Procesing big data with Vaex, dataframes and no clusters

Marco Carranza

@mcrrnz

Agenda

- Intro
- ¿Qué es Vaex ?
- Similitudes y diferencias con Pandas
- Algunas características interesantes
- Limitaciones
- Demo - Analizando un archivo grande
  (+140 000 000 registros y +24GB)

Marco Carranza

Co-founder Teamcore Solutions

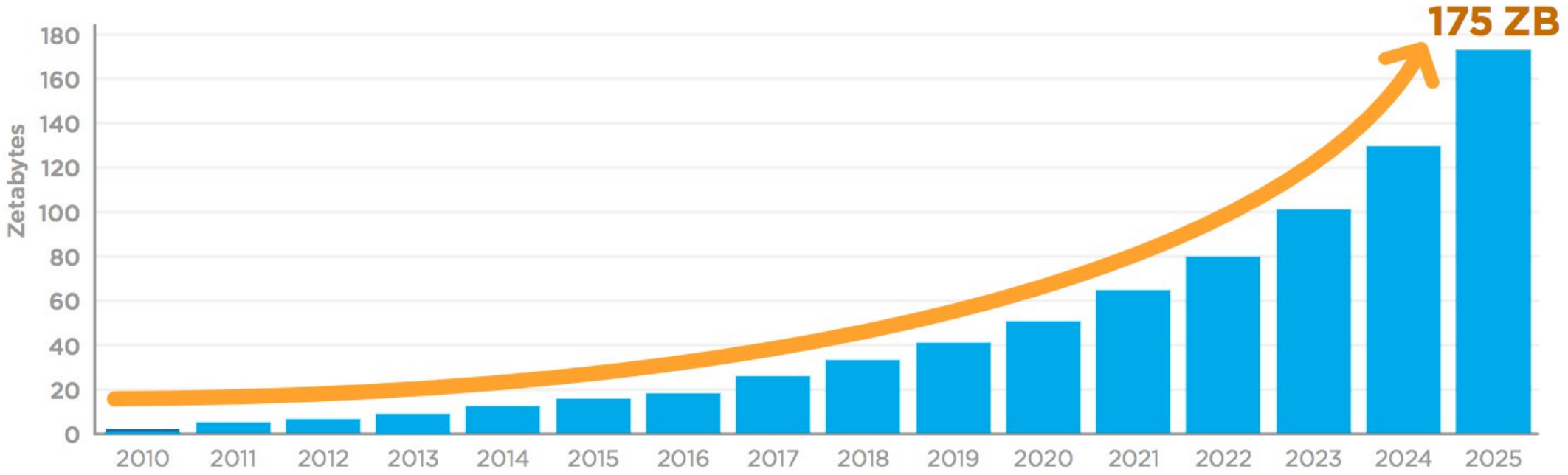
Tech Lead

@mcrrnz

https://github.com/marcocarranza/pyconar19_vaex

La data hoy

- Hoy en día, los datos son cada vez más grandes, lo que hace que sea casi imposible procesarlos en máquinas de escritorio.



Source: IDC Data Age 2025

https://github.com/marcocarranza/pyconar19_vaex

¿Qué tan grandes son los datasets?

- Datasets de 20GB, 50GB, 100 GB son cada vez más comunes.
- Una resonancia magnética genera 20 000 imágenes
- En instagram se suben 54 000 imágenes por minuto.
- Un vehículo autónomo genera 11 Terabytes de data al día
- En Twitter se postean 3 000 tweets por segundo
- Son fácilmente almacenables en el disco duro, pero difíciles de procesar en memoria.

https://github.com/marcocarranza/pyconar19_vaex

Algunas estrategias para procesar la data

- Generar una muestra de la data, pero se corre el riesgo de omitir insights.
- Utilizar la computación distribuida, existen muchos frameworks como Skpark, Hadoop, Presto, Dask, etc. Esto implica un ovehead ya que es necesario configurar y administrar los clusters.
- Buscar una instancia con muchos recursos. Por ejemplo AWS tiene instancias de con varias Teras de ram y usar librerías como Pandas.

https://github.com/marcocarranza/pyconar19_vaex

¿Qué es Vaex ?

- Vaex es una librería, similar a pandas que permite visualizar, explorar y analizar data tabular, siempre que esta se pueda guardar en el disco duro.

#	x	y	z	vx	vy	vz
0	-0.777470767	2.10626292	1.93743467	53.276722	288.386047	-95.2649078
1	3.77427316	2.23387194	3.76209331	252.810791	-69.9498444	-56.3121033
2	1.3757627	-6.3283844	2.63250017	96.276474	226.440201	-34.7527161
3	-7.06737804	1.31737781	-6.10543537	204.968842	-205.679016	-58.9777031
4	0.243441463	-0.822781682	-0.206593871	-311.742371	-238.41217	186.824127
...	...	...	...	...	...	...
329995	3.76883793	4.66251659	-4.42904139	107.432999	-2.13771296	17.5130272
329996	9.17409325	-8.87091351	-8.61707687	32.0	108.089264	179.060638
329997	-1.14041007	-8.4957695	2.25749826	8.46711349	-38.2765236	-127.541473
329998	-14.2985935	-5.51750422	-8.65472317	110.221558	-31.3925591	86.2726822
329999	10.5450506	-8.86106777	-4.65835428	-2.10541415	-27.6108856	3.80799961

https://github.com/marcocarranza/pyconar19_vaex

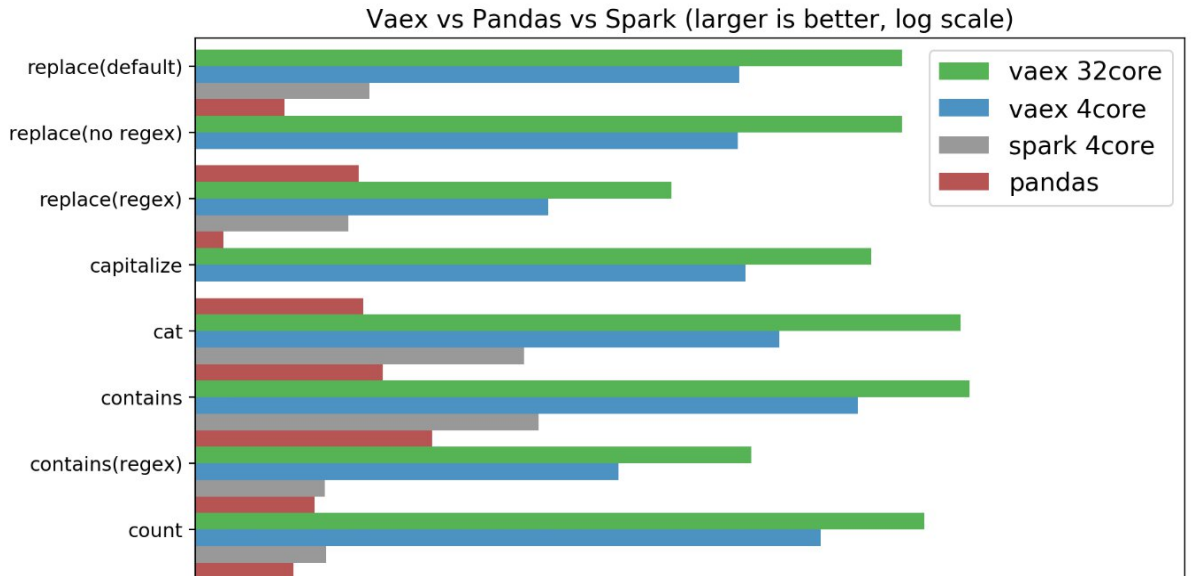
Algunas catacterísticas

- Performance capaz de procesar mas de 1 billon de filas/sec
- Columnas Virtual que ejecutan cálculos sobre la marcha, sin necesidad de desperdiciar memoria.
- Uso eficiente de la memoria: Se evita hacer copias en memoria al hacer filtros, selecciones, subconjunto de datos.
- Soporte para hacer visualizaciones.
- Integración con Jupyter.
- Sintaxis similar a Pandas.
- Buen performance para graficar datasets muy grandes.

https://github.com/marcocarranza/pyconar19_vaex

Algunas catacterísticas

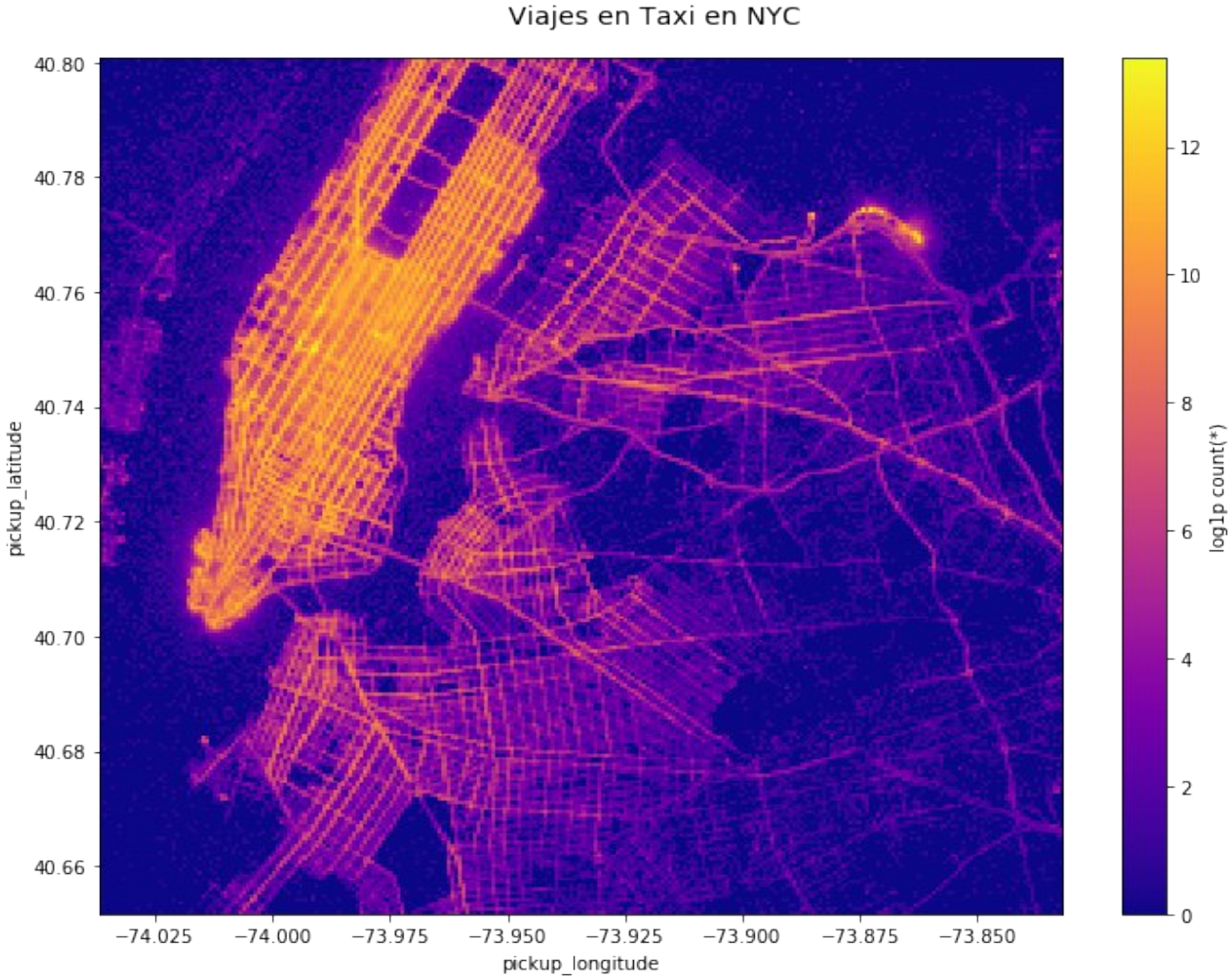
- Strings se procesan en c++ para evitar los bloqueos del GIL y utilizar todos los cores.
- Arrow permite un uso eficiente de la memoria.



Source: https://towardsdatascience.com/vaex-a-dataframe-with-super-strings-789b92e8d861

https://github.com/marcocarranza/pyconar19_vaex

Demo



https://github.com/marcocarranza/pyconar19_vaex

Recomendaciones / Conclusiones

- Es posible abrir archivos CSV mas grandes que la memoria sin que falle, pero el desempeño no es bueno.
- Transformar la data a un soporte Mapeo de memoria (apache arrow, hdf5,etc)
- Vaex puede ser muy útil para trasformar, limpiar data (ETL) en entornos cloud cuando el tamaño de la data en muy variable.
- Vaex permite visualizar fácilmente la data para detectar outliers con pocos recursos y líneas de código.

https://github.com/marcocarranza/pyconar19_vaex

¡Muchas gracias!

- Vaex :

  http://docs.vaex.io/en/latest/index.html

- Slides & Notebook :
  https://github.com/marcocarranza/pyconar19_vaex

- Twitter: @mcrrnz