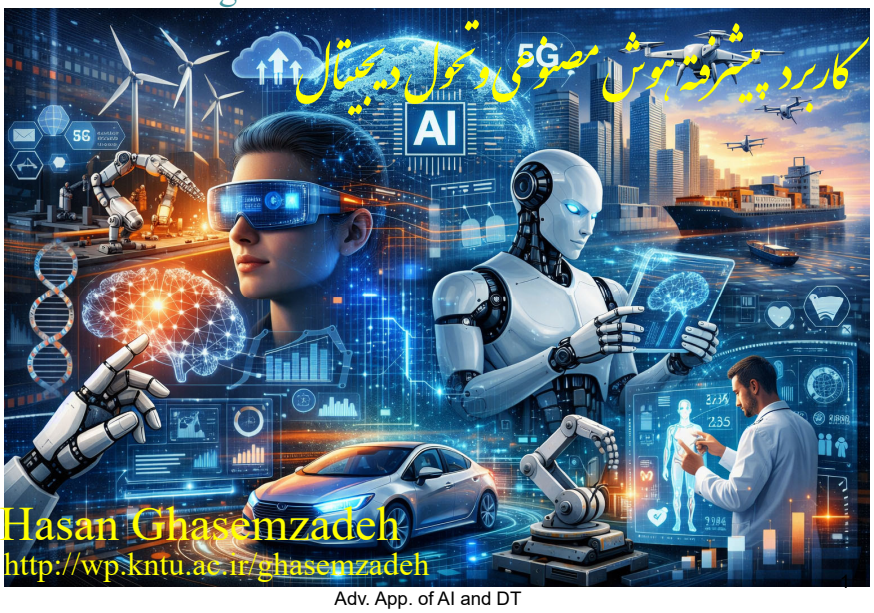
Advanced Application of Artificial Intelligence and Digital Transformation

K.N. Toosi University of Technology



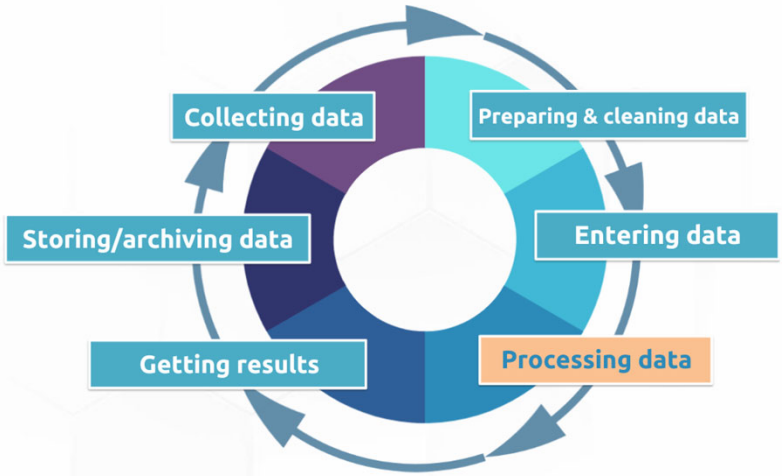
Adv. App. of AI and DT

Soft computing

K.N.TOOSI University of Technology

Adv. App. of AI and DT 2

پردازش داده‌ها



پردازش داده‌ها

- پاکسازی داده‌ها
- داده های پوچ null
- روش وان هات One Hot Encoding

پاکسازی داده‌ها

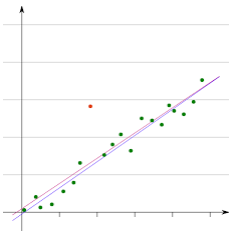
- موفقیت یک پروژه بطور قابل توجهی به پاکسازی داده‌ها وابسته است.
- داده‌ی خوب، بهتر از الگوریتم‌های پیچیده است.

- در پاکسازی، داده‌های گم‌شده، تکراری یا غیرمرتبط شناسایی و حذف می‌شوند.
- داده‌های غلط یا نامناسب می‌توانند عملکرد مدل یادگیری ماشین را به خطر بیاندازند.

پاکسازی داده‌ها

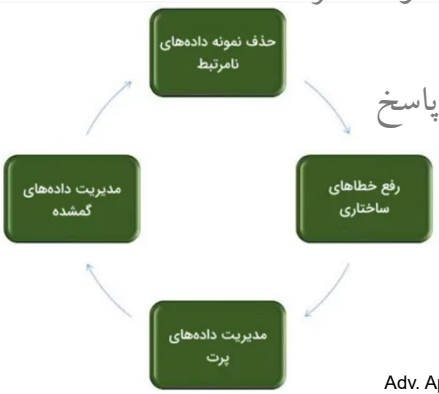
✓ خلاصه دستورهای پاکسازی داده ها

```
# Drop rows with missing values
df_cleaned = df.dropna()

# Fill missing values with a specific value (e.g., 0)
df_filled = df.fillna(0)

# Drop columns
df_dropped = df.drop(columns=['column_name'])

# Rename columns
df_renamed = df.rename(columns={'old_name': 'new_name'})

# Remove duplicates
df_no_duplicates = df.drop_duplicates()

# Convert data types
df['column_name'] = df['column_name'].astype('int')
```



مراحل پاکسازی داده‌ها

- مرحله یک: حذف مشاهدات تکراری یا نامربوط
- مرحله دو : رفع خطاهای ساختاری
- مرحله سه: اصلاح داده‌های پرت ناخواسته مرحله چهار: مدیریت داده‌های گمشده
- مرحله پنج: اعتبارسنجی و پرسش‌وپاسخ



مراحل پاکسازی داده‌ها

مثال: بررسی فایل مسافران قطار

- ✔ وارد کردن کتابخانه‌های لازم
- ✔ بارگذاری مجموعه داده
- ✔ بررسی اطلاعات داده

```
import pandas as pd
import numpy as np

# Load the dataset
df = pd.read_csv('train.csv')
print(df.head())
```

```
   survived  pclass                                               name  ...     fare cabin  embarked
0         0       3                             Braund, Mr. Owen Harris  ...   7.2500   NaN         S
1         1       1  Cumings, Mrs. John Bradley (Florence Briggs Th...  ...  71.2833   C85         C
2         1       3                              Heikkinen, Miss. Laina  ...   7.9250   NaN         S
3         1       1        Futrelle, Mrs. Jacques Heath (Lily May Peel)  ...  53.1000  C123         S
4         0       3                            Allen, Mr. William Henry  ...   8.0500   NaN         S

[5 rows x 11 columns]
```

مراحل پاکسازی داده‌ها

مثال: بررسی فایل مسافران قطار

- ✔ وارد کردن کتابخانه‌های لازم
- ✔ بارگذاری مجموعه داده
- ✔ بررسی اطلاعات داده

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 11 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   survived  891 non-null    int64
 1   pclass    891 non-null    int64
 2   name      891 non-null    object
 3   sex       891 non-null    object
 4   age       714 non-null    float64
 5   sibsp     891 non-null    int64
 6   parch     891 non-null    int64
 7   ticket    891 non-null    object
 8   fare      891 non-null    float64
 9   cabin     204 non-null    object
 10  embarked  889 non-null    object
dtypes: float64(2), int64(4), object(5)
memory usage: 76.7+ KB
```

```
df.info()
```

4 age 714 non-null float64 ←

9 cabin 204 non-null object ←

9

Adv. App. of AI and DT

مراحل پاکسازی داده‌ها

مثال

- ✔ بررسی سطرهای تکراری

```
print(df.duplicated())
```

```
0      False
1      False
2      False
3      False
4      False
       ...
886    False
887    False
888    False
889    False
890    False
Length: 891, dtype: bool
```

10

Adv. App. of AI and DT

مراحل پاکسازی داده‌ها

مثال

✔ دیدن ساختار داده

```
print(df.describe())
```

	survived	pclass	age	sibsp	parch	fare
count	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

11

Adv. App. of AI and DT

مراحل پاکسازی داده‌ها

✔ بررسی ستون‌های دسته‌بندی و عددی

```
# Categorical columns
cat_col = [col for col in df.columns if df[col].dtype == 'object']
print('Categorical columns :',cat_col)
```

```
Categorical columns : ['name', 'sex', 'ticket', 'cabin', 'embarked']
```

```
# Numerical columns
num_col = [col for col in df.columns if df[col].dtype != 'object']
print('Numerical columns :',num_col)
```

```
Numerical columns : ['survived', 'pclass', 'age', 'sibsp', 'parch', 'fare']
```

12

Adv. App. of AI and DT

مراحل پاکسازی داده‌ها

- تعداد کلی مقادیر منحصر به فرد در ستون‌های دسته‌بندی

```
print(df[cat_col].nunique())
```

```
name        891
sex           2
ticket      681
cabin       147
embarked      3
dtype: int64
```

13

Adv. App. of AI and DT

داده‌های پوچ

بسته به نوع داده بایستی تصمیم گیری نمود

- حذف کل سطر داده از داده ها
- قرار دادن داده قبلی یا بعدی به جای داده پوچ
- قرار دادن میانگین داده قبلی و بعدی به جای داده پوچ
- قرار دادن میانگین کل داده‌ها به جای داده پوچ

14

Adv. App. of AI and DT

آنالیز داده‌ها

```
# Calculate correlation matrix
correlation_matrix = df.corr()

# Calculate mean, median, mode(Mode is the most common number)
mean_value = df['column_name'].mean()
median_value = df['column_name'].median()
mode_value = df['column_name'].mode()

# Count unique values
unique_counts = df['column_name'].value_counts()

# Cross-tabulation (Cross tabulation and Chi-square or contingency table is a table to
#  reveal the frequency distribution of the variables)
cross_tab = pd.crosstab(df['column1'], df['column2'])
```


مشاهده داده‌ها

```
import matplotlib.pyplot as plt
import seaborn as sns

# Histogram
df['column_name'].hist()
plt.show()

# Box plot
sns.boxplot(x='category_column', y='numeric_column',
data=df)
plt.show()

# Scatter plot
plt.scatter(df['column1'], df['column2'])
plt.show()

# Heatmap
sns.heatmap(df.corr(), annot=True)
plt.show()
```

روش وان هات

اگر از افراد بپرسند که رنگ مورد علاقه آن‌ها چیست، پاسخ‌های افراد محدود به یک سری گزینه خاص (مثلا: سفید، آبی، قرمز، مشکی ...) خواهد بود.
وضعیت یک ساختمان پس از زلزله با چند گزینه مثل سالم، آسیب جزیی، آسیب کلی، تخریب شده قابل توصیف است. جهت پردازش یک روش استفاده از عدد یک داغ است.

بندر بارگیری چهار مقدار S و C و Q و nan را دارد

	A	B	C	D	E	F	G	H	I	J	K	L
1		survived	pclass	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked
2	0	0	3	Braund, Mr	male	22	1	0	A/5 21171	7.25		S
3	1	1	1	Cumings, M	female	38	1	0	PC 17599	71.2833	C85	C
4	2	1	3	Heikkinen,	female	26	0	0	STON/O2.	7.925		S
5	3	1	1	Futrelle, M	female	35	1	0	113803	53.1	C123	S
6	4	0	3	Allen, Mr. \	male	35	0	0	373450	8.05		S
7	5	0	3	Moran, Mr	male		0	0	330877	8.4583		Q
8	6	0	1	McCarthy,	male	54	0	0	17463	51.8625	E46	S
9	7	0	3	Palsson, M	male	2	3	1	349909	21.075		S
10	8	1	3	Johnson, M	female	27	0	2	347742	11.1333		S
11	9	1	2	Nasser, Mr	female	14	1	0	237736	30.0708		C
12	10	1	3	Sandstrom	female	4	1	1	PP 9549	16.7	G6	S
13	11	1	1	Bonnell, M	female	58	0	0	113783	26.55	C103	S

17

Adv. App. of AI and DT

روش وان هات

در کدبندی وان هات، برای هر برچسب مثل انواع رنگ یک ستون مجزا در نظر گرفته می‌شود و برای هر نمونه فقط در ستون رنگ مربوطه یک قرارداده می شود.

Color
Red
Red
Yellow
Green
Yellow

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

تابع unique مقادیر منحصر به فرد را می دهد

18

Adv. App. of AI and DT

روش وان هات

تابع unique مقادیر منحصر به فرد را می دهد

```
# Determin unique data
print(df['embarked'].unique())
```

خروجی

```
['S' 'C' 'Q' nan]
```

در این مثال مقادیر پوچ را لازم نداریم و داده‌های مربوطه را می توانیم حذف کنیم

```
# Drop houses where the target is missing
df.dropna(axis=0, subset=['embarked'], inplace=True)
target = df.embarked
print(target.unique())
```

خروجی

```
['S' 'C' 'Q']
```

19

Adv. App. of AI and DT

روش وان هات

```
# One hot encoding
one_hot_encoded_embarked = pd.get_dummies(df,
columns = ['embarked'])
print(one_hot_encoded_embarked)
```

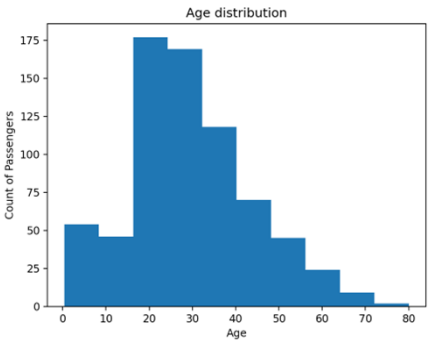
	survived	pclass	name	sex	...	cabin	embarked_C	embarked_Q	embarked_S
0	0	3	Braund, Mr. Owen Harris	male	...	NaN	False	False	True
1	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	...	C85	True	False	False
2	1	3	Heikkinen, Miss. Laina	female	...	NaN	False	False	True
3	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	...	C123	False	False	True
4	0	3	Allen, Mr. William Henry	male	...	NaN	False	False	True
..	...	...	...	...	...	...	...	...	...
886	0	2	Montvila, Rev. Juozas	male	...	NaN	False	False	True
887	1	1	Graham, Miss. Margaret Edith	female	...	B42	False	False	True
888	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	...	NaN	False	False	True
889	1	1	Behr, Mr. Karl Howell	male	...	C148	True	False	False
890	0	3	Dooley, Mr. Patrick	male	...	NaN	False	True	False

20

Adv. App. of AI and DT

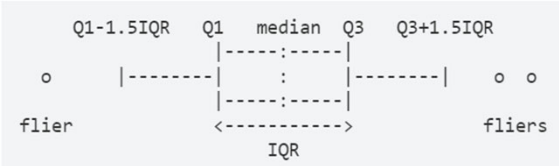
رسم یکی از داده ها

```
# example for plot - age
fig = plt.figure()
ax = fig.add_subplot(111)
ax.hist(df['age'], bins = 10, range =
(df['age'].min(),df['age'].max()))
plt.title('Age distribution')
plt.xlabel('Age')
plt.ylabel('Count of Passengers')
plt.show()
```



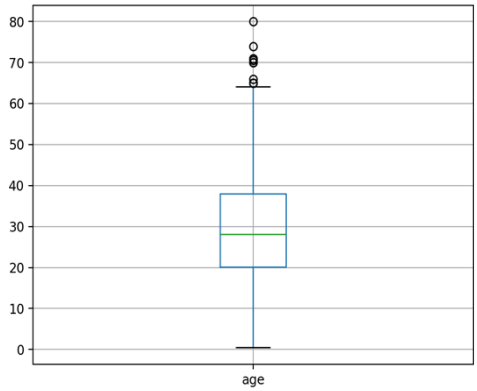
Boxplot

- The box extends from the first quartile (Q1) to the third quartile (Q3) of the data, with a line at the median. The whiskers extend from the box to the farthest data point lying within 1.5x the inter-quartile range (IQR) from the box. Flier points are those past the end of the whiskers.



رسم جعبه داده

```
# example of Box Plots raw data
df.boxplot(column='age',return_type='axes')
```



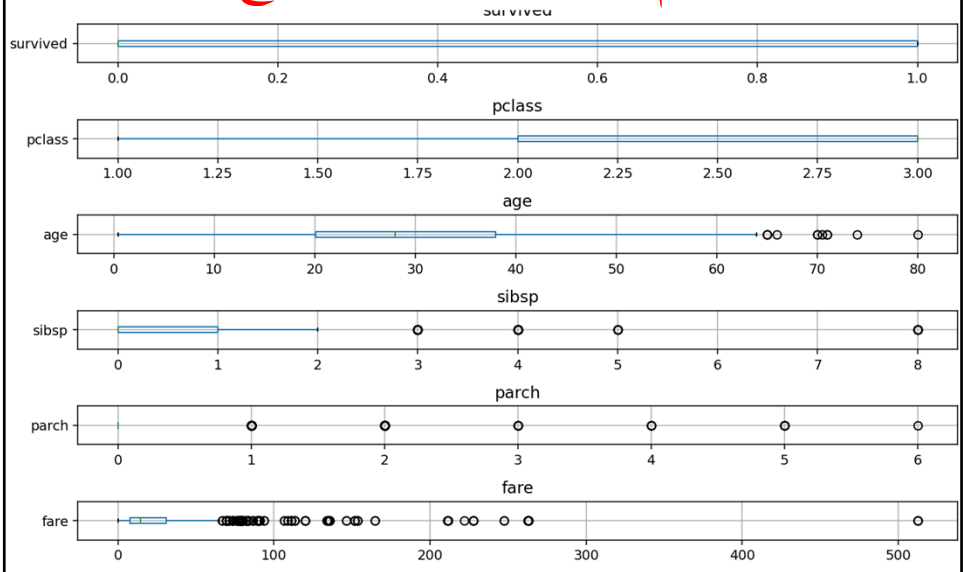
Adv. App. of AI and DT 23

رسم جعبه داده قبل از اصلاح

```
# Create subplots: 1 plot per column
fig, axs = plt.subplots(nrows=len(num_col), , dpi=80, figsize=(10,
6))

# Ensure axs is iterable even if there is only one column
if len(num_col) == 1:
    axs = [axs]

# Plot each numerical column in a separate subplot
for i, column in enumerate(num_col):
    df.boxplot(column=column, ax=axs[i], vert=False)
    axs[i].set_title(column)  # Set the title as the column name

plt.tight_layout()  # Adjust layout to prevent overlap
plt.show()
```

Adv. App. of AI and DT 24

رسم جعبه داده قبل از اصلاح



پاکسازی داده ها

```
# data cleaning
# Identify the quartiles
i = 0
for col in num_col:
    q1, q3 = np.percentile(df[col], [25, 75])
# Calculate the interquartile range
    iqr = q3 - q1
# Calculate the lower and upper bounds
    lower_bound = q1 - (1.5 * iqr)
    upper_bound = q3 + (1.5 * iqr)
# Drop the outliers
    clean_data = df[(df[col] >= lower_bound)
                & (df[col] <= upper_bound)]
    i+=1
```

اصلاح داده ها

```
print(df.describe())
```

	survived	pclass	age	sibsp	parch	fare
count	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

```
print(clean_data.describe())
```

	survived	pclass	age	sibsp	parch	fare
count	775.000000	775.00000	613.000000	775.000000	775.000000	775.000000
mean	0.339355	2.48000	28.946574	0.437419	0.340645	17.822091
std	0.473796	0.73439	14.368139	0.899838	0.785914	13.578085
min	0.000000	1.00000	0.420000	0.000000	0.000000	0.000000
25%	0.000000	2.00000	20.000000	0.000000	0.000000	7.895800
50%	0.000000	3.00000	28.000000	0.000000	0.000000	13.000000
75%	1.000000	3.00000	37.000000	1.000000	0.000000	26.000000
max	1.000000	3.00000	80.000000	5.000000	6.000000	65.000000

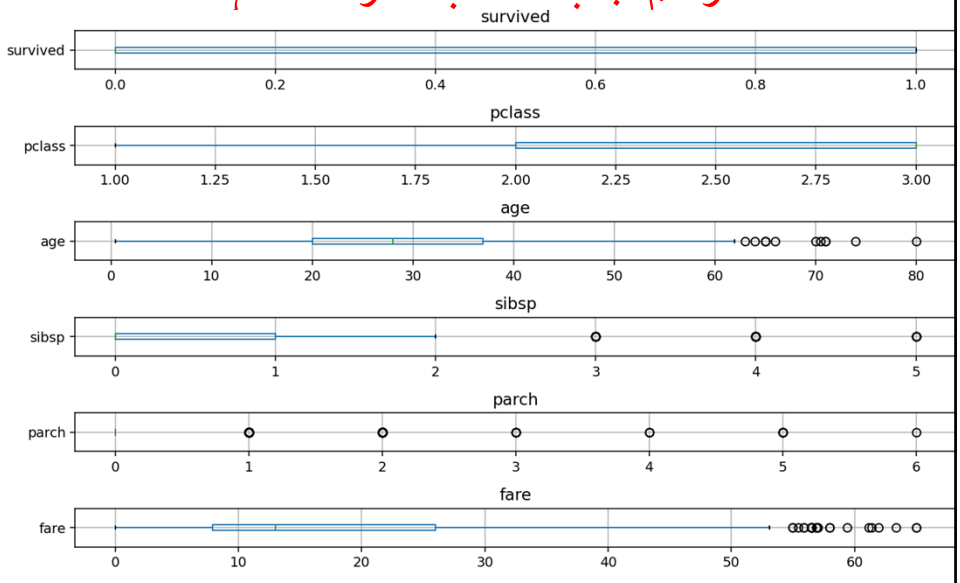
Adv. App. of AI and DT 27

رسم جعبه داده بعد از اصلاح

```
# Create subplots: 1 plot per column
fig, axs = plt.subplots(nrows=len(num_col), ncols=1, , dpi=80,
figsize=(10, 6))

# Ensure axs is iterable even if there is only one column
if len(num_col) == 1:
    axs = [axs]

# Plot each numerical column in a separate subplot
for i, column in enumerate(num_col):
    clean_data.boxplot(column=column, ax=axs[i], vert=False)
    axs[i].set_title(column)  # Set the title as the column name

plt.tight_layout()  # Adjust layout to prevent overlap
plt.show()
```

Adv. App. of AI and DT 28

رسم جعبه داده بعد از اصلاح



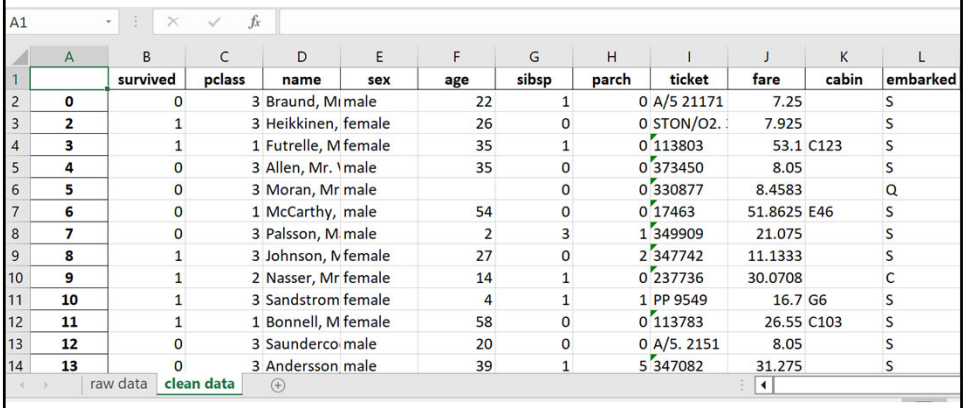
همبستگی بین داده‌ها

```
#correlation
numerical_data =
df.select_dtypes(include=[np.number])
print(numerical_data)``
corr = numerical_data.corr()
plt.figure(dpi=130)
sns.heatmap(corr, annot=True, fmt= '.2f')
plt.show()
```

نوشتن داده ها در فایل اکسل

```
# write raw and clean data to excel file
# Specify a writer
writer = pd.ExcelWriter('train_clean_data.xlsx',
engine='xlsxwriter')
# Write your DataFrame to a file
df.to_excel(writer, 'raw data')
clean_data.to_excel(writer, 'clean data')
# Save the result
writer.close()
```


نوشتن داده ها در فایل اکسل



		survived	pclass	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked
2	0	0	3	Braund, Mr	male	22	1	0	A/5 21171	7.25		S
3	2	1	3	Heikkinen,	female	26	0	0	STON/O2.	7.925		S
4	3	1	1	Futrelle, M	female	35	1	0	113803	53.1	C123	S
5	4	0	3	Allen, Mr.	male	35	0	0	373450	8.05		S
6	5	0	3	Moran, Mr	male		0	0	330877	8.4583		Q
7	6	0	1	McCarthy,	male	54	0	0	17463	51.8625	E46	S
8	7	0	3	Palsson, M	male	2	3	1	349909	21.075		S
9	8	1	3	Johnson, M	female	27	0	2	347742	11.1333		S
10	9	1	2	Nasser, Mr	female	14	1	0	237736	30.0708		C
11	10	1	3	Sandstrom	female	4	1	1	PP 9549	16.7	G6	S
12	11	1	1	Bonnell, M	female	58	0	0	113783	26.55	C103	S
13	12	0	3	Saunderco	male	20	0	0	A/5. 2151	8.05		S
14	13	0	3	Andersson	male	39	1	5	347082	31.275		S

raw data clean data

تمرین برنامه نویسی

تمرین سوم : یک برنامه به زبان پایتون بنویسید که یک فایل داده را بگیرد و داده ها پرت را تعیین کند.

Depth (m)	Temperature (c)
0.0	4.3
2.7	4.3
5.4	4.3
8.2	4.4
10.9	4.4
13.7	4.4
16.4	4.5
19.2	4.5
21.9	4.6
24.7	4.6
27.4	4.7
30.2	4.7
32.9	4.8
35.7	4.9

۱- داده ها را رسم کنید

۲- تعداد داده های پرت را بیابید

۳- داده ها پرت را حذف کنید

۴- از روش وان هات طبقه بندی داده انجام دهید

۵- داده های ورودی، خروجی و گرافها را در شیت های مختلف فایل اکسل ذخیره نمایید