

Presentación del software Python

Damián Silva Fernández

Universidad de Buenos Aires

July 2019

- 1 Motivación
- 2 Discusiones al respecto
- 3 Recursos
- 4 Software e IDE
- 5 Cronograma de Clases

Contexto

- Crecimiento de la disponibilidad de datos
- Desarrollo de modelos de aprendizaje estadístico
- Mejoras en la capacidad de cómputo y métodos de cómputo numérico

Disponibilidad de datos

- Gran cantidad de series macroeconomicas
- Texto en los medios, documentos de organismos, redes sociales
- Información sobre búsquedas de información (Google Trends)
- Información proveniente de imagenes, posicionamiento....

Modelos

- Estadística Clásica
- Modelo Lineal
- Modelo Lineal Múltiple
- Econometría Dinámica (Modelos VAR, VEC, TAR)
- Economía de Grandes Datos (Lasso, Ridge, Elastic Net)
-

Tecnología de la información

- Capacidad de cómputo
- Digitalización de la información
- Recolección de la información (Webscrapping)
- Métodos de optimización numérica

Temas recurrentes e interconectados

- Definición del objetivo (pronóstico, interpretación, medición de la causalidad, ...)
- Incertidumbre sobre el modelo (Model uncertainty). Combinación de modelos.
- Sobreajuste (overfitting). Regularización (tuning)
- Rol de la teoría / conocimiento del experto

Objetivos del curso

- Reconocer y entrenar modelos de aprendizaje estadístico.
- Entender conceptos sobre bases de datos.
- Adquirir herramientas de programación en el lenguaje Python.
- Utilizar esas herramientas relevantes en economía.

Recursos

- Link de instalación del software Python: *Anaconda Distribution*
- Página Web: *Página del curso*
- Link del repositorio Github: *Repositorio Github*

IDE Spyder

Arquivo Editar Pesquisar Código Executar Depurar Consoles Projetos Ferramentas Ver Ajuda

Editor - C:\Users\jduc\Desktop\barra_numerico.py

```

147 desperdicio = l - sum(Bin[l])
148 desp.append(desperdicio)
149
150 return Bin, len(Bin), sum(desp)
151
152 def waste(Bin,l):#Calcular a perda de um vetor Bin
153     perda = []
154     for aux in Bin:
155         perda+= [l - sum(aux)]
156     return perda
157
158 def graphic_bars(Bin,l,perda):#Constrói o Gráfico das barras
159     pos= arange(len(Bin)) +0.5
160     value = []
161     ticks = []
162     for aux in Bin:
163         value+=sum(aux)
164         for i in range(len(value)):
165             ticks +=['Barra '+ str(i + 1)]
166     plt.yticks(pos,ticks)
167     plt.ylim(0,len(pos)+0.2)
168     plt.xlim(0,l+0.2)
169     plt.barh(pos, value, align='center', color= '#8394f5')
170     trans = [l]*len(ticks)
171     pos= arange(len(Bin)) +0.1
172     plt.barh(pos,trans, linestyle = '--', alpha=0.3, color='#80C4DE')
173     plt.xlabel('Tamanhos das Barras')
174     plt.grid(True)
175     plt.title('Desperdício Máximo: %i'%sum(perda))
176     return plt.show()
177
178 print(nf(itens,l))
179 print(ff(itens,l))
180 print(bf(itens,l))
181 print(crescente(itens,l))
182 print(decrecente(itens,l))
183

```

Console [Python]

```

...: [10],
...: [10, 2],
...: [13],
...: [4, 9],
...: [5, 2, 11]].l,[2, 2, 4, 5, 5, 4, 7, 7, 5, 3, 2, 2, 7]]

```

Desperdício Máximo: 55

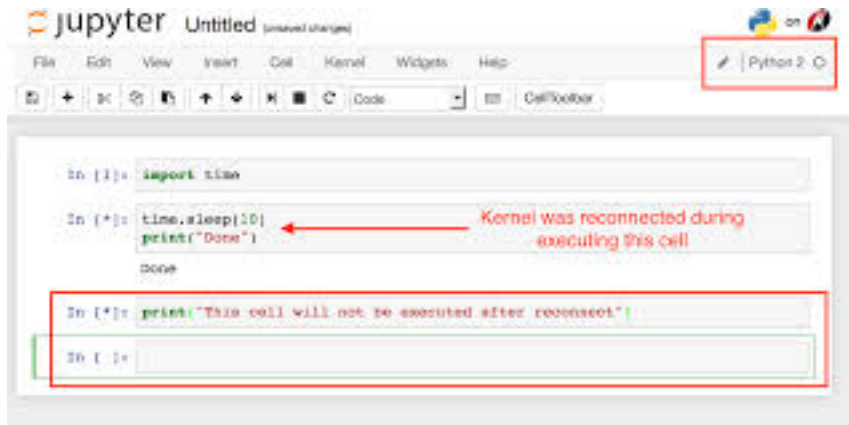
```

In [10]: itens
Out[10]: [4, 9, 4, 2, 9, 10, 10, 11, 8, 8, 10, 10, 2, 13, 4, 9,
5, 2, 1]
In [11]:

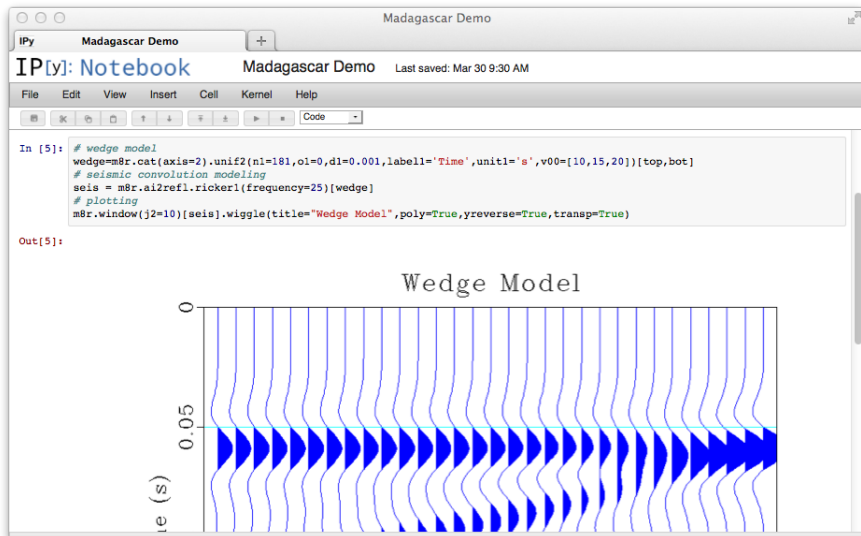
```

Permissões: RW Fim de linha: CRLF Codificação: UTF-8 Linha: 155 Coluna: 31 Memória: 74 %

Jupyter Notebook



Ipython



Unidades Tematicas

- Clase 1: Uso del lenguaje Python como calculadora
- Clase 2: Estructuras de datos
- Clase 3: Entrenamiento y estimación de modelos estadísticos
- Clase 4: Visualización gráfica
- Clase 5: Herramientas de Economía de Grandes Datos

VAMOS A LA APLICACIÓN!!!