Lista de comandos para análise de dados em python

Para entender o Jupyter

- Dois modos
  1. modo edição: escreve em uma célula
  2. modo comando: movimenta as células

- Dois tipos de células
  1. código (esc, depois y)
  2. markdown (esc, depois m)

esc - entra em modo comando (azul)
enter - entra em modo edição (verde)

Modo comando (azul)

m - muda célula para markdown
y - muda célula para python

a - cria célula acima (*above*)
b - cria célula abaixo (*bellow*)

v - cola célula
c - copia célula
x - recorta/apaga célula
f - pesquisar e substituir palavras
l - mostra números das linhas
h - lista de atalhos
00 (zero duas vezes) - reinicia o notebook

SHIFT - M: mescla células
CTRL + SHIFT + (sinal de menos): divide célula

Modo edição (verde)

SHIFT-ENTER - executa os comandos que estão dentro da célula
objeto.<TAB> - sugestões de métodos

- transforma a linha em comentário, isto é, algo que não será executado.
%matplotlib inline - para que os gráficos apareçam no documento (um resquício de versões antigas)
import pandas as pd - carrega algum pacote (p. ex., pandas) como alguma sigla (p. ex., pd)

Listas
x = [209, 32, 4, 23, 45] - cria um objeto chamado x contendo uma lista com os números 209, 32, 4, 23 e 45.
Atenção: o python começa a contar em zero.
x[0] - mostra o primeiro elemento da lista (209), x[1] mostra o segundo elemento (32) etc.
x[1:4] - mostra os números nas posições 1, 2, 3 (32, 4 e 23)
x[:2] - mostra todos os elementos até o terceiro (209 e 32), *não* incluindo o terceiro (intervalo aberto).
x[3:] - mostra todos os elementos a partir do quarto (23 e 45), incluindo o quarto.
x[:] - mostra todos os elementos da lista.

len(x) - mostra o tamanho de x. Nesse caso, 3.
sum([209, 32, 4, 23 e 45]) ou sum(x) - soma os valores.
min(x) e max(x) - mostra os valores mínimos e máximos.

Operadores
3 * 2 - multiplicação
6 / 3 - divisão
2 ** 3 - potenciação (p. ex, dois elevado a três)
= - atribuição (p. ex., x recebe 3)
== - teste de igualdade (p. ex., 'município == "Varginha"' testa se o município é Varginha).
!= - teste de desigualdade (p. ex., 'município != "Varginha"' testa se o município não é Varginha.
& - e
| - ou
>=, <= - maior ou igual que, menor ou igual que

Ler dados e mostrar informações básicas
pnad = pd.read_csv('pnad022017.csv') - lê o arquivo csv e o transforma em um objeto dataframe chamado pnad
pnad = pd.read_excel('pnad022017.xls') - lê o arquivo do Excel (xls ou xlsx) e o transforma em um objeto dataframe.
pnad.shape - mostra o número de linhas e colunas do dataframe. Obs: nesse caso não é preciso abrir e fechar parêntesis.
pnad.columns - mostra o nome das colunas (variáveis). Obs: nesse caso não é preciso abrir e fechar parêntesis.
pnad.head(10) - mostra as primeiras dez linhas do dataframe, o padrão é mostrar 5.
pnad.tail(10) - mostra as últimas dez linhas do dataframe.
pnad.info() - mostra os tipos das variáveis e o tamanho do dataframe na memória.
pnad.loc[:, ['sexo', 'idade', 'renda']] - *subseleção* por rótulos (*labels*), mostra todas as linhas, apenas das variáveis indicadas.
pnad.iloc[:10, [3, 4, 5]] - *subseleção* por posição (i). No exemplo, mostra apenas as primeiras das variáveis nas posições indicadas.
Atenção: note o uso de dois pares de colchetes em .loc e .iloc. O primeiro deles deriva do fato de que funções usam colchetes ao invés pois são subseleções, como no caso das listas explicado acima. O segundo par de colchete é usado quando queremos indicar mais de um elemento, situação em que é preciso criar uma lista.

Estatísticas básicas
pnad.describe() - mostra o resumo estatístico (mínimo, máximo, média, mediana e quartis) de todas as variáveis do dataframe.
pnad.min() - mostra o valor mínimo de todas as variáveis. O mesmo pode ser feito com todos os comandos abaixo.
pnad.renda.describe() - mostra o resumo estatístico da variável selecionada.
pnad.renda.min() - mostra a observação com a menor renda.
pnad.renda.max() - a maior renda.
pnad.renda.mean() - a média da renda.
pnad.renda.std() - o desvio padrão da renda.
pnad.renda.var() - a variância da renda.
pnad.renda.mad() - o desvio absoluto médio da renda.
pnad.renda.sum() - o somatório da renda.
pnad.renda.cumsum() - a soma acumulada da renda.

pnad.cov( ) - a matriz de covariâncias
pnad.corr( ) - a matriz de correlações.
pnad.loc[:, [‘renda’, ‘idade’, ‘estudo’]].corr() - a correlação entre a renda, a idade e anos de estudo.

Operações com dataframes
pd.ocup_cod.unique() - mostra todos os valores não repetidos da variável ocup_cod (código da ocupação).
pd.sexo.value_counts() - mostra quantas observações há em cada um valores da variável sexo.
pnad.query(‘renda > 10000’) - seleciona apenas as observações com renda maior do que 10 mil.
pnad.query(‘renda > 10000 & idade < 60 & sexo == 2’) - apenas observações com renda maior do que 10 mil, idade menor do que 60 e que sejam mulheres (o sexo feminino foi codificado como 2).
pnad.groupby('sexo').idade.mean() - *agrupa* as observações por sexo e mostra a média de idade de cada um dos sexos.
pnad.sort_values(by='renda', ascending=False) - ordena as observações por renda, em ordem decrescente.

Ajuda

SHIFT-TAB quando o curso estiver em algum comando. Segure o SHIFT e vá apertando o TAB até quatro vezes para ir aumentando o tamanho da ajuda.
pnad.head? - o mesmo que apertar SHIFT-TAB quatro vezes, mostra a documentação do comando .head().
pnad.head?? - mostra o código usado pela função .head().