Linux and more. Data processing

Artem Trunov for Ozon Masters

Bash vs. python

- Jupyter - это очень просто и быстро.
- Не будем пытаться его заменить во всем.
- Только кое в чем :)
- А так же посмотрим, как работать с ноутбуками из командной строки.

Что уже делали

- Grep
- Pipes, redirection
- sorting

grep

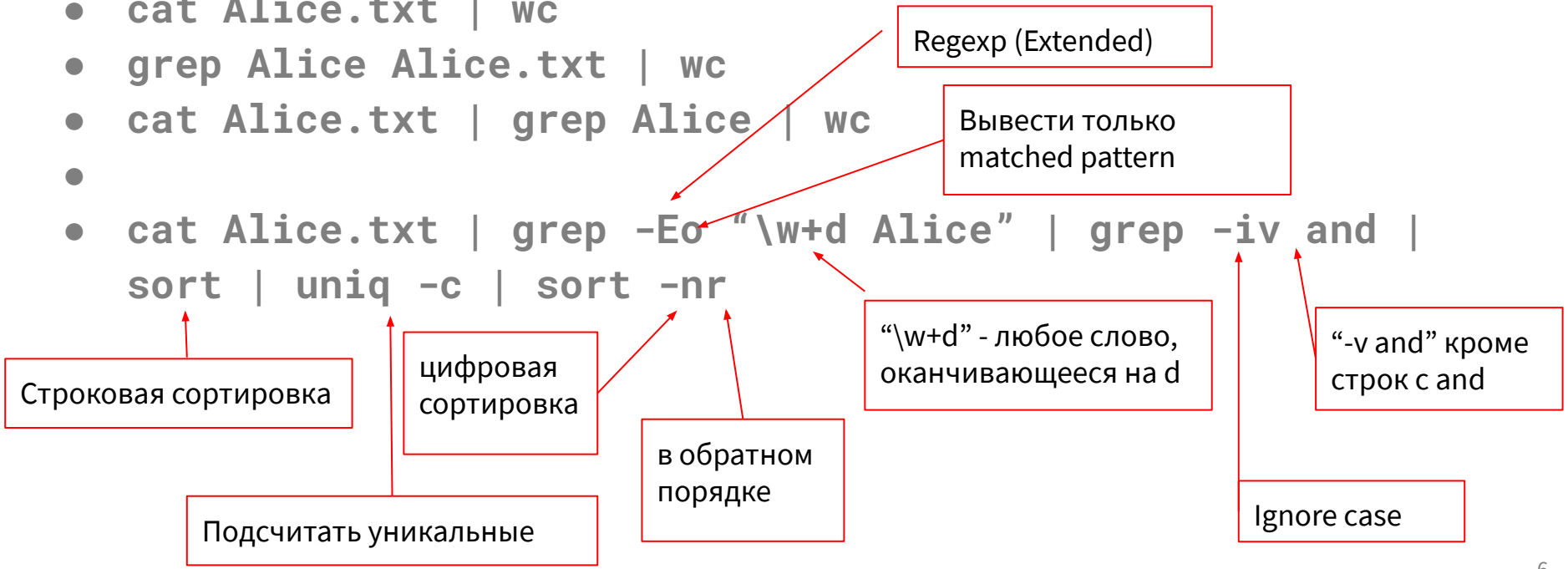
- wget -O Alice.txt http://www.gutenberg.org/files/11/11-0.txt
- grep Alice Alice.txt
- grep -c Alice Alice.txt
- grep -o Alice Alice.txt
- grep -Eo “\w+d Alice” Alice.txt
  - -E - extended regexp, -F, -P
- grep -Eo -A 1 “\w+d Alice” Alice.txt
  - -B, -C
- grep -i Alice Alice.txt
- grep -v Alice Alice.txt
- grep Alice *
- grep -r Alice *

Как программы работают вместе? Pipes

- Выход одной программы может подаваться на вход другой программы
- `cat Alice.txt | wc`
- `grep Alice Alice.txt | wc`
- `cat Alice.txt | grep Alice | wc`
- `cat Alice.txt | grep -Eo “\w+d Alice” | grep -iv and | sort | uniq -c | sort -nr`

Как программы работают вместе? Pipes

- Выход одной программы может подаваться на вход другой программы
- `cat Alice.txt | wc`
- `grep Alice Alice.txt | wc`
- `cat Alice.txt | grep Alice | wc`
-
- `cat Alice.txt | grep -Eo “\w+d Alice” | grep -iv and | sort | uniq -c | sort -nr`



sort | uniq -c - что это?

sort | uniq -c - что это?

Group by + count

Другие команды-фильтры

- awk, sed - строчные редакторы
- tr - замена и удаление символов
- head, tail, split, csplit - файл по частям
- sort - сортировка
- uniq - фильтр уникальных слов/строк
- shuf - перемешивание строк
- comm, cmp, diff - сравнение двух файлов
- join - слияние двух файлов по общему полю
- cut - вырезание колонок из файла
- paste - собрать файл с колонками из других файлов
- nl - пронумеровать строки

awk, sed

- Потоковые (построчные процессоры) - для обработки строковых и числовых данных.
  - Подразумевается цикл по строкам взодного файла
    - На питоне так нельзя (а на perl можно, кстати)
- Разработаны в Bell Labs
- По сути - языки программирования
- Очень хороши (и главным образом используются) для one-liners - набора команд в “трубопроводе” (pipeline)
- Sed - для замены строк
- Akw - операции над полями (колонками)

sed

```
$ less Alice.txt | sed 's/said/exclaimed/g' | grep said
$ less Alice.txt | sed 's/said/exclaimed/g' | grep exclaimed | head -2
```
many miles I’ve fallen by this time?” she exclaimed aloud. “I must be
before,” exclaimed Alice,) and round the neck of the bottle was a paper

```
$ head -3 Alice.txt
```
The Project Gutenberg EBook of Alice’s Adventures in Wonderland, by Lewis Carroll

This eBook is for the use of anyone anywhere in the United States and most

```
$ head -3 Alice.txt | sed '1d'
```

This eBook is for the use of anyone anywhere in the United States and most

Sed - шаблоны

- Шаблон-действие (адрес-команда)
- Шаблоны (регвыры)
  - / - просто разделитель шаблона, но можно использовать и другой с командой s
  - . - любой символ
  - * - несколько предшествующих символов (a* несколько а подрядб ю* несколько любых подряд)
  - ? - один или ноль предшествующих символов
  - + - один или несколько предшествующих
  - ^, $ - матчинг в начале или конце строки (/^Alice/ - матчить Alice в начале строки)

```
$ cat Alice.txt | sed -n '/^Alice/p' | head -2
```

Alice’s Adventures in Wonderland
Alice was beginning to get very tired of sitting by her sister on the

Sed address

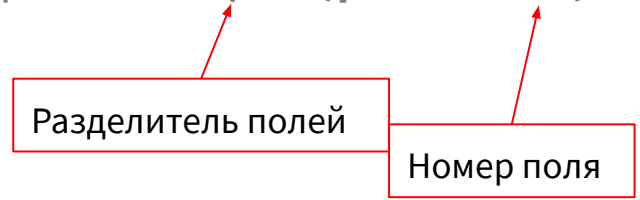
```
$ cat Alice.txt | sed -n '3p'
```
This eBook is for the use of anyone anywhere in the United States and most
```
$ cat Alice.txt | sed -n '3,5p'
```
This eBook is for the use of anyone anywhere in the United States and most
other parts of the world at no cost and with almost no restrictions
whatsoever. You may copy it, give it away or re-use it under the terms of

```
$ cat Alice.txt | sed -n '/CHAPTER I.\s*$/,$p' | head -5
```
CHAPTER I.
Down the Rabbit-Hole

Alice was beginning to get very tired of sitting by her sister on the
```
$ cat Alice.txt | sed -rn '/CHAPTER [IVX]+\.\s*$/,+4p'
```
-печатает по четыре строчки с начала каждого раздела (обратите внимание -r)

awk

```
$ head -3 /home/datamove/Bodyfat.csv | awk -F, '{print $5}'
"Height"
67.75
72.25
```



Попробуем перевести дюймы в сантиметры

```
$ head -3 Bodyfat.csv | awk -F, '{print $5*2.54}'
0
172.085
183.515
```

Оставляем заголовок

```
$ head -3 Bodyfat.csv | awk -F, 'NR==1{print $5} NR>1{print $5*2.54}'
"Height"
172.085
183.515
```

awk - код

- Условие{действие}
- Условие
  - Сравнение, шаблон регвыра, BEGIN, END

$ head -3 Bodyfat.csv | awk -F, 'NR==1{print $5} NR>1 {print $5*2.54} END{print “Записей:” NR}'
"Height"
172.085
183.515
записей:3
$ head -3 Bodyfat.csv | awk -F, 'BEGIN{x=0} {print $5; x+=$5} END{print “сумма”, x}'
"Height"
67.75
72.25
сумма 140
(на самом деле переменные можно не инициализировать, т.е. секцию END тут можно опустить)

Awk - переменные

- NR - номер текущей записи (строки - по умолчанию)
- NF - число полей в текущей записи
- FS,OFS - входной, выходной разделитель полей (по умолчанию - пробел)
- RS, ORS - входной, выходной разделитель записей

$n - значение в n-м поле

$(NF-2) - значение второго с конца поля

awk - запуск

```
#!/usr/bin/awk

BEGIN{
   FS=’,’
}

{
   print $5
   x+=$5
}

END{
  print “сумма”, x
}
```

```
$ chmod +x summa.awk; ./summa.awk Bodyfat.csv
```

awk - еще много всего

Функции, опции, перенаправление потока в действиях…

https://en.wikipedia.org/wiki/AWK

man awk

Пакеты для конкретных задач

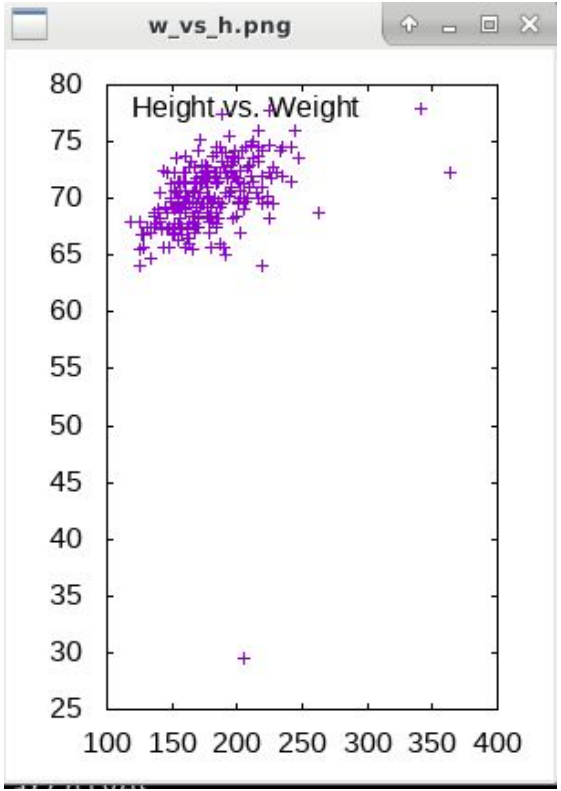
- Csvkit
  - pip3 install csvkit

```
$ in2csv Bodyfat.csv | csvsql --query "select Height,Weight from stdin where Height>70" | csvsort -r -c Weight | head -5 | csvlook
```

Height	Weight
72.25	363.15
73.50	247.25
76.00	244.25
74.50	241.75

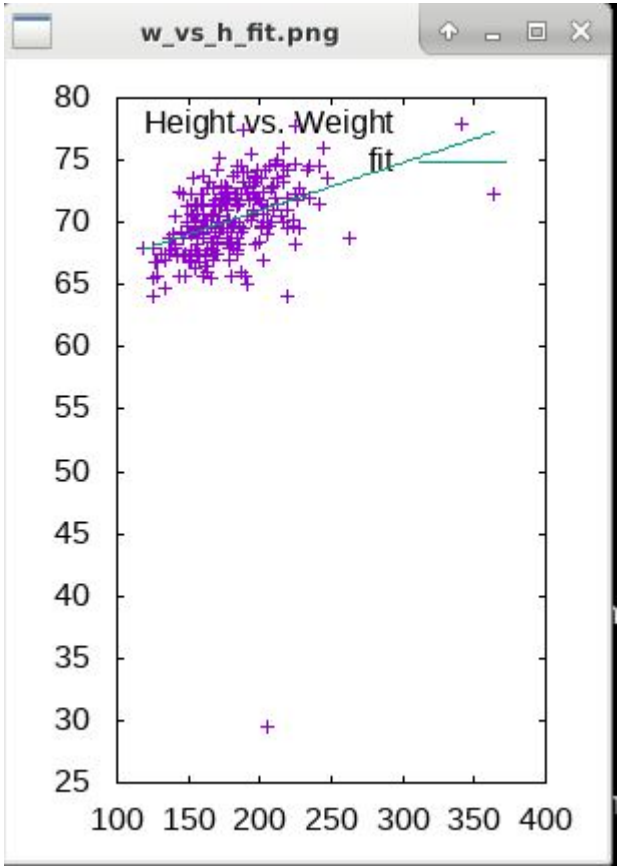
Gnuplot - графики из командной строки

```
$ gnuplot
gnuplot> set terminal png size 300,400
gnuplot> set output 'w_vs_h.png'
gnuplot> set datafile separator comma
gnuplot> plot '/home/datamove/Bodyfat.csv' using 4:5
title 'Height vs. Weight' with points
```



Gnuplot - regression

```
gnuplot> f(x) = m*x +b
gnuplot> fit f(x) '/home/datamove/Bodyfat.csv'
using 4:5 via m,b
gnuplot> set output 'w_vs_h_fit.png'
gnuplot> plot '/home/datamove/Bodyfat.csv'
using 4:5 title 'Height vs. Weight' with
points, f(x) title 'fit'
```



Запуск юпитера из командной строки

(pip3 install nbconvert)

nbconvert --to notebook --ExecutePreprocessor.timeout=-1 --execute my.ipynb

Вычисления

- В awk
  - ‘BEGIN{A=0} {A=A+$1} END{print A}’
- Используя синтаксис bash
  - $((2 +2))
    - Но только с целыми числами
- Используя bc
  - echo “3*3” | bc

Полезные утилиты

- convert - преобразование картинок
- ffmpeg - работа с файлами видео
- pdftk - работа с пдф
  - кстати , lifehack - можно смотреть содержимое пдф командой less, которая и в пакетном режиме работает (без интерактива).
- enca, iconv - перекодировщики текста
-

Проект 1

- Выбирайте
  - Свои данные, своя обработка - 15 баллов
  - Общий проект (я даю датасет и задание) - 10 баллов
- В проекте должны использоваться
  - Датасет (<20 МБ)
  - Вычисления 3-х каких-либо характеристик (см. Задание к общему проекту) и вывод в терминал
  - Отрисовка графика либо запуск тетрадки юпитера либо запуск программы на питоне
    - Я не буду запускать process.sh для вашего проекта, только для общего. Так что о пакетах для питона, пути запуска и т д не надо беспокоиться.