Spark Streaming

Requirements

- Apache Spark 2.4.4
- Apache Kafka 2.3.0 (with Scala 2.11.12)

Setup

Compile and run the application:

```
 sbt clean assembly
```

Start Kafka, Zookeeper and Kafka servers, in two different sessions:

```
 sudo /opt/Kafka/kafka_2.11-2.3.0/bin/zookeeper-server-start.sh config/zookeeper.properties
sudo /opt/Kafka/kafka_2.11-2.3.0/bin/kafka-server-start.sh config/server.properties
```

Create the Kafka topics `cancelaciones`, `facturas_erroneas`, `anomalias_kmeans`, `anomalias_bisect_kmeans` and `purchases`:

```
 chmod +x create_topics.sh
./create_topics.sh
```

We use the –list option of `kafka-topics.sh` to verify that the topics were created correctly:

```
 /opt/Kafka/kafka_2.11-2.3.0/bin/kafka-topics.sh --list --zookeeper localhost:2181
```

Execution

Model training

First, the K-means and Bisecting k-means models must be trained:

```
 chmod +x execute.sh
chmod +x start_training.sh
./start_training.sh
```

Once the training is over, should have created the following folders and files:

- clustering/
- clustering_bisect/
- threshold
- threshold_bisect

Streaming run

Streaming pipeline application execution:

```
 chmod +x start_pipeline.sh
./start_pipeline.sh
```

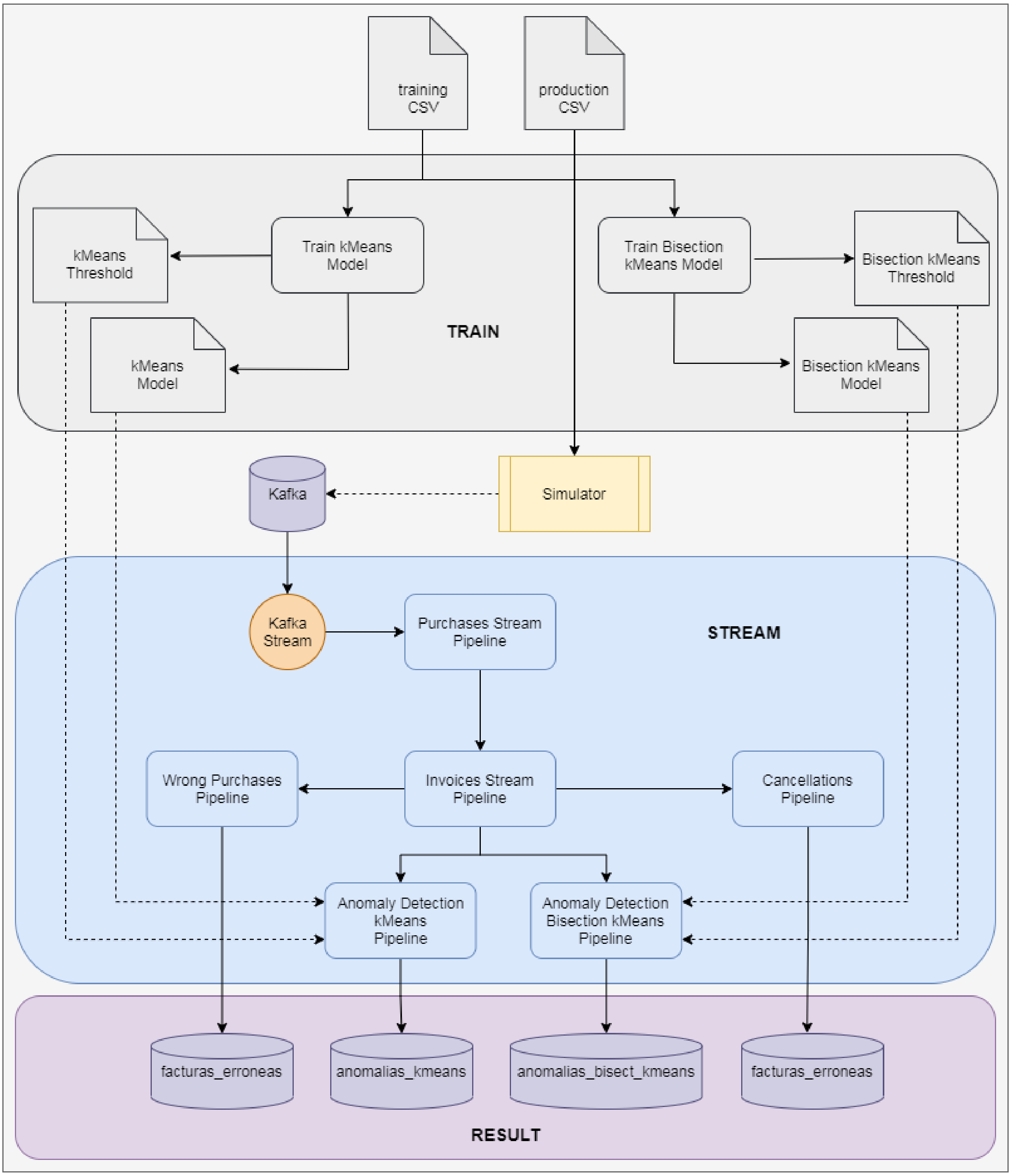
Once the streaming pipeline application is running, we can run the purchases simulator:

```
 chmod +x productiondata.sh
./productiondata.sh
```

Result

The information created/extracted by the streaming pipeline execution is saved into `cancelaciones`, `facturas_erroneas`, `anomalias_kmeans` and `anomalias_bisect_kmeans` Kafka topics.

Scenario



Credits: https://github.com/jgoodman8/streaming-retail-analysis/blob/master/assets/SparkStreaming.png