```
 # This is formatted as code
```

DHV LAB Sheet 7

```
#Import Python Libraries
import numpy as np
import scipy as sp
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Enable inline plotting
%matplotlib inline
```

Pandas is a python package that deals mostly with :

- **Series** (1d homogeneous array)
- **DataFrame** (2d labeled heterogeneous array)
- **Panel** (general 3d array)

Pandas Series

Pandas *Series* is one-dimentional labeled array containing data of the same type (integers, strings, floating point numbers, Python objects, etc.). The axis labels are often referred to as *index*.

```
# Read a dataset with missing values
flights = pd.read_csv("flights.csv")
flights.head()
```

```
flights.info()
```

Missing Values

```
# Select the rows that have at least one missing value
flights[flights.isnull().any(axis=1)].head()
```

```
# Filter all the rows where arr_delay value is missing:
```

```
flights1 = flights[ flights['arr_delay'].notnull( )]
flights1.head()


# Remove all the observations with missing values
flights2 = flights.dropna()


# Fill missing values with zeros
nomiss =flights['dep_delay'].fillna(0)
nomiss.isnull().any()
```

Exercise

```
# Count how many missing data are in dep_delay and arr_delay columns
```

Common Aggregation Functions:

Function	Description
min	minimum
max	maximum
count	number of non-null observations
sum	sum of values
mean	arithmetic mean of values
median	median
mad	mean absolute deviation
mode	mode
prod	product of values
std	standard deviation
var	unbiased variance

```
# Find the number of non-missing values in each column
flights.describe()


flights.info()


# Find mean value for all the columns in the dataset
flights.min()


# Let's compute summary statistic per a group':
flights.groupby('carrier')['dep_delay'].mean()


# We can use agg() methods for aggregation:
flights[['dep_delay','arr_delay']].agg(['min','mean','max'])
```

```
# An example of computing different statistics for different columns
flights.agg({'dep_delay':['min','mean',max], 'carrier':['nunique']})
```

Basic descriptive statistics

Function	Description
min	minimum
max	maximum
mean	arithmetic mean of values
median	median
mad	mean absolute deviation
mode	mode
std	standard deviation
var	unbiased variance
sem	standard error of the mean
skew	sample skewness
kurt	kurtosis
quantile	value at %

```
# Convinient describe() function computes a veriety of statistics
flights.dep_delay.describe()
```

```
# find the index of the maximum or minimum value
# if there are multiple values matching idxmin() and idxmax() will return the first match
flights['dep_delay'].idxmin()  #minimum value
```

```
# Count the number of records for each different value in a vector
flights['carrier'].value_counts()
```

```
#factorplot
sns.catplot(x='carrier',y='dep_delay', data=flights, kind='bar')
```

Exercise

```
#Using seaborn package explore the dependency of arr_delay on dep_delay (scatterplot or re
```

```
#Use matplotlib to draw a histogram of a salary data
plt.hist(df['salary'],bins=20, density=True)
```

```
# Use regular matplotlib function to display a barplot
df.groupby(['rank'])['salary'].count().plot(kind='bar')
```

```
# Use seaborn package to display a barplot
sns.set_style("whitegrid")

ax = sns.barplot(x='rank',y ='salary', data=df, estimator=len)



# Split into 2 groups:
ax = sns.barplot(x='rank',y ='salary', hue='sex', data=df, estimator=len)


#Violinplot
sns.violinplot(x = "salary", data=df)


#Scatterplot in seaborn
sns.jointplot(x='service', y='salary', data=df)


sns.scatterplot(x='service', y='salary', data=df)


#If we are interested in linear regression plot for 2 numeric variables we can use regplot
sns.regplot(x='service', y='salary', data=df)


# box plot
sns.boxplot(x='rank',y='salary', data=df)


# side-by-side box plot
sns.boxplot(x='rank',y='salary', data=df, hue='sex')


# swarm plot
sns.swarmplot(x='rank',y='salary', data=df)


# Pairplot
sns.pairplot(df)
```

Colab paid products - Cancel contracts here

0s completed at 10:03 AM