Praktikum 1 Explorative Datenanalyse des Million Song Dataset (MSD)* und Datenvorbereitung zum Clustering

Datenverständnis und Datenvorbereitung mit Spark 2.4 Visualisierungen mit Tableau 2019

Ziel des Praktikums ist es, die Datenstruktur der Metadaten und Timbre-Vektoren des MSD kennenzulernen, die Daten explorativ zu analysieren und zu visualisieren.

Pro Track wird anschließend ein Profile-Vektor zum Timbre „Loudness“ generiert, um auf den so vorbereiteten Daten in Praktikum 2 ein Clustering durchführen zu können.

Daten

Alle für das Praktikum erforderlichen Daten liegen auf dem **Hadoop des Big Data Clusters zum Download** (http://141.100.62.85:50070/explorer.html#/data/msd) für den eigenen Rechner bereit:

1. **msd10k_some-metadata** – (< 1 MB)
2. **msd10k_timbre** – (1020 MB)
3. **msd10k_timbre0.tsv** – nur Timbre 0 – (**30 MB**)
4. **msd10k_lastfm_tracks_tags** – (< 1 MB)

Hier stehen die Dateien zur Stichprobe von 10.000 Songs im tsv-Format zur Verfügung (**msd10k_**…-Dateien) und hier finden sich auch **readme-Dateien** mit einer Beschreibung der jeweiligen Datenstruktur. Des Weiteren stehen neben den o.g. Stichproben-Dateien, auch die entsprechenden Dateien aller 1.000.000 Songs zur Verfügung (**msd1m_**…-Dateien). Für die Bearbeitung der Praktikumsaufgaben liegen Ihnen diese Daten als Tabellen in Apache Hive vor. Die Tabellen sind im Parquet-Format (column based) persistiert, außer der Tabellenname hat den Suffix „_row“, dann sind es csv- oder tsv-Dateien (row based-Format).

Vorbereitung zu Hause

1. Überlegen Sie sich mindestens drei **SQL-Anfragen**, mit denen Sie (statistische) Informationen zum Datenverständnis auf den Daten der drei o.g. Datenstrukturen ermitteln können.
2. Überlegen Sie sich mindestens drei **Visualisierungsmöglichkeiten** zur visuellen Darstellung aggregierter statistischer Informationen zu den Daten der drei o.g. Datenstrukturen.
3. Gehen Sie davon aus, dass nach einem entsprechenden **Binning** (s. Teilaufgabe 3) die folgende Datenstruktur vorliegt – **ein Datensatz pro Segment eines Tracks**:

   Track_ID | Timbre0-Segmentwert | Bin

   Machen Sie sich im Vorfeld klar, welche der beiden u.g. möglichen Methoden der **Pivotierung** Sie verwenden möchten, um **einen Datensatz pro Track** wie folgt zu erhalten:

   Track_ID | relative Häufigkeit der Segmente in Bin *1* | … | rel. H. der Segmente in Bin *n*

* Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The Million Song Dataset. In Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011), 2011. pdf und bib zum Download auf der Seite Schestag / Big Data Analytics.

Bearbeitungsdatum: 10.04.19 (Mi6y), 11.04.19 (Do5y) und 17.04.19 (Mi6x)

Aufgabe 1 – Anmelden und Datenverständnis durch Abfragen mit Spark SQL

- Loggen Sie sich auf Zeppelin ein und kopieren Sie das Notebook praktikum1/_istaccount (vergleiche „KurzeEinführungzumSparkBigDataCluster.pdf“)
- Im ersten Paragraphen des Notebooks sehen Sie, wie Sie die `applicationId` Ihrer Session ermitteln können. Führen Sie diesen Abschnitt aus und überprüfen Sie auf der http://141.100.62.85:8080 welche Ressourcen Ihrer Session zugeteilt wurden.
- Wenden Sie Ihre vorbereiteten SQL-Anweisungen auf den Datenbestand an. Schauen Sie sich auch an, welche Informationen Sie durch die **describe()-Methode** für stetig-wertige Merkmale erhalten und führen Sie mit dieser einen Performance-Vergleich zwischen dem row based und dem column based Speicherformat durch.

Aufgabe 2 – Visualisierung von Informationen aus den (Meta-)Daten des MSD

a) Öffnen Sie das Tool Tableau und verbinden Sie sich mit der entsprechenden Metadaten-Datei auf dem Cluster.

   In einem sogenannten Join-Bereich sehen Sie Ihren Dateinamen – hier können Sie ggf. weitere Dateien zum Join hinzufügen. Im unteren Bereich sehen Sie die Daten selbst. Sie sollten eine Vorstellung von der Bedeutung aller zur Verfügung gestellten Meta-Daten pro Track haben – falls die Bedeutung einzelner Merkmale unklar ist, fragen Sie!

b) Öffnen Sie ein „Blatt 1“ als Ihr erstes Arbeitsblatt zum eingelesenen Datensatz.
   - In welcher **Skalierung** liegen die Merkmale des MSD vor?
   - Gibt es Merkmale mit **fehlenden Werten**? Wenn ja, wie sind diese „codiert“?
   - Generieren Sie ein **Diagramm**, das die Anzahl der Datensätze pro Jahr (als stetigwertiges Merkmal) anzeigt. Hier bekommen Sie einen Überblick, aus welchen Jahrzehnten das Gros der Titel stammt.
   - Generieren Sie auf weiteren Arbeitsblättern weitere Visualisierungen entsprechend Ihrer vorbereiteten Fragestellungen zu diesem Praktikum.

Aufgabe 3 – Generierung eines Profile-Vektors pro Track

Wir arbeiten für diese Teilaufgabe wieder in Zeppelin:

a) Aus Aufgabe 1 wissen Sie, in welchem Wertebereich das Timbre „Loudness“ (= Timbre 0) vorliegt und welche Werte dessen Minimum und Maximum haben – diese Min-/Max-Werte sind für die msd10k_timbre- und die msd1m_timbre-Tabelle unterschiedlich!

   Führen Sie ein **Binning des Timbre 0-Wertes** über die Segmente aller Tracks durch.

   Generieren Sie n Bins gleicher Breite (z.B. $n=10$) und übergeben Sie ein geeignetes Split-Array einer Bucketizer-Instanz als Parameter, gemeinsam mit den weiteren erforderlichen Parametern entsprechend der API:

   http://spark.apache.org/docs/2.4.0/api/python/pyspark.ml.html#pyspark.ml.feature.Bucketizer

   https://spark.apache.org/docs/2.4.0/ml-features.html#bucketizer

Speichern Sie die so generierten Daten mit der zusätzlichen Spalte zur Bin-Zugehörigkeit als Tabelle mit dem Namen "`<Gruppenpräfix>_bucketeddata`".

b) Erstellen Sie auf der Basis des generierten Binnings durch geeignete Pivotierung eine Tabelle, die einen **Profile-Vektor pro Track** als Datensatz enthält und persistieren Sie die so generierten Datensätze als Tabelle mit dem Namen "`<Gruppenpräfix>_profilevectors`".

Als **Pivotierungsmethode** können Sie unter den beiden folgenden Optionen wählen:

1. Pivotierung mit Hilfe von geeigneten **pivot- und aggregat-Methoden in Scala- bzw. Python**. Bitte beachten Sie die folgenden Hinweise:
   a. Eine gute Einführung in das Pivotieren mit Spark finden Sie unter dem folgenden Link: https://svds.com/pivoting-data-in-sparksql
   b. Die benötigten Methoden zum Pivotieren und Aggregieren finden Sie in der Klasse *RelationalGroupDataset*:
   http://spark.apache.org/docs/2.4.0/api/python/pyspark.sql.html#pyspark.sql.GroupedData
   c. Im Ergänzung hierzu die Methoden der Klasse *Dataframe:*
   http://spark.apache.org/docs/2.4.0/api/python/pyspark.sql.html#pyspark.sql.DataFrame

   Wichtige Hinweise: Aufgrund eines Bugs in Spark (`https://issues.apache.org/jira/browse/SPARK-12965`), dürfen die beim Pivotieren entstehenden neuen Spaltenbezeichner keinen Punkt enthalten! Deshalb ist es notwendig, die Bin-Werte 0.0, 1.0, …. wie folgt umzubenennen, bevor Sie das Pivotieren und Aggregieren anwenden. Zunächst erzeugen Sie eine Liste mit den gewünschten Bezeichnern:

   ```
   newNames = ["track_id","bin1","bin2","bin3","bin4","bin5","bin6", …]
   ```

   Als Methodenaufruf nach Pivotierung verwenden Sie dann zum Umbenennen der Spaltenbezeichner:

   ```
   df.toDF(*newNames)
   ```

2. Pivotierung mit Hilfe eines nativen **SQL-Statements ohne Pivot-Operator**
   *Hinweis: mit Hilfe der **Count(Case when … then … end)**-Anweisung und geeigneter group by-Klausel können Sie die neuen Spalten mit zugehörigem Wert pro Track generieren.*
   Die Verwendung eines PIVOT-Operators innerhalb einer SQL-Anweisung wird in Hive nicht unterstützt!

   Beim Pivotieren entsteht zunächst ein **Profile-Vektor mit absoluten Häufigkeiten** der Segmente in den Bins pro Track. Ausgehend hiervon generieren Sie dann den **Profile-Vektor mit den relativen Häufigkeiten** der Segmente in den Bins pro Track.

Aufgabe 4 – Visualisierung der Profile-Vektoren in Tableau

- Visualisieren Sie die Profile-Vektoren in Tableau auf einem neuen Arbeitsblatt.
  Achten Sie darauf, dass die entsprechenden Balkendiagramme normiert sind!

> ***Hinweis zur Normierung von Balkendiagrammen auf den Bereich [0,1]:***
> *Rechte Maus Spalte unten bzw. Zeile links (je nach Orientierung) → "Achse bearbeiten …", "Bereich": fixiert, "fester Anfang": 0.0, "festes Ende": 1.0*

- Wählen Sie aus den Songs einen „leisen" und einen „lauten" Song anhand der Lautstärkeprofile aus und überzeugen Sie sich anhand einer Hörprobe von der Güte des Profils.

Aufgabe 5 – Histogramme und Binning-Vergleich

- Überlegen Sie sich zunächst, unter welchen Verteilungsannehmen bzgl. der Daten die Wahl von Bins fester Breite gut ist und wann nicht. Finden Sie heraus, wie die Verteilung für Timbre 0 aussieht. Dabei hilft Ihnen folgendes:
  - Die histogram Funktion von PySpark
    Siehe PySpark-API http://spark.apache.org/docs/2.4.0/api/python/pyspark.html
  - Das Umwandeln der timbre_0 Spalte in ein RDD:
    df.select('timbre_0').rdd.flatMap(lambda x: x)
  - Die Plot Funktion von matplotlib.pyplot
- Generieren Sie ein alternatives Binning mit approxQuantile für timbre_0. Vergleichen Sie die entstandenen Splits mit den vorher erzeugten Bins fester Breite. Was hat sich geändert? Welches Verfahren zur Splitbestimmung würden Sie bevorzugen?

Status des eigenen Jobs nachverfolgen

In Aufgabe eins haben Sie bereits die Übersichtsseite des Spark Masters (http://141.100.62.85:8080/) verwendet. Im Folgenden sollen Sie sich anschauen welche weiteren Informationen Sie durch diese Seite gewinnen können. Starten Sie dafür in Zeppelin einen länger laufenden Job. Klicken Sie auf der 8080 in der Namensspalte auf Zeppelin entsprechend der ApplicationId Ihrer Zeppelin-Session. Schauen Sie sich die Event Timeline und den DAG des laufenden Jobs an.

Testat

Weisen Sie spätestens 1 Woche nach dem Praktikumstermin ihrem Notebook die Gruppe „Dozent" zu und laden Sie ihr erstelltes Tableaubuch als Dokument (pdf) in Moodle hoch. In Ihrer Auswertung sollten Sie folgende Ergebnisse aus dem Praktikum dokumentiert haben:

- Analysen zum Datenverständnis auf Spark mit .describe() und Spark-SQL
- Visualisierung von Analysen mit Tableau
- **Python-Script zum Binning und zur Pivotierung mit dem Ziel, einen Profile-Vektor pro Track zu erhalten als Vorbereitung zum Clustering in Praktikum 2**
- Visualisierung der Profile-Vektoren in Tableau
- Nennen Sie je einen „leisen" und einen „lauten" Song entsprechend des generierten Profile-Vektors.