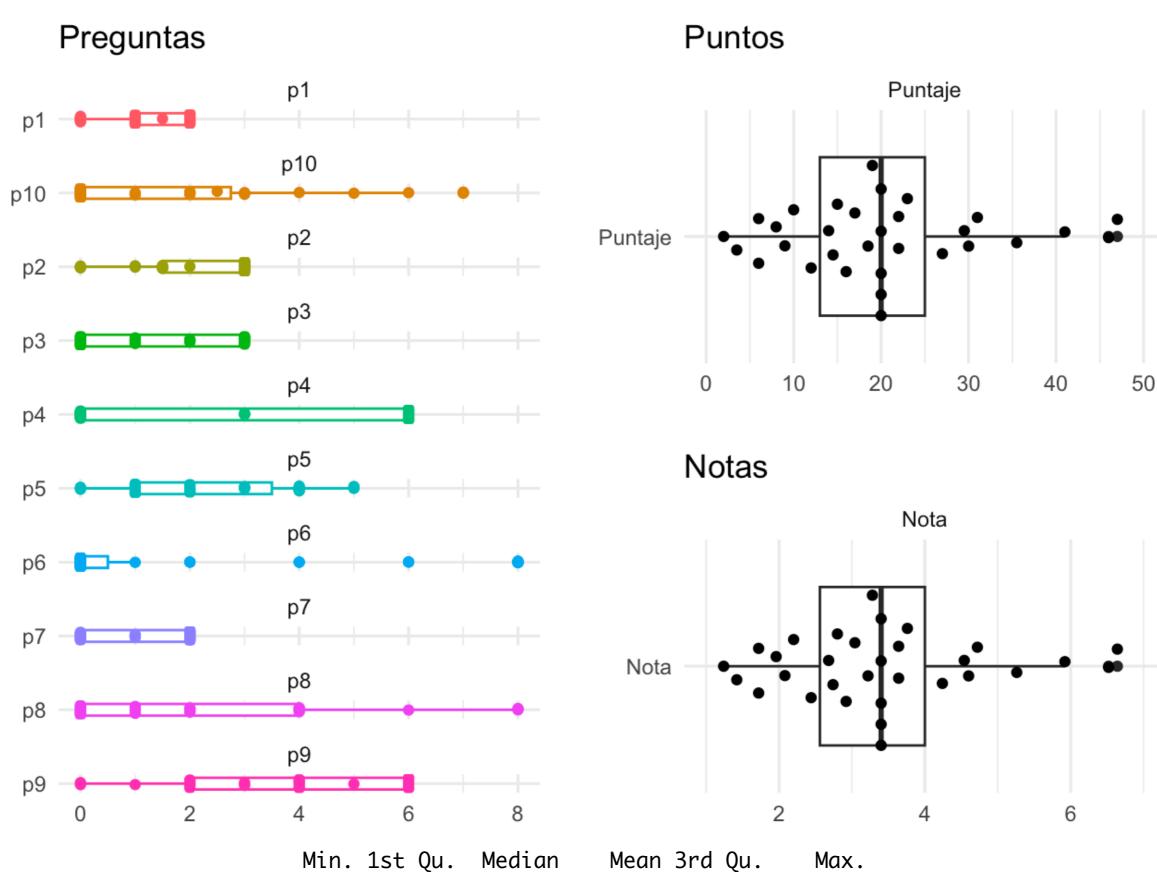
Ejemplos Hadoop y Nextflow

Alex Di Genova

Notas Control 1



2.560

1.240

3.400

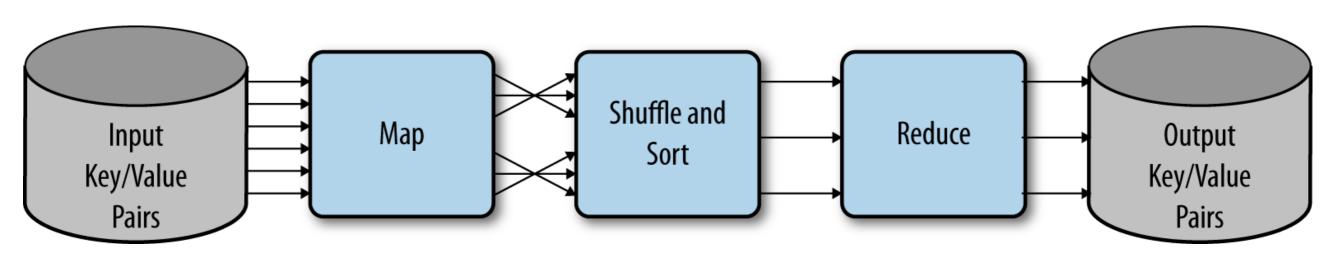
3.417

4.000

6.640

Hadoop

MapReduce: Un modelo de programación funcional



- 1.Los datos locales se cargan en un proceso de mapeo como pares clave/valor de HDFS.
- 2.Los mapeadores generan cero o más pares clave/valor, asignando valores calculados a una clave en particular.
- 3.Luego, estos pares se ordenan/barajan en función de la clave y luego se pasan a un reductor de modo que todos los valores de una clave estén disponibles.
- 4.Los reductores deben generar cero o más pares clave/valor finales, que son la salida.

Hadoop Map/Reduce

```
def map(dockey, line):
   for word in value.split():
      emit(word, 1)

def reduce(word, values):
      count = sum(value for value in values)
      emit(word, count)
```

```
# Input to WordCount mappers
(27183, "The fast cat wears no hat.")
(31416, "The cat in the hat ran fast.")
# Mapper 1 output
("The", 1), ("fast", 1), ("cat", 1), ("wears", 1),
("no", 1), ("hat", 1),(".", 1)
# Mapper 2 output
("The", 1), ("cat", 1), ("in", 1), ("the", 1),
("hat", 1), ("ran", 1),("fast", 1),(".", 1)
# Input to WordCount reducers
# This data was computed by shuffle and sort
(.", [1, 1])
("cat", [1, 1])
("fast", [1, 1])
("hat", [1, 1])
("in", [1])
("no", [1])
("ran", [1])
("the", [1])
("wears", [1])
("The", [1, 1])
# Output by all WordCount reducers
(.", 2)
("cat", 2)
("fast", 2)
("hat", 2)
("in", 1)
```

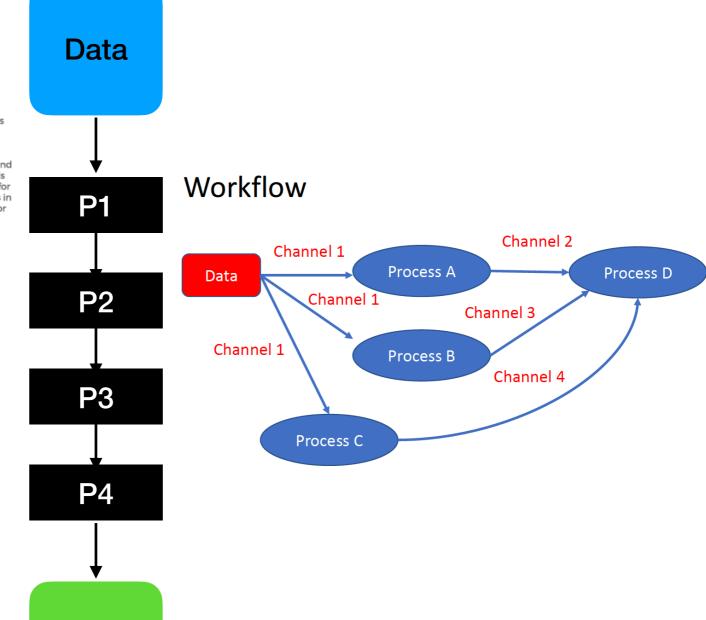
("no", 1)

https://github.com/adigenova/ uohpmd/blob/main/code/ Haddop_nextflow.ipynb

NATIONAL CANCER INSTITUTE Genome Characterization Pipeline Tissue samples are processed Tissue Source Sites into molecular analytes. Clinical data is harmonized. samples & clinical data Clinical Trials and oncolgy groups identify consenting patients and collect Biospecimen Core Resource samples and clinical data. Clinical, biospecimen, & Molecular analytes are characterized pathology data via sequencing and other high-throughput technologies. Molecular data is Molecular harmonized via analytes best practice bioinformatic **Genome Characterization Centers** pipelines. Raw and processed data is made available for analysis by users in the data portal or by download. Raw sequencing & other characterization Raw, harmonized & derived data **Genomic Data Commons** Network researchers analyze the data and publish results. **Analyses** & results 11 harmonized & derived data **Genome Data Analysis Network** Researchers around the world access de-identified clinical and harmonized genomic data. **Research Community** cancer.gov/ccg

NextflowGenómica

Result



NATIONAL CANCER INSTITUTE **Genome Characterization Pipeline** Tissue samples are processed into molecular analytes. Clinical data is harmonized samples & clinical data Clinical Trials and oncolgy groups identify consenting patients and collect Biospecimen Core Resource samples and clinical data Clinical, biospecimen, & Molecular analytes are characterized pathology data via sequencing and high-throughput technologies Molecular data is harmonized via analytes best practice bioinformatic Genome Characterization Centers pipelines. Raw and processed data is made available for analysis by users in the data portal or by download. Raw sequencing & other characterization Raw, harmonized & derived data Genomic Data Commons Network researchers analyze the data and publish results. Analyses harmonized & derived data Genome Data Analysis Network Researchers around the world access de-identified clinical and harmonized genomic data **Research Community** cancer.gov/ccg

NextflowGenómica

Requisitos de pipelines para proyectos genómicos a gran escala

- Reproducible: los resultados genómicos deben ser totalmente reproducibles
- **Escalable**: Facil ejecucion en cluster HPC o cloud.
- Portátil: puede ejecutarse en varias infraestructuras (diferente sistema operativo, cloud)
- Manejar la heterogeneidad: funciona con dependencias de software en conflicto y varios requisitos de recursos.

Nextflow

Primer pipeline

```
//pipeline 01.nf
                     nextflow_enable_dsl=2
Procesos
                    process INDEX {
                           path transcriptome
                         output:
                           path 'index'
                         script:
                           salmon index -t $transcriptome -i index
                      process QUANT {
                           each path(index)
                          tuple(val(pair_id), path(reads))
                         output:
                           path pair_id
                         script:
                           salmon quant --threads $task.cpus --libType=U \
                              −I $index \
                              -1 ${reads[0]} -2 ${reads[1]} -o $pair_id
                    //Definición del pipeline
                     workflow {
                         //canales de entrada
                       transcriptome_ch =
                         channel.fromPath('transcriptome/*.fa.gz',checkIfExists: true)
                     read_pairs_ch =
                         channel.fromFilePairs('reads/*_{1,2}.fq.gz',checkIfExists: true)
                        //el proceso INDEX toma el calar transcriptome_ch
                        index ch = INDEX(transcriptome ch)
                        //el proceso QUANT toma dos canales como argumentos: Indice y las lecturas
                        QUANT( index_ch, read_pairs_ch ).view()
```

https://github.com/ adigenova/uohpmd/blob/ main/code/Nextflowl.ipynb

Preguntas?