**2019.11.15 INTRO + GIT + COMMAND LINE**

Proyecto de master serán unas 50-80 horas.

**The data science process:**

* Ask and interesting question.
* Get the data.
* Explore the data.
* Model the data: darle forma para que coincida con la manera en que quieres analizar los datos.
* Communicate and visualize the results.

**Qué tipo de preguntas podemos resolver?**

MACHINE LEARNING & STATISTICS APPLICATIONS:

* Supervised Learning: classification, regression, ranking
* Unsupervised learning: clustering, association mining, segmentation, dimension reduction
* Reinforcement learning: decision process, rewards system, recommendation systems. (**No lo vemos en el master).**

**NOTA**: Subir código a Github todas las semanas y compartir para crear portfolio y tenerlo ahí cuando vayas a buscar trabajo.

Hay que poner el enlace de github en el CV.

Es importante que se vea un commit semanal, no hacerlo todo al final. Subir lo que se dé cada semana en clase a Git para que el recruiter vea todo el trabajo hecho en meses.

Ver blog/post de Amadeus data scientist

<https://amadeus.com/en/insights/blog/ten-things-to-look-for-when-hiring-a-data-scientist>

<http://www.amadeus.com/blog/tag/data-scientist/>

<https://www.avanzaentucarrera.com/orientacion/profesiones/digitales/>

cd ..

mkdir : make directory

mkdir git: crear directorio llamado dir

cd : change directory ir al directorio home

cd /home/angelrps: me lleva a la ruta marcada.

cd .. subir un nivel de carpetas

cd ../.. : subir dos niveles de carpetas

ls: list, listar los archivos que hay en la carpeta en la que te encuentras

pwd: print working dir, muestra la ruta en la que te encuentras

echo: mostrar en pantalla un texto o una variable

cat README.md : muestra lo que contiene el archivo README.md dentro

TRABAJANDO CON GIT EN LINEA DE COMANDOS

Primero crear repositorio nuevo desde Github

Crear ese directorio en nuestra carpeta local (mkdir nombrerepositorio)

**git init** : convierte la carpeta actual en un repositorio git

**git remote add origin https://github.com/angelrps/master\_kschool\_tst.git**

le dice a git añademe este repositorio local a este repositorio remote en githiub. Hay que hacerlo solo una vez al empezar a trabajar con el repositorio. Si no se hace no se puede hacer push.

**git add “nombre de archivo”** : añade un archivo al repositorio

**git commit –m** “Texto explicativo sobre el commit” : git hace una foto del contenido del repositorio y añade un comentario explicativo de qué ha cambiado en ese commit. Pero ojo, esto pasa en el servidor local.

**Git push origin master** : para enviar el trabajo hecho a la rama master (que siempre existe por defecto). El push es el que manda el trabajo local al remoto. “origin” es la conexión en este caso github. Master es la rama. Se debe haber configurado el remote origin previamente.

Esto de abajo hay que hacer una vez para que git sepa quien eres:

Git config –global user.name “Angel Ruiz-Peinado”

Git config –global user.email [angelruizpeinado@gmai.com](mailto:angelruizpeinado@gmai.com)

**rm –rf directory** : eliminar directorio y todo lo que hay dentro

**git remote show origin** : muestra el repositorio remote al que estás haciendo push y pull.

**git ls-files** : show all files being tracked

**git checkout** <rama o master> : establece esa rama o master como actual

**rm –rf ./.git** : I don´t what this to be a local git repository

**2019.11.16 LÍNEA DE COMANDOS +**

**Mirar si instalar Shell zsh en lugar de bash**

Data Science challenge: prueba que hace Amadeus para entrar. Igor nos avisará con 10 días de antelación por si lo queremos hacer.

Cuándo se usa la línea de comandos:

Raw Data: conectarse a datos de origen confidenciales como las reservas de una aerolínea y sacar parte de ellos para analizar.

Structure Date:

Cuando hablamos de Linux hablamos de UNIX:

Command line tools

Terminal: pantalla negra donde se escribe

Shell: lo que interpreta el comando

Operating System: se llama kernel

Qué es GNU vs Linux:

Linux es el kernel

GNU es todo: command line + terminal + Shell + kernel (que cogieron el de linux)

GIT:

Antes de empezar a trabajr hacer primero un pull para tener la última versión actualizada.

Basic commands:

* Clone

LINEA DE COMANDOS

Primero se escribe comando y luego sus parámetros.

echo: devuelve lo que viene después

$: significa que después viene una variable

/ : significa que vamos a dar un path absoluto

Echo $SHELL: devuelve la Shell en la que estoy. Lo haces cuando te conectas a una máquina desconocidad para ver en qué Shell estás trabajando

Whoami : este comando te devuelve tu nombre de usuario

Cat : devuelve el contenido entero de un fichero

Cat /etc/os-release : este fichero está en todos los shells y nos da información relevante

Cat /proc/cpuinfo : nos da información del cpu

¿cómocrear una variable?

MASTER = Kschool

Echo $MASTER : me muestra “kschool”

Comillas dobles agrupan varios parámetros en uno e interpretan los parámetros contenidos

Comillas simples **no** interpretan el valor de los parámetros contenidos, interpretan lo de dentro como texto.

Acento inverso ´$MASTER´: ejecutan el comando que está debajo

Para ejecutar varios comandos en la misma línea usamos punto y coma ;

Echo; date : los dos son comandos

$ + () = ejecutar la variable que hay dentro del paréntesis

Echo “$(date)” : ejecuta el comando date. Ojo una variable (llamada por $) puede tener un comando dentro o no

Osea que hay varias maneras de ejecutar una variable. Intentar resumirlas juntas.

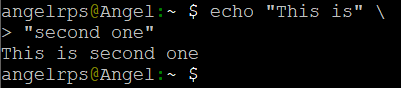
Como introducir una nueva línea: (varias formas)

Echo “lkshdjflkkhsd <enter>

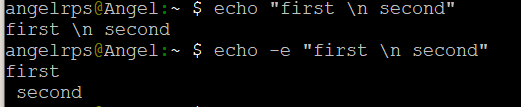
Continuoescribiendo” (cierro comillas y enter)

Echo “This is” \ <barra inversa y enter. La barra inversa no manda al Shell la info después de enter>

\n es nueva linea



Si no funciona poner -e



\t es tabulador

Para ignora una barra inversa hay que poner otra barra delante



Cd - : te lleva al directorio anterior en el que has estado

Cd ./ : mirar desde donde estamos. Para asegurarte que seleccionas el folder dentro deñl folder en que estás.

Cd . : muestra los directorios ocultos del folder donde estás

Osea para moverme a un directorio oculto pongo: cd .config <siendo config oculto>

Hay diferentes comandos con ls. Buscar en el pdf y hacer lista

-1, -s, -S, -r, -l, -lt, -a …..un montón

Cat : muestra contenido del archive

Cat –n : lo muestra enumerando las líneas. Si pongo varios ficheros unos detrás de otro concatena su contenido.

Cat \* + tab muestra contenido de todos los ficheros

Cat \*.txt : muestra todos los ficheros .txt

Cat \*.txt\* : que incluya txt

Cat \*dk\* : todos los ficheros cuyo nombre incluye dk

Chmod : comando para cambiar permisos (user, group, others)

Chmod u-w README.md : quito el permiso de write a user del archivo README

Chmod u+w,o-x README.md : doy permiso write a users y quitar execute a others.

Chmod 777 : dar permisos a todos en binario

Crear directorio + subdirectorio a la vez. –p es la clave

Mkdir –p two/three/four

Para crear un fichero es touch

Para copiar es cp y ponemos origne y destino

Cp file file.copy

Copiar archivo a otro directorio

Cp + “fichero” + directorio destino

Copiar varios ficheros a un directorio. El ultimo parámetro será el directorio, los del medio los ficheros

Con –r copiadmos todos lo de dentro de un directorio a otro

Duplicar un directorio con otro nombre

Cp –r two two\_copy

-p (preserve) para preservar todos los permisos de un archivo cuando lo copiamos

Mover un directorio un nivel por envima

Mv four ../

Mover un directorio y cambiarle el nombre. En realidad no lo hemos movido, es como cambiar el nomnbre

Mv four four\_new

Ahora sí mover y cambiar el nombre

Mv four and/new\_file (add/ sería el destino)

Para borrar rm + las ruta del fichero

Rm –r (borrar todo lo de dentro)

Rm –rf (f es forzar, que no me pregunte si quiero elminar ciertos ficheros)

Alias son como shorcuts. Cómo crear un alias

Alias cp=’cp –p’

Como añadir mis alias de un Shell a otro? No lo he pillado

Para meterte en un fichero y moverte dentro de el y leerlo el comando es less

Less permite hacer búsqueda dentro del texto con barra / más el textoa buscar

Poniendo –N te pone el número de línea

Q es para salir del less

Cuando se hace una búsqueda en less siempre se hace desde la línea primera de la pantalla no desde el principio del ficho

g vas al princpio del fichero y G vas al final

encontrar el manual de un comando: man + comando(ej: man ls)

ctrl + d salgo del Terminal pero la línea actual debe estar vacía

crtl + u : cortar línea

crtl + y : pegar

crtl + k : cortar desde cursor hasta el final

comando history muestra todos los comandos ejecutados en la sesión

crtl + r : para buscar un comando ejecutado en el historial. Sigues pulsando crtl +r va pasando a través de todos los comandos que contienen tu búsqueda

cómo ver principio, final y contar palabras de un fichero

head + normbre fichero : por defecto te saca primeras 10 lineas

head -5 nombrearchivo : te saca 5 lineas

cuando después de head se pone –n el programa sabe que después va un numero

head –n +5 muestras también las primeras 5 lineas

head –n -5 muestra TODO excepto las últimas 5 líneas

OJO la diferencia entre head -5 y head –n -5!!!!

Tail muestra el final de un documento. Y lo explicado antes sobre –n +-5 funciona al revés

Wc (wordcamp) te da el número total de líneas, palabras y bytes



Wc –l : lineas

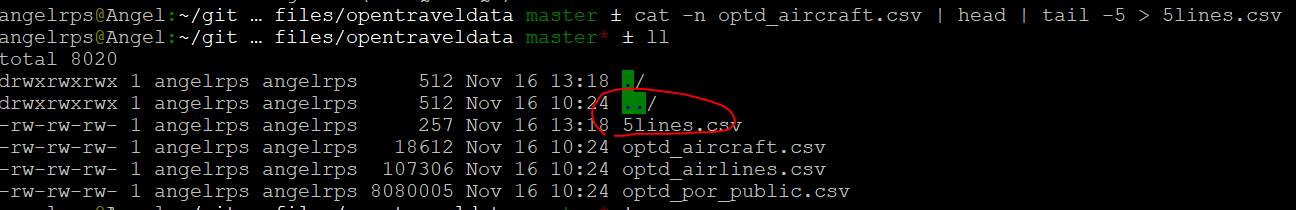
Wc –w : palabras

Wc –c : bytes

Encadenar comandos con Pipe |

>este símbolo graba texto en un fichero

>>doble símbolo lo añade al texto existente



Comando find para buscar (con filtros)

Te devuelve todos los archivos por debajo del directorio donde estás

Find | wc : número de líneas de la búsqueda para ver si es muy grande y se va a tirar dos años mostrando líneas

Primer parámetro siempre es el directorio donde comenzar la búsqueda

Find –name \*text\_file\*

iname : ignore case

-type d coje solo directorio

-type f coje solo ficheros

-maxdepth 4 maximo nivel de subfolders para buscar

-perm 777 mirar ficheros con todos los permisos

¡-perm

-empty tamaño 0

-size -10M menor de 10M

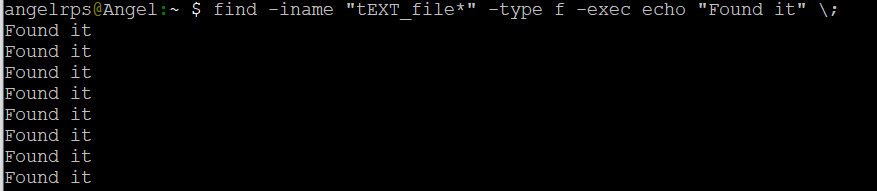
-size +10M mayor de 10M

Siempre poner – o + porque si no estoy buscando el tamaño EXACTO de 10M

-mmin -60 ficheros tocados en los últimos 60 minutos o menos

-mmin +60 ficheros tocados hace 60 minutos o más

-exec para ejecutar el input que tenga detrás



Ha executado la escritura de Found it tantas veces con archivos aha encontrado

{} para que ejecute el comando sobre cada fichero encontrado

Lo último de la clase no lo he pillado, coger la info y después de –exec crear un Shell

Qué cojones es un shell