Numerical Methods

Math 3338 – Spring 2022

Worksheet 27

Pandas

1 Overview

What is Pandas?[1] Pandas is a data manipulation tool that has gained popularity due to its ease of use and expandability.

Spyder is built to use Pandas naturally and fluidly. If you aren't using Spyder, good luck. We'll be transitioning from the text editor to the iPython terminal quite frequently. If you don't have Spyder, this will be far more difficult.

2 Pandas - A Simple Example

Let's say we have a data set that we'd like to analyze. For now that will be "housing.csv" on Canvas. This is a dataset about housing prices in California. Here is the code I would enter to load the data into Python,

```
import pandas as pd
import numpy as np

data = pd.read_csv("data/housing.csv")
```

This doesn't print anything... Go to the interpreter (the thing where graphs and output display), type `data` and hit enter. You should have seen a large display of data. Scroll up to see some of the column IDs. This doesn't show all the columns, to see what they are type,

```
data.columns
```

The columns "median_housing_value" and "ocean_proximity" look interesting. Let's make a boxplot using these variables. Using Logic, it would make sense that closer to the ocean leads to higher housing values. Lets find out,

```
data.boxplot("median_house_value",by = "ocean_proximity")
```

The boxplot does verify our suspicions, at least a little. There are a ton of outliers in the "INLAND" category.

3 Working with Data

The core data structure in Pandas is called a DataFrame. Think of a dataframe almost like a spreadsheet, it has rows and columns. You should organize data so that each column is a Variable and each row is an Observation.

The housing data is already like this. Each column is a different property (or variable) and each row represents a neighborhood (an observation). Typically data is not this nice and needs manipulating to make look nice. Pandas has many, many methods to help you do this. We're skipping it all.

Extracting columns is similar to using a dictionary. For example, say we wanted to know the average number of bedrooms. We just need to type,

[1] It's an endangered bear native to China, but that's not important right now

```
data['total_bedrooms'].mean()
```

and we see the mean is 537.8. To see even more statistics type,

```
data['total_bedrooms'].describe()
```

Of course you could just type

```
data.describe()
```

but this may be too much useless info (like latitude/longitude).

You can select multiple columns,

```
data[['total_bedrooms','population']].describe()
```

Notice you need to put a list inside the selection. That's because it's usually used like this,

```
columns = ['total_bedrooms','population']
data[columns].describe()
```

Suppose you wanted to see the average number of people in each household in a given region. To get that you would take the population divided by the number of households. This is easily done in Pandas using array vectorization.

```
data['density'] = data['population']/data['households']
```

4 Graphing and Filtering

There are many, many ways to plot in Pandas. The easiest is using `data.plot`. In the interpreter type `data.plot.` and press tab. You'll see a bunch of plotting options. Let's make a scatterplot with population on the x-axis and median house value on the y-axis.

```
data.plot.scatter("population","median_house_value")
```

This is equivalent to,

```
data.plot("population","median_house_value",'scatter')
```

Both methods are valid and both work and both have advantages/disadvantages. Remember you can view method syntax by typing `data.plot?` in the interpreter.

Let's say you only wanted to see the above plot for properties that are classified as being near the bay. Pandas is built on top of Numpy, you can use array filtering and slicing naturally.

```
X = data[data['ocean_proximity'] == "NEAR BAY"]
X.plot.scatter("population","median_house_value")
```

You should also view X. You'll see all the ocean proximities are near the bay.

Numerical Methods

Math 3338 – Spring 2022

Homework 27 (Due: Tuesday, April 26)

Include all graphs in your write up of the problems.

Problem 1 (1 pt) In the "housing.csv" dataset, how many regions are in each category of the "ocean_proximity" column?

Problem 2 (1 pt) Create a "density" column (we did this in the worksheet, that you read and understood). Make a table of the descriptive statistics and a boxplot of this column. Describe what you see and speculate what is happening.

Problem 3 (1 pt) You should have noticed something bizarre in the previous problem. Use array filtering to find the largest density outlier. What California city is this data point closest to?

Problem 4 (1 pt) Do something similar to the "density" column except compare median income and median house value. Try to figure out what's happening here.

Problem 5 (1 pt) Explore the dataset further. Make several graphs and describe what you see and find.