Getting Started with PySpark

Scaling data processing from a single machine to a distributed system

---

Hanna Torrence

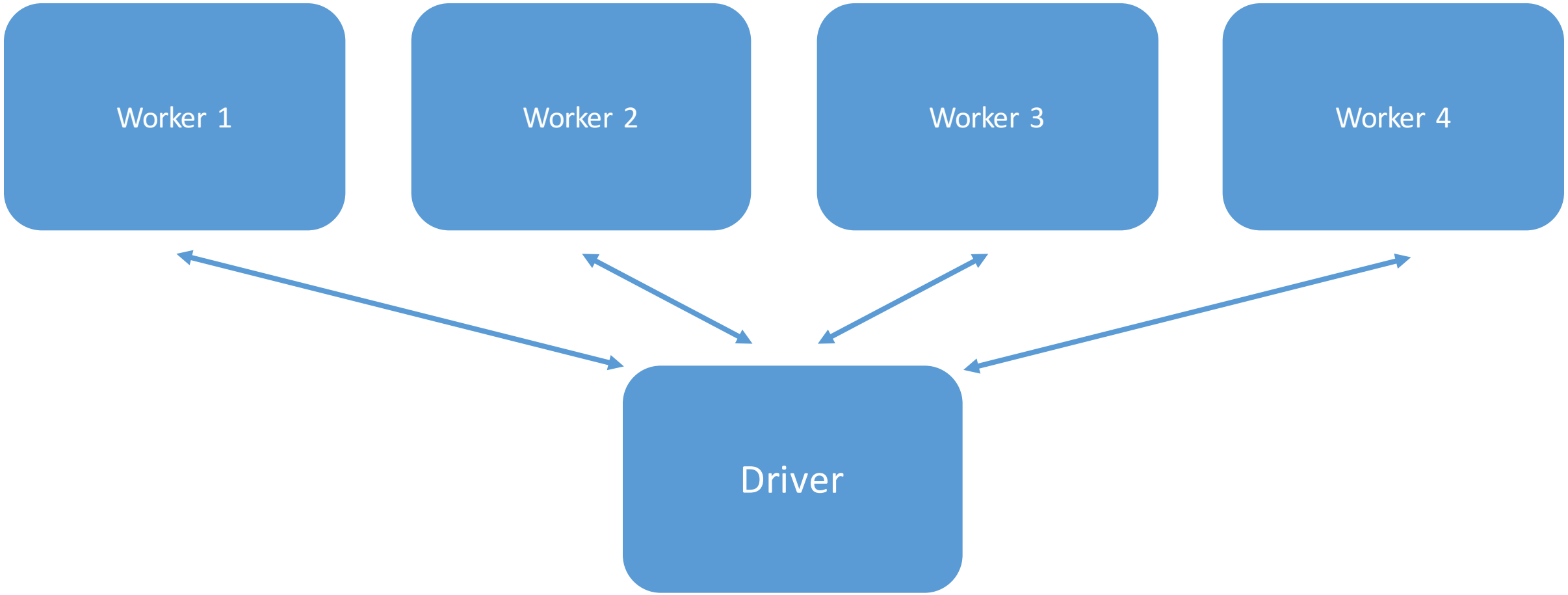
Data Scientist at ShopRunner

@HannaTorrence

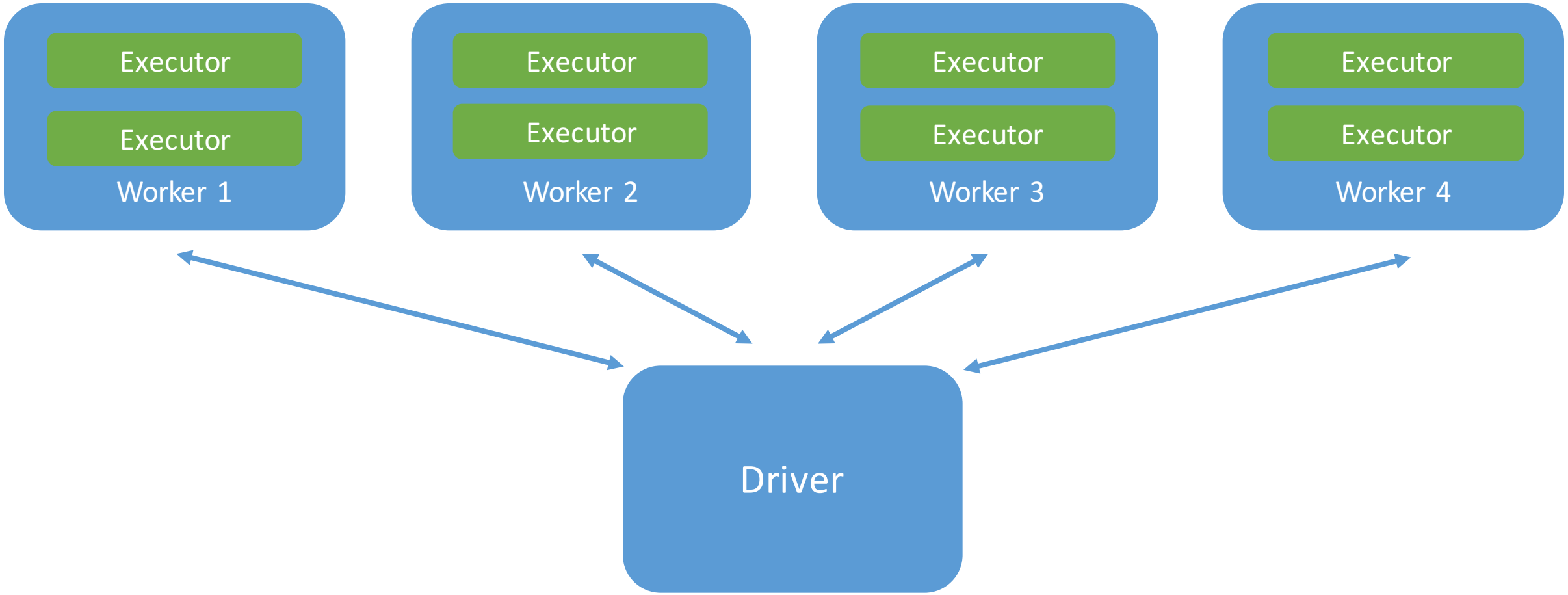
- ❑ **Intro to Spark**
- Data Exploration
- Data Processing
- Debugging Errors
- Building a Model

Why PySpark?

Distributed Computing



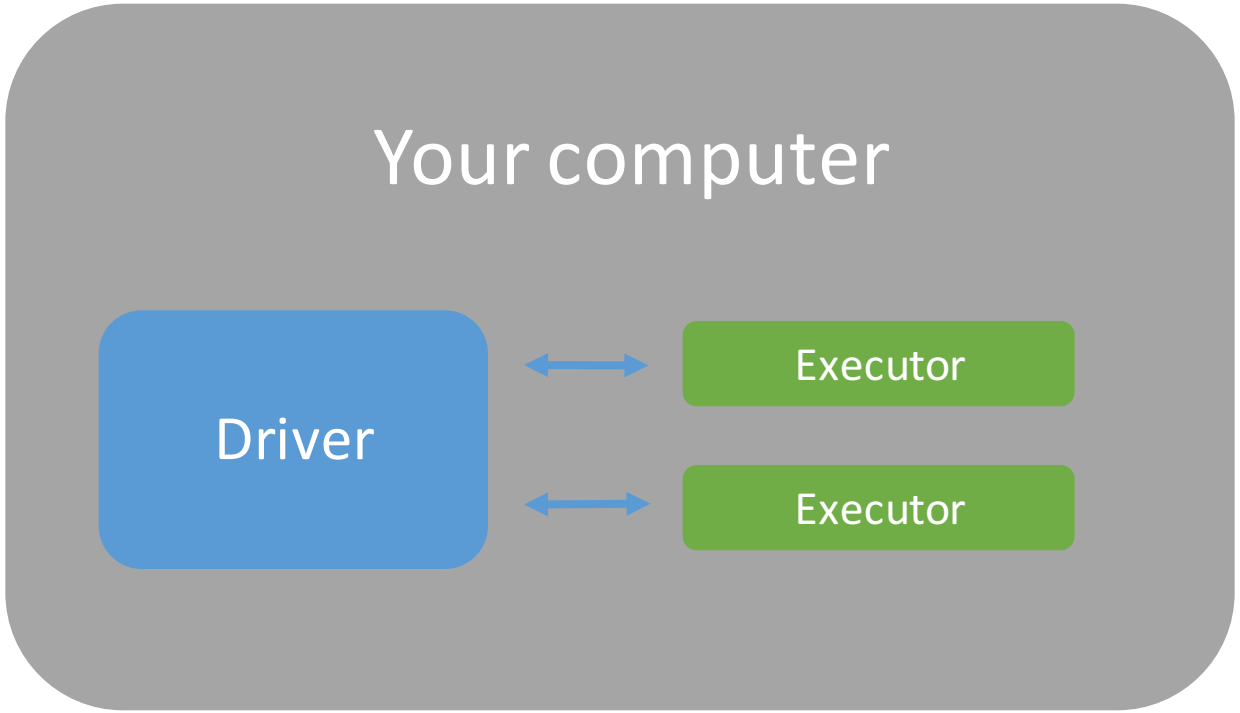
Distributed Computing



Spark Local Mode

```
In [2]: from pyspark.sql import SparkSession

spark = (
    SparkSession
    .builder
    .appName('intro')
    .master('local[2]')
    .getOrCreate()
)
```



Actions + Transformations

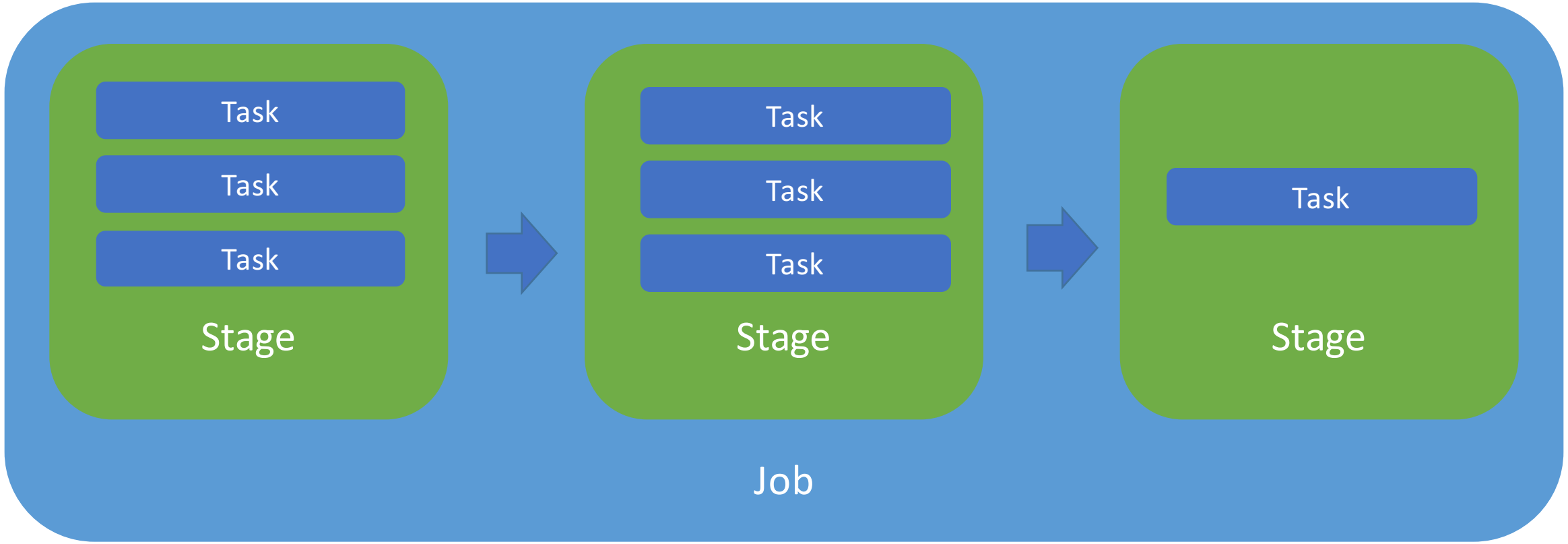
Transformation:

- Tell spark what data manipulations it should plan to do
- Will always return another dataframe
- Examples: select, groupBy, orderBy, join

Action:

- Tell spark to execute whatever it needs to to give you the result you asked for **this won't always be what you think!**
- Can return any type, or have side effects
- Examples: count, show, write

Spark Jobs



One **job** for each action called

One **stage** for each data shuffle

One **task** for each partition of data to be processed

Notebook

- Intro to Spark
- ❑ **Data Exploration**
- Data Processing
- Debugging Errors
- Building a Model

Spark SQL

```
# pyspark
df.groupBy('animal').count()
```

```
# pandas
df.groupby('animal').count()
```

```
# SQL
select animal, count(*)
from df
group by animal
```

Docs (also linked in notebooks):
https://spark.apache.org/docs/latest/api/python/pyspark.sql.html

Visualizations

… are not great in raw pyspark

Mostly sample or group data down to something that can be put in a pandas dataframe, and use your favorite python plotting functions from there.

HandySpark is a cool open source library to handle some of this, but comes at a performance cost.

Notebook

Break

- Intro to Spark
- Data Exploration
- ❑ **Data Processing**
- Debugging Errors
- Building a Model

Common Spark SQL Functions

DataFrame Methods

- select
- withColumn
- groupBy
- orderBy
- where / filter
- distinct

```
df.select(f.col('animal'))
```

Common Spark SQL Functions

DataFrame Methods

- select
- withColumn
- groupBy
- orderBy
- where / filter
- distinct

```
df.select(f.col('animal'))
```

Column Methods

- alias
- cast
- isNull
- isNotNull

```
df.select(f.col('animal').alias('type_of_pet'))
```

Common Spark SQL Functions

DataFrame Methods

- select
- withColumn
- groupBy
- orderBy
- where / filter
- distinct

Column Methods

- alias
- cast
- isNull
- isNotNull

Column Functions

- lit
- sum
- max
- min
- avg
- countDistinct

```
df.select(f.col('animal'))
```

```
df.select(f.min(f.col('age')))
```

```
df.select(f.col('animal').alias('type_of_pet'))
```

Notebook

Break

- Intro to Spark
- Data Exploration
- Data Processing
- ❑ **Debugging Errors**
- Building a Model

Common Categories of Errors

- Memory errors
  - You tried to collect too much data to the driver
  - Too much data ended up in a single partition

- Programming mistake
  - ... and now you need to navigate the stack trace to find the error

- This is taking longer than it should
  - Something is getting re-computed
  - Actions aren’t happening where you think they are

Stack Traces

```
Py4JJavaError: An error occurred while calling o451.save.
: org.apache.spark.SparkException: Job aborted.
```

```
Caused by: org.apache.spark.SparkException: Job aborted due to stage failure: Task 42 in stage 15.0 failed
4 times, most recent failure: Lost task 42.3 in stage 15.0 (TID 231125, 10.132.90.25, executor 57): Executo
rLostFailure (executor 57 exited caused by one of the running tasks) Reason: Remote RPC client disassociate
d. Likely due to containers exceeding thresholds, or network issues. Check driver logs for WARN messages.
```

The most informative chunks out of hundreds of lines of stack trace, which was actually caused by trying to gather too much data to a worker.

Scanning for Py4JavaError or the name of a python exception (e.g. AttributeError, AnalysisException) is often useful.

Notebook

- Intro to Spark
- Data Exploration
- Data Processing
- Debugging Errors
- ❑ **Building a Model**

Simple Regression Model

Given data about the trip, can we predict what the tip would be?

- **Model**: Random Forest Regressor
  - A group of decision trees, each of which predicts a tip
  - The responses are then aggregated to produce the final result

- **Metric**: Root Mean Squared Error (RMSE)
  - Take each error, square it, average them, then take the square root
  - Very roughly, a measure of “on average, how far off are we”

Notebook

Thanks all!

A recording of this class will be available soon, and full solution notebooks are available in the 'solutions' folder

Hanna Torrence

Data Scientist at ShopRunner

@HannaTorrence