dt1['EUR'].mean(), 0, dt1['EUR'].size, colors = "Red")
ax.vlines(dt1['EUR'].median(), 0, dt1['EUR'].size, colors = "Green")
ax.vlines(dt1['EUR'].mode()[0], 0, dt1['EUR'].size, colors = "Yellow")
dt1['EUR'].plot.hist()
plt.show()

#Графики распределения
dt['USD'].plot.kde()
plt.show

dt['CAD'].plot.kde()
plt.show

dt['EUR'].plot.kde()
plt.show

#Возьмем метод Шапиро-Уилка, так как наша выборка не большая и данные имеют распределение близкое к нормальному и нашей целью является
#определение того на сколько данные близки к нормальному распределению

#5
#Шапиро-Уилка (уровень значимости возьмем 5% = 0,05)
w1 = scipy.stats.shapiro(list(dt1['USD']))
print(w1)
w2 = scipy.stats.shapiro(list(dt1['CAD']))
print(w2)
w3 = scipy.stats.shapiro(list(dt1['EUR']))
print(w3)

# w1, w2 отвергаются так как p-value < 0.05, w3 принимается

#6 выбросов не найдено

#7

dt1.plot.scatter(x = 'USD', y = 'CAD',  c = 'Red')
dt1.plot.scatter(x = 'USD', y = 'EUR',  c = 'Green')
dt1.plot.scatter(x = 'EUR', y = 'CAD',  c = 'Blue')
plt.show()

#8
#USD, CAD
cvm = dt1[['USD', 'CAD']].cov()
print(cvm, '\n')

pr = scipy.stats.pearsonr(dt1['USD'], dt1['CAD'])
print(pr, '\n')
```

```
pr = scipy.stats.spearmanr(dt1['USD'], dt1['CAD'])
print(pr, '\n')

pr = scipy.stats.kendalltau(dt1['USD'], dt1['CAD'])
print(pr)

#CAD, EUR
cvm = dt1[['CAD', 'EUR']].cov()
print(cvm, '\n')

pr = scipy.stats.pearsonr(dt1['CAD'], dt1['EUR'])
print(pr, '\n')

pr = scipy.stats.spearmanr(dt1['CAD'], dt1['EUR'])
print(pr, '\n')

pr = scipy.stats.kendalltau(dt1['CAD'], dt1['EUR'])
print(pr)

#USD, EUR
cvm = dt1[['USD', 'EUR']].cov()
print(cvm, '\n')

pr = scipy.stats.pearsonr(dt1['USD'], dt1['EUR'])
print(pr, '\n')

pr = scipy.stats.spearmanr(dt1['USD'], dt1['EUR'])
print(pr, '\n')

pr = scipy.stats.kendalltau(dt1['USD'], dt1['EUR'])
print(pr)

#Все три критерия корреляции позволяют дать оценку нулевой гипотезе (H0) об отсутствии
корреляции между двумя переменными.
#Значения критериев Пирсона, Спирмена, Кендалла находятся в диапазоне от -1 до 1 (0 – нет
корреляции, 1 – полная корреляция,
#-1 – полная обратная корреляция).
#Также Python выводит также p-value, определяющее уровень значимости гипотезы H0. Если он
очень маленький (как в нашем случае), то гипотеза об
#отсутствии корреляции должна быть отброшена.
#лучше использовать метод Пирсона
```