

USING HADOOP TO IDENTIFY FALSE POSITIVES IN PYROPRINTING

Colin Adams & Skyler Gordon

IAB Presentation

Spring 2016

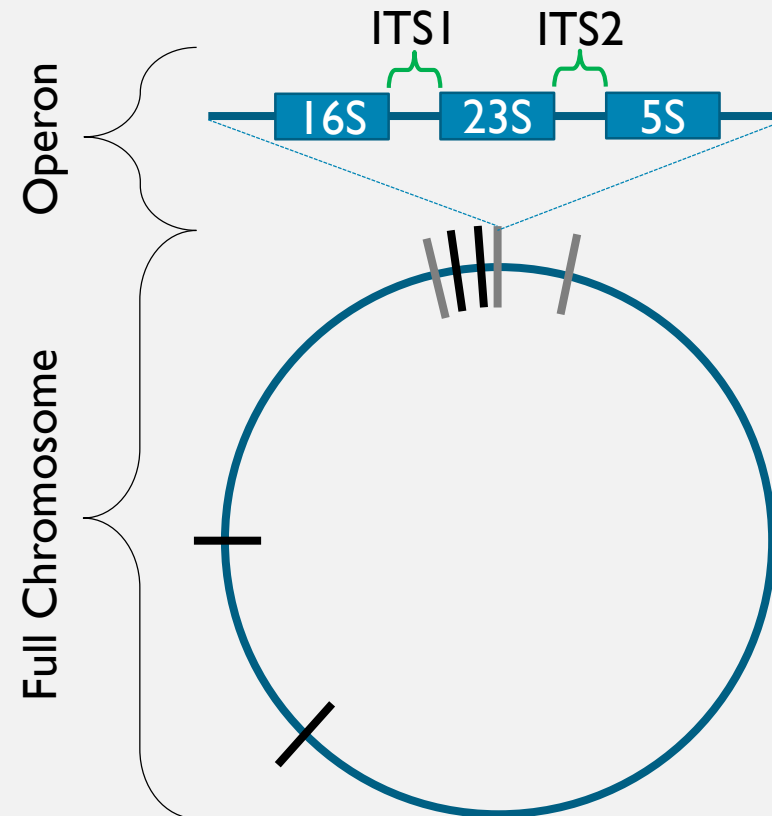
AGENDA

- Background
 - *E. coli* and Pyroprinting
- Implementation
 - Hadoop
- Results

BACKGROUND

E. COLI DNA

- Structure
 - Circular chromosome
 - Seven copies of rRNA operons
 - Operon structure
- **Interested in copies of ITS1 and ITS2 alleles**
- Allele Ratio
 - 7 - all seven alleles are same sequence
 - 6:1 – six of one allele, one different
 - 5:2 – five of one allele, two of another
 - ...
 - 1:1:1:1:1:1:1 – every allele is different



PYROPRINTING

- Rapid, inexpensive strain typing method
- Creates fingerprint from ITS1 or ITS2 alleles
- Two *E. coli* with matching pyroprints considered same strain

Question: How often do the pyroprints between *E. coli* match but the allele ratio or sequences differ? In other words, how frequent is a false positive?

Seven copies of ITS2 (4:3 ratio)

```
ITS2-1 GCCGAAGATGTTTT...
ITS2-2 GCCGAAGATGTTTT...
ITS2-3 GCCGAAGATGTTTT...
ITS2-4 GCCGAAGATGTTTT...
ITS2-5 GCCGAAGGTGTTTT...
ITS2-6 GCCGAAGGTGTTTT...
ITS2-7 GCCGAAGGTGTTTT...
```

ITS2 Pyroprint

7, 14, 7, 14, 10, 4, 7, 7, 28, ...

SOLUTION STRUCTURE

- Determine alleles and allele ratio for each *E. coli* isolate
 - If two isolates have matching pyroprints but different alleles/ratio → false positive

- Process

- Choose 100 *E. coli* isolates
- Amplify & Modify ITS Regions (PCR)
- Have sequenced (Illumina/Mr. DNA)
 - 5-20 million sequences
- Analyze sequences (Hadoop)
 - Count alleles for each isolate and region
 - Determine allele ratio

TAT	<u>ATCG</u>	T	<u>GGA...C</u>	<u>ACGTACGT...</u>
	barcode		its 1/2 primer	allele

IMPLEMENTATION

HADOOP BACKGROUND

- Distributed storage and processing framework
- Simple data processing pipeline
 - Map(obj) -> <key, value>...
 - Transforms each input object into any number of <key, value> pairs
 - Shuffle
 - Map() output is shuffled so all objects with same key are grouped for Reduce()
 - Reduce(key, values...) -> <key, value>...
 - Performs reduction on values with matching key and outputs any number of results

FIRST HADOOP JOB

Text File (10m)

TATATCG...

Map

Attempts to parse into Sequence object

KEY:
validity:barcode:region:allele
VALUE:
sequence

Reduce

VALID
Sums occurrences
INVALID
Outputs individual

Valid

KEY:
barcode:region
VALUE:
allele:count

Invalid

```
{  
  "Validity":  
  "Barcode":  
  "Region":  
  "Allele":  
  "Original Sequence":  
}
```

Possible Validity Codes:

- **Valid**
- Invalid Barcode
- Invalid Region*
- Too Short

**Configurable Approx. Match*

TAT GTCA T GGAA...AC CTCC...GT

KEY:
Valid:GTCA:ITS1:CTCC...GT
VALUE:
TATGTCATGGAA...ACCTCC...GT

KEY:
GTCA:ITS1
VALUE:
CTCC...GT:1

SECOND HADOOP JOB

Text File (Prior)

```
KEY:  
  barcode:region  
VALUE:  
  allele:count  
...
```

Identity Map

Reduce

```
Mutated  
Allele count < 1% of Total  
Result  
Analyze ratio of other allele counts
```

Analysis:

- Look at allele ratios w/ same number
 - E.g. 2 alleles = 6:1, 5:2, 4:3
- Run Chi-Square w/ expected counts
- If there is only one that isn't rejected, it is valid

Mutated

```
"barcode":  
"region":  
"allele":  
"count":
```

Result

```
"barcode":  
"region":  
"statisticalValidity":  
"pValue":  
"ratios":  
"alleleCountRatios": []
```

```
GTCA:ITS1   CTCC...GT:40  
GTCA:ITS1   CTCC...AT:30
```

```
{  
  "barcode": GTCA,  
  "region": ITS1,  
  "statisticalValidity": true,  
  "pValue": 1.0,  
  "ratios": [4:3],
```

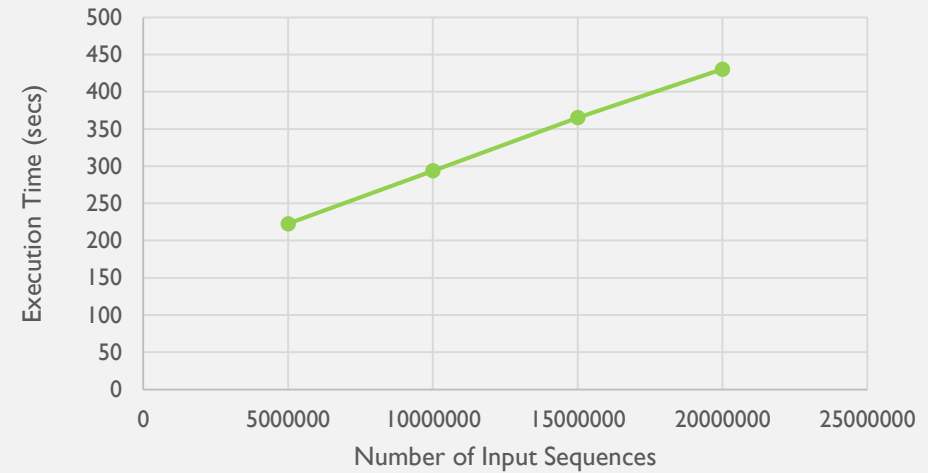
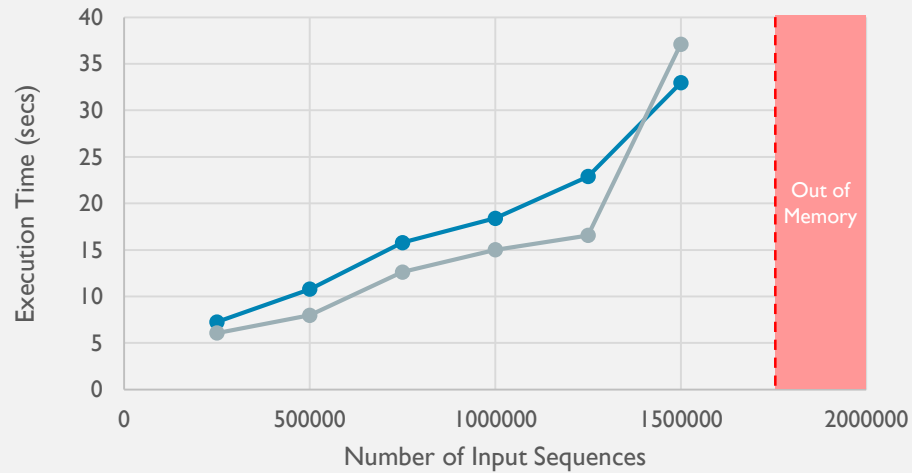
```
  "alleleCountRatios": [  
    {"allele": "CTCC...GT", "count": 40, "ratio": 4},  
    {"allele": "CTCC...AT", "count": 30, "ratio": 3}  
  ]  
}
```

RESULTS

RAW RESULTS

```
{ "catalogNumber": 9, "region": "ITS1", "strainName": "Av-009", "statisticallyValid": false,
  "ratios": ["THREE_ONE_ONE_ONE_ONE"], "pValue": 0.0, "barcode": "GTCA",
  "alleleCountRatios": [
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGAAGTGCTCACACAGATTGTCTGATAGAAAAGTGAAAAGCAAGGCCGT", "count": 6483, "ratios": [3] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGAAGTGCTCACACAGATTGTCTGATGAAAATGAGCAGTAAAACCTCT", "count": 1370, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTACTTTGCAGTGCTCACACAGATTGTCTGATGAAAATGAGCAGTAAAA", "count": 1072, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGCAGTGCTCACACAGATTGTCTGATGAAAAGTAAATAGCAAGG", "count": 227, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTACTTTGCAGTGCTCACACAGATTGTCTGATAGAAAAGTGAAAAGCAA", "count": 210, "ratios": [1] }
  ]
}
{ "catalogNumber": 9, "region": "ITS1", "strainName": "Cw-053", "statisticallyValid": false,
  "ratios": ["THREE_ONE_ONE_ONE_ONE"], "pValue": 0.0, "barcode": "CGAG",
  "alleleCountRatios": [
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGAAGTGCTCACACAGATTGTCTGATAGAAAAGTGAAAAGCAAGGCCGT", "count": 6497, "ratios": [3] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGAAGTGCTCACACAGATTGTCTGATGAAAATGAGCAGTAAAACCTCT", "count": 1413, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTACTTTGCAGTGCTCACACAGATTGTCTGATGAAAATGAGCAGTAAAA", "count": 1035, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTACTTTGCAGTGCTCACACAGATTGTCTGATAGAAAAGTGAAAAGCAA", "count": 201, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGCAGTGCTCACACAGATTGTCTGATGAAAAGTAAATAGCAAGG", "count": 171, "ratios": [1] }
  ]
}
{ "catalogNumber": 9, "region": "ITS1", "strainName": "Pg-085", "statisticallyValid": false,
  "ratios": ["THREE_ONE_ONE_ONE_ONE"], "pValue": 0.0, "barcode": "GCCA",
  "alleleCountRatios": [
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGAAGTGCTCACACAGATTGTCTGATAGAAAAGTGAAAAGCAAGGCCGT", "count": 6269, "ratios": [3] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGAAGTGCTCACACAGATTGTCTGATGAAAATGAGCAGTAAAACCTCT", "count": 1276, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTACTTTGCAGTGCTCACACAGATTGTCTGATGAAAATGAGCAGTAAAA", "count": 1030, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTACTTTGCAGTGCTCACACAGATTGTCTGATAGAAAAGTGAAAAGCAA", "count": 222, "ratios": [1] },
    { "allele": "CTCCTTACCTTAAAGAAGCGTTCTTTGCAGTGCTCACACAGATTGTCTGATGAAAAGTAAATAGCAAGG", "count": 154, "ratios": [1] }
  ]
}
```

PERFORMANCE

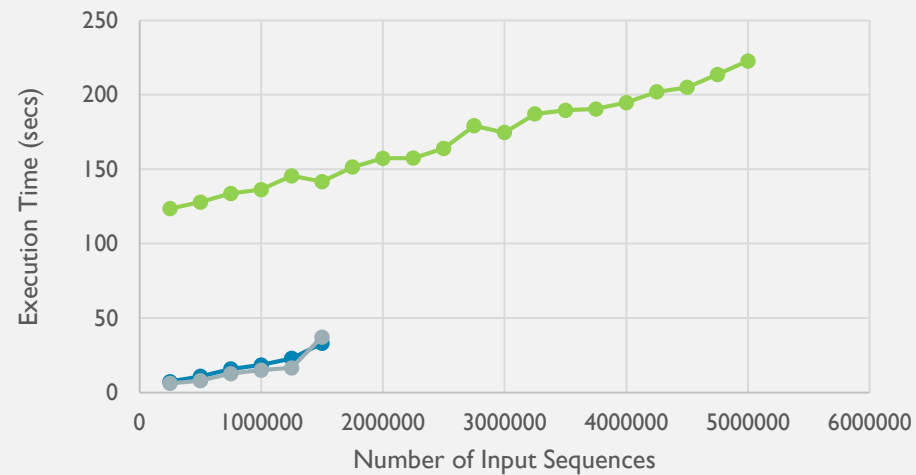


Sequential Parallel

Hadoop

Sequential & Parallel

- Single-Machine
 - Intel i7
 - 8 GB RAM
- Java 8 Streams API



Sequential Parallel Hadoop

Hadoop

- Google Cloud Compute
 - 1 Master
 - 3 Workers
 - 8 GB RAM each

LESSONS LEARNED

COLIN

- Technical
 - Hidden Complexity
 - Sequential → Hadoop
 - Cloud services
- Interdisciplinary
 - Be involved early and often
 - Many problems need CS
 - Efficiency

SKYLER

- Future Implications

THANK YOU