Week 6 – Automatic & Semi-Automatic Music Data Labeling Report

1) Objective

To evaluate the reliability of a semi-automatic pipeline for **music data labeling** (BPM, key, instrumentation, genre, mood, etc…) on both **AI-generated** and **real classical** datasets (MAESTRO).
Goal: build a hybrid system that integrates automatic extraction and manual QA in a realistic data-annotation workflow.

---

2) Method

All experiments were run in **Visual Studio Code (VS Code)** using **Python 3.11** inside a virtual environment (`venv`).
Installed libraries: `librosa`, `music21`, `numpy`, `pandas`, `soundfile`.

Scripts used:

- `extract_bpm.py` → automatic tempo estimation
- `extract_key.py` → automatic key estimation from audio; also computes a normalized confidence score (0–1) representing the reliability of the automatic key detection.
- `extract_midi_key_instruments.py` → automatic key + instrument detection for MIDI
- `extract_mfcc.py` → automatic extraction of **13 MFCC features** (timbre fingerprint)
- `merge_extended.py` → dataset merging and CSV creation

Main files:

- **`labels_core.csv`** → core labels schema for AI-generated and MAESTRO tracks (current dataset: 6 AI-generated, 3 MAESTRO); manual + automatic metadata (MFCC excluded)
- **`ai_tracks_auto_bpm.csv`**
- **`ai_tracks_auto_key.csv`**
- **`ai_tracks_mfcc.csv`**
- **`maestro_tracks_auto_bpm.csv`**
- **`maestro_tracks_auto_key_instr.csv`**
- **`maestro_tracks_mfcc.csv`**

Columns overview (labels core):
`title, source, source/model, duration, audio_link, bpm_manual, bpm_est, key_manual, key_est, instruments (manual, auto, merged), genre, mood, confidence, error, usage_context, notes`

Notes:

- MFCC features are stored in separate CSV files and are not part of the core labels schema.

- `instruments_merged` = merged column combining **manual (AI tracks)** + **automatic (MAESTRO MIDI)** detection.
- `source =` generated automatically based on each file's directory : files from the data/ folder were labeled **"AI_generated"**, and files from the data_external/maestro_subset/ folder were labeled **"MAESTRO"**.
- `source/model` = Specific AI model/version and MAESTRO dataset.
- `confidence` = automatic confidence score (0–1) for audio-based key estimation, computed by extract_key.py (librosa).
- `error` = technical error log (if any).

---

3) Results

Automatic vs Manual Comparison

- **BPM detection:** mostly accurate but with several double-time or pulse-misaligned readings (e.g., 100 → 198.8; 50 → 99.4; 40 → 120 BPM). Expressive rubato and non-percussive textures (Tracks 3, 4, 6, 8) caused inconsistent beat estimation.
- **Key detection:** highly consistent overall. Exceptions : Track 6 - Intro in Gb major; sections B–C shift to Eb minor (relative modulation); auto key Eb minor coherent. Track 7 (MAESTRO) : Auto key (E minor) matches central section only; overall key G major.
- **Instruments detection (for MAESTRO tracks MIDI) :** correctly detected *Piano* for two files; one mis-detection ("Denisova11-15.MID") corrected manually to *Piano*.

---

4) Track-Level QA Highlights (MAESTRO)

Midi Unprocessed 43 — C major: Some tonicization in the central section, returning to C major; expressive rubato.
Manual BPM ≈ 161 (automatic ≈ 161 → correct).
Key detection OK (C major); instrument = Piano OK.

Midi Unprocessed 41 — E major: solid tonal center with short tonicizations; expressive rubato.
Manual BPM ≈ 73 (automatic 112 → overestimated).
Key detection OK (E Major); instrument = Piano OK.

Midi Unprocessed 02 — G major: overall structure G major (A/A′) → middle section E minor.
Automatic Key detection E minor → plausible relative key;
BPM ≈ 139.7 (auto) vs 142 (manual) → acceptable variance.
Instrument auto = "Denisova11-15.MID" → corrected manually to Piano.

---

6) What This Demonstrates

- Realistic **audio ML pipeline** integrating feature extraction, metadata fusion, and manual QA.
- Practical use of **VS Code + Python + Librosa/music21** for music annotation.
- Semi-automatic alignment of BPM/key across AI and real datasets.
- Foundation for **Week 7 segmentation & QA simulations**.