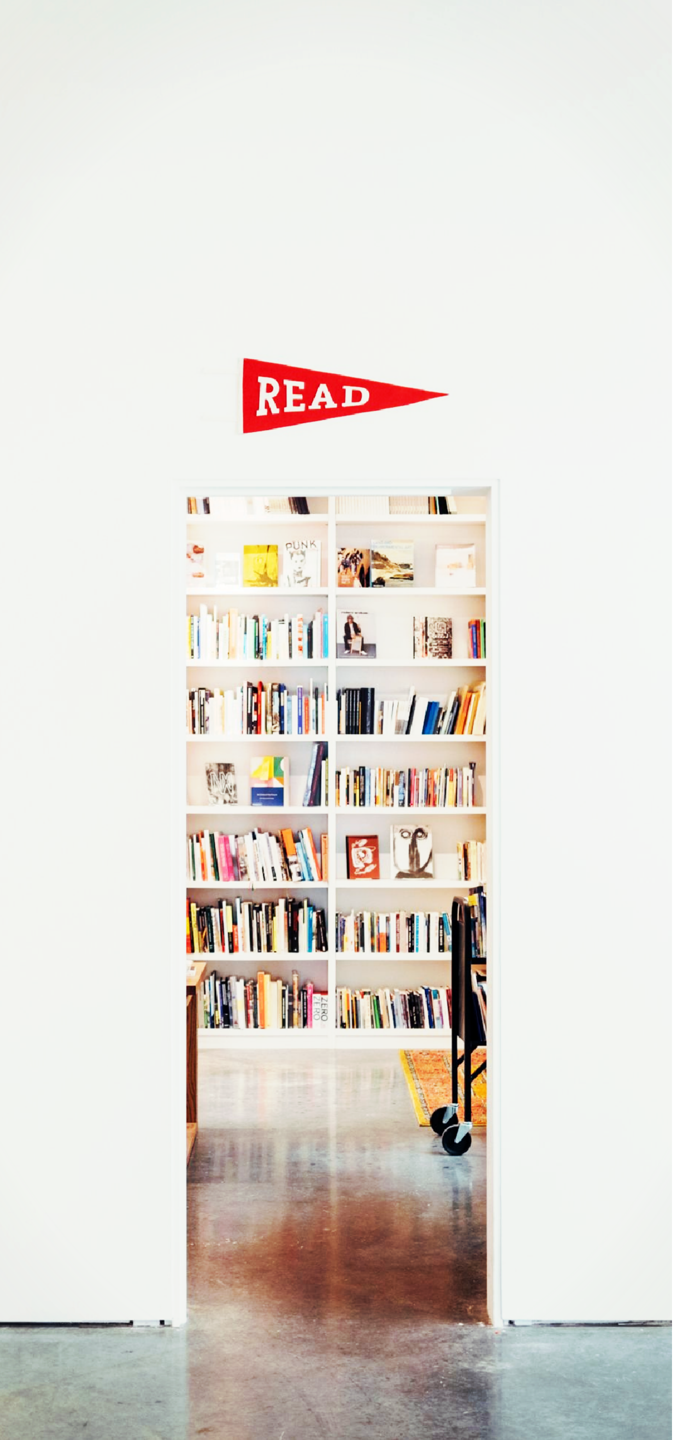
# MUSIC GENRE CLASSIFICATION

Corvin Idler | 23.6.2022

Victoria University Wellington, AIML 427, T2-2022, Assignment 3 - Individual Project

MADE WITH
beautiful.ai




# CONTENTS



MADE WITH
beautiful.ai

# THE FREE MUSIC ARCHIVE (FMA)

The FMA consists of A LOT! of Creative Commons licensed audio. https://freemusicarchive.org/home

Luckily somebody extracted a lot of audio features already: https://github.com/mdeff/fma

## FMA in numbers

106,574 tracks from 16,341 artists and 14,854 albums, arranged in a hierarchical taxonomy of 161 genres grouped together into 16 top level genres

## Question

Can we do some music genre classification with these audio features?

## Answer

Yes... well.. I guess it depends :)

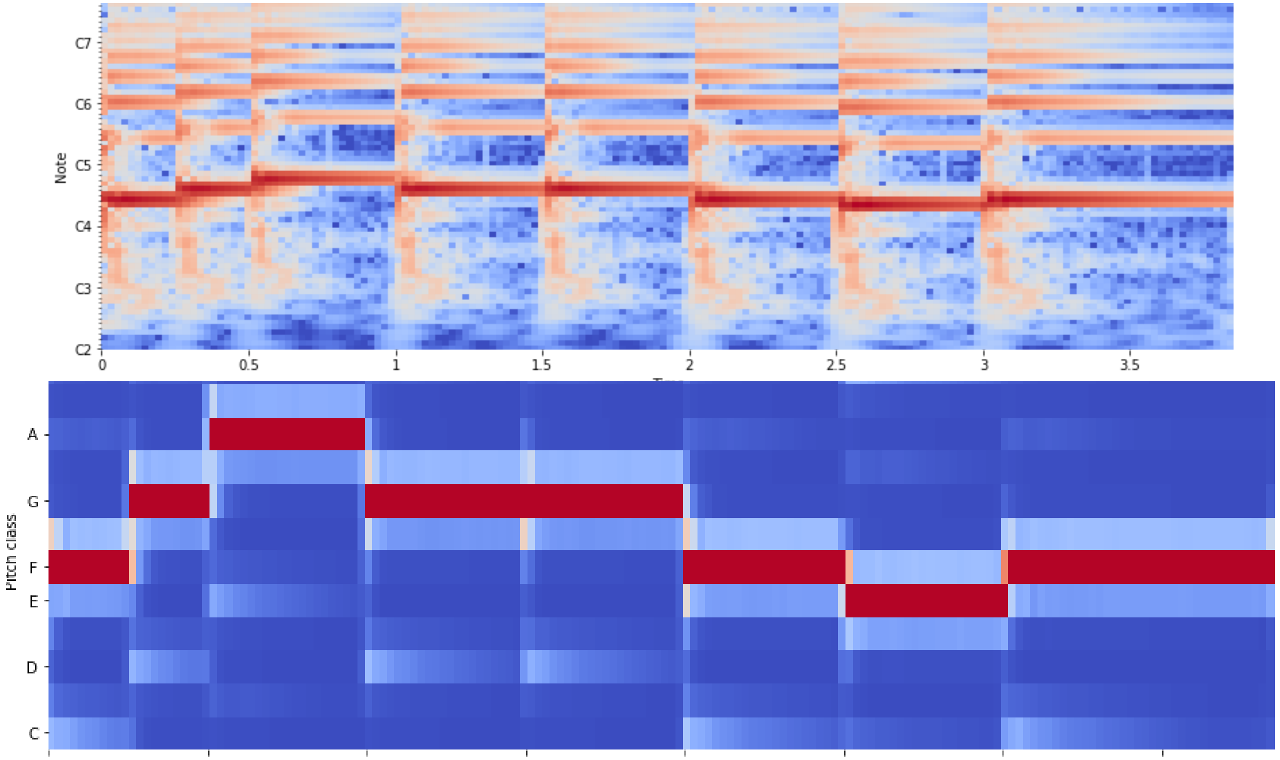
## Detailed results

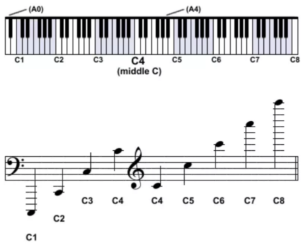
Fasten your seatbelt and stay tuned :)

MADE WITH
beautiful.ai

# AUDIO FEATURES PRIMER

A spectogram (e.g. Fourier Transformation) is turned into a chromatogram when
“broken down” by the 12 pitch classes (C, C#, D, D#, E, F, F#, G, G#, A, A#, and B)
Each of these pitch classes exist several times (an 88-key piano has seven octaves)
The chromatogram tries to capture the energy for all octaves but grouped into pitch classes.




Octaves

MADE WITH
beautiful.ai

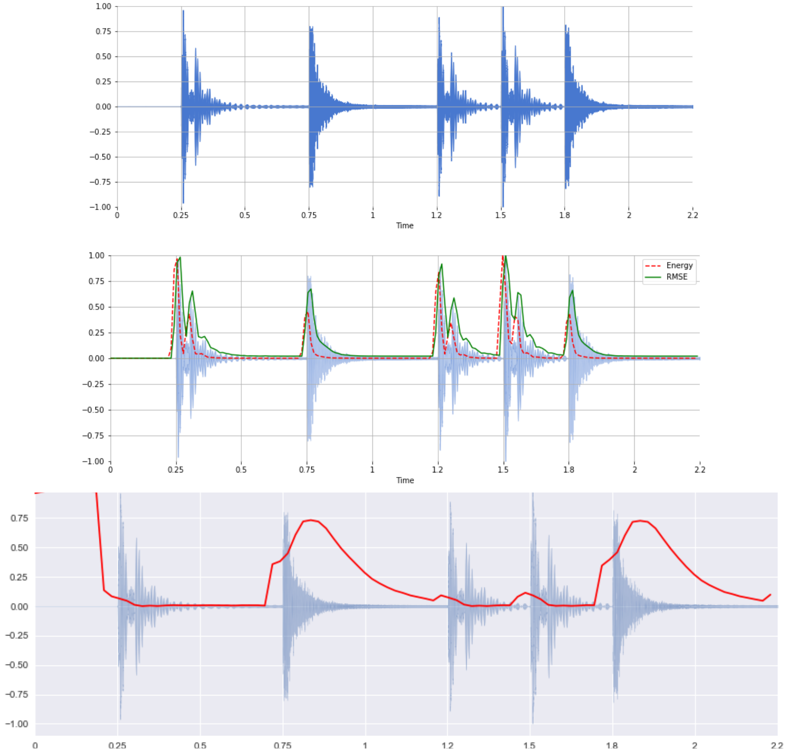
# AUDIO FEATURES CONTINUED

There are also slightly less sophisticated feature one can calculate

Zero crossing rate, Root Mean Square Energy, Spectral Centroid, Spectral Bandwidth, etc.

These features need to cover the whole song. So if calculated for many time slices we get a distribution not a scalar!

-> kurtosis, max, mean, median, min, skew and standard deviation were calculated to create scalar features for a whole song




Have a look at

https://github.com/librosa/librosa

https://github.com/mdeff/fma

https://musicinformationretrieval.com/

MADE WITH
beautiful.ai

# EDA / PRE-PROCESSING

Quite some NaNs for the Genres, and not all features particularly “useful”

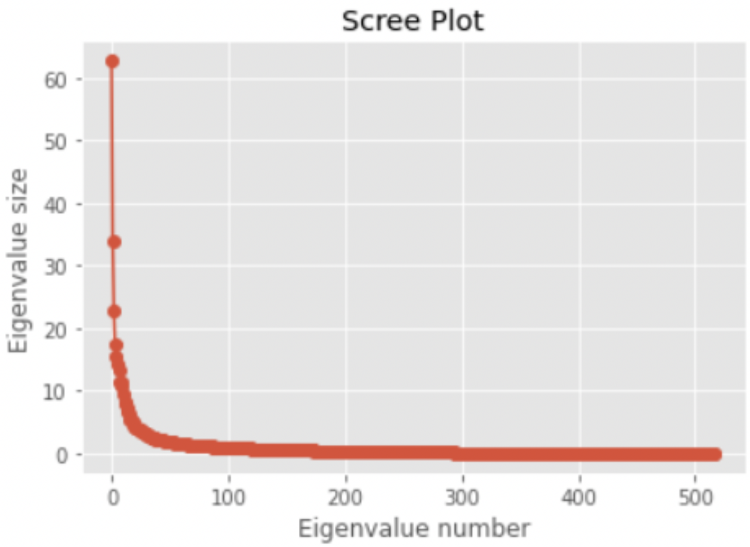
| Genre | count |
|---|---|
| NaN | 56976 |
| Rock | 14182 |
| Experimental | 10608 |
| Electronic | 9372 |
| Hip-Hop | 3552 |
| Folk | 2803 |
| Pop | 2332 |
| Instrumental | 2079 |
| International | 1389 |
| Classical | 1230 |
| Jazz | 571 |
| Old-Time/Historic | 554 |
| Spoken | 423 |
| Country | 194 |
| Soul-RnB | 175 |
| Blues | 110 |
| Easy Listening | 24 |

| feature | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|
| chroma stft.max.12 | 0.99 | 0.003 | 0.79 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.10 | 0.99 | 0.003 | 0.51 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.01 | 0.99 | 0.004 | 0.66 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.05 | 0.99 | 0.004 | 0.68 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.03 | 0.99 | 0.004 | 0.54 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.04 | 0.99 | 0.004 | 0.49 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.06 | 0.99 | 0.004 | 0.66 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.02 | 0.99 | 0.005 | 0.72 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.11 | 0.99 | 0.005 | 0.58 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.08 | 0.99 | 0.006 | 0.53 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.09 | 0.99 | 0.006 | 0.58 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma stft.max.07 | 0.99 | 0.007 | 0.48 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma cqt.max.01 | 0.99 | 0.007 | 0.49 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma cqt.max.03 | 0.99 | 0.009 | 0.32 | 1.0 | 1.0 | 1.0 | 1.0 |
| chroma cqt.max.04 | 0.99 | 0.009 | 0.31 | 1.0 | 1.0 | 1.0 | 1.0 |

...

MADE WITH
beautiful.ai

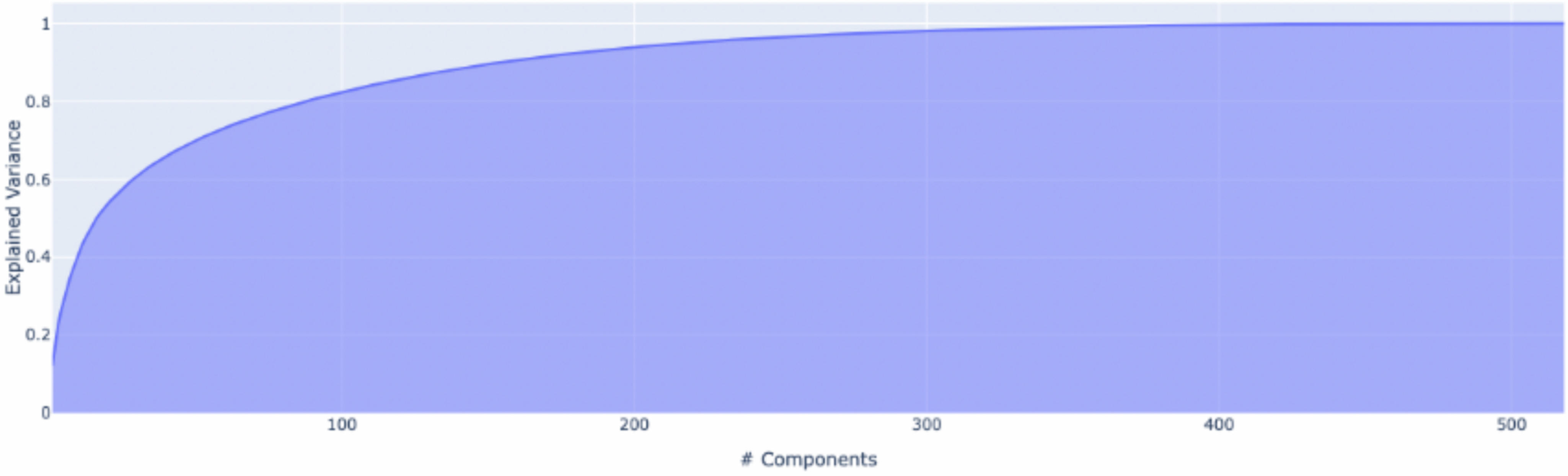
# PCA RESULTS




"elbow" in Scree plot around 20-30 factors

but ca. 100 factors needed for 80% variance...

so a loooooong tail...



# GRADIENT BOOSTED DECISION TREES

LightGBM

-multi class

-Gradient-based One-Side Sampling

-Mutual exclusion feature binding

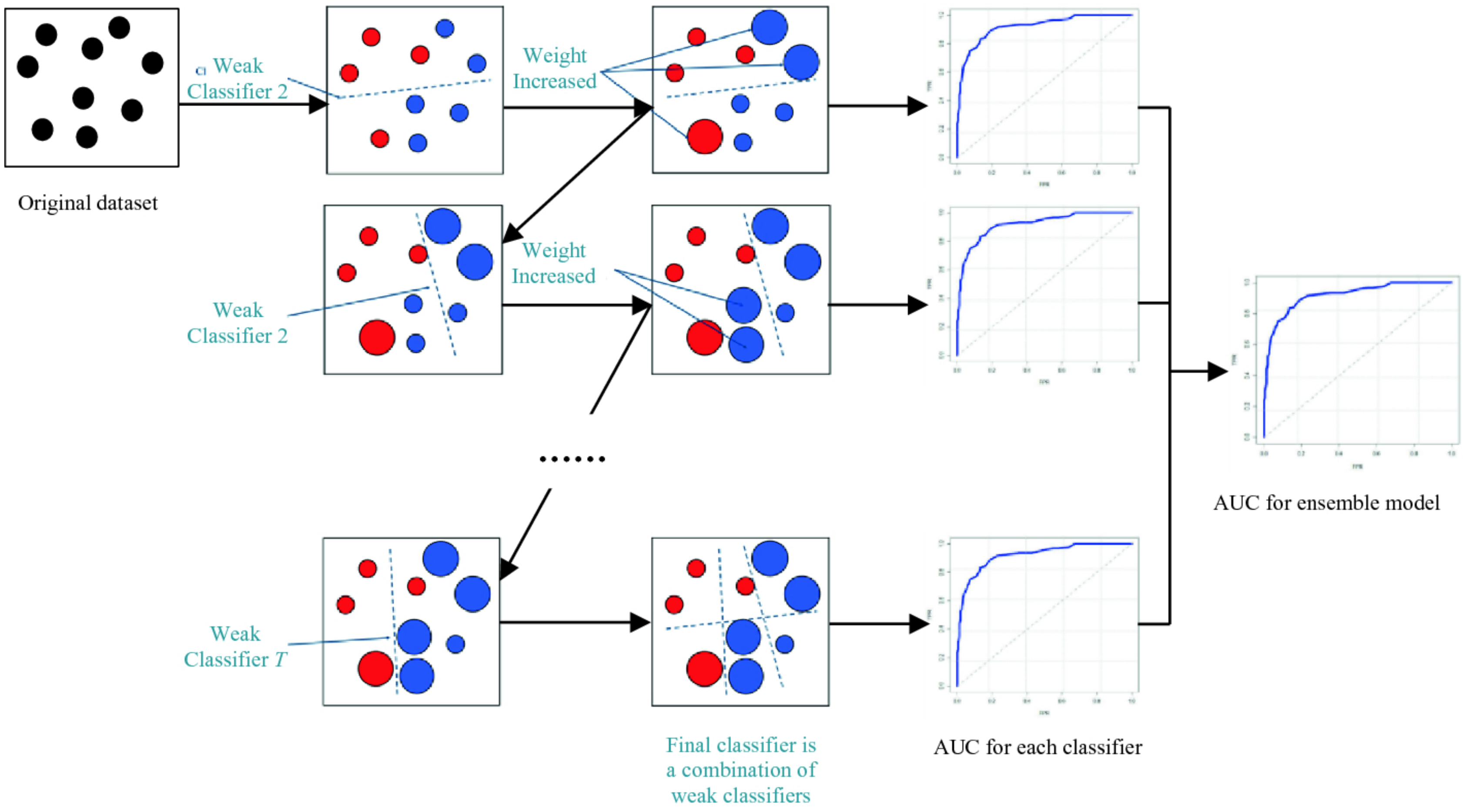
-runs on "anything" when installed

Spark-ML GBDT

- only binary classification :(

- runs on native spark

(cluster e.g. for x-val)



# NOT MUCH TO BE SEEN!?

A CLOSER LOOK:

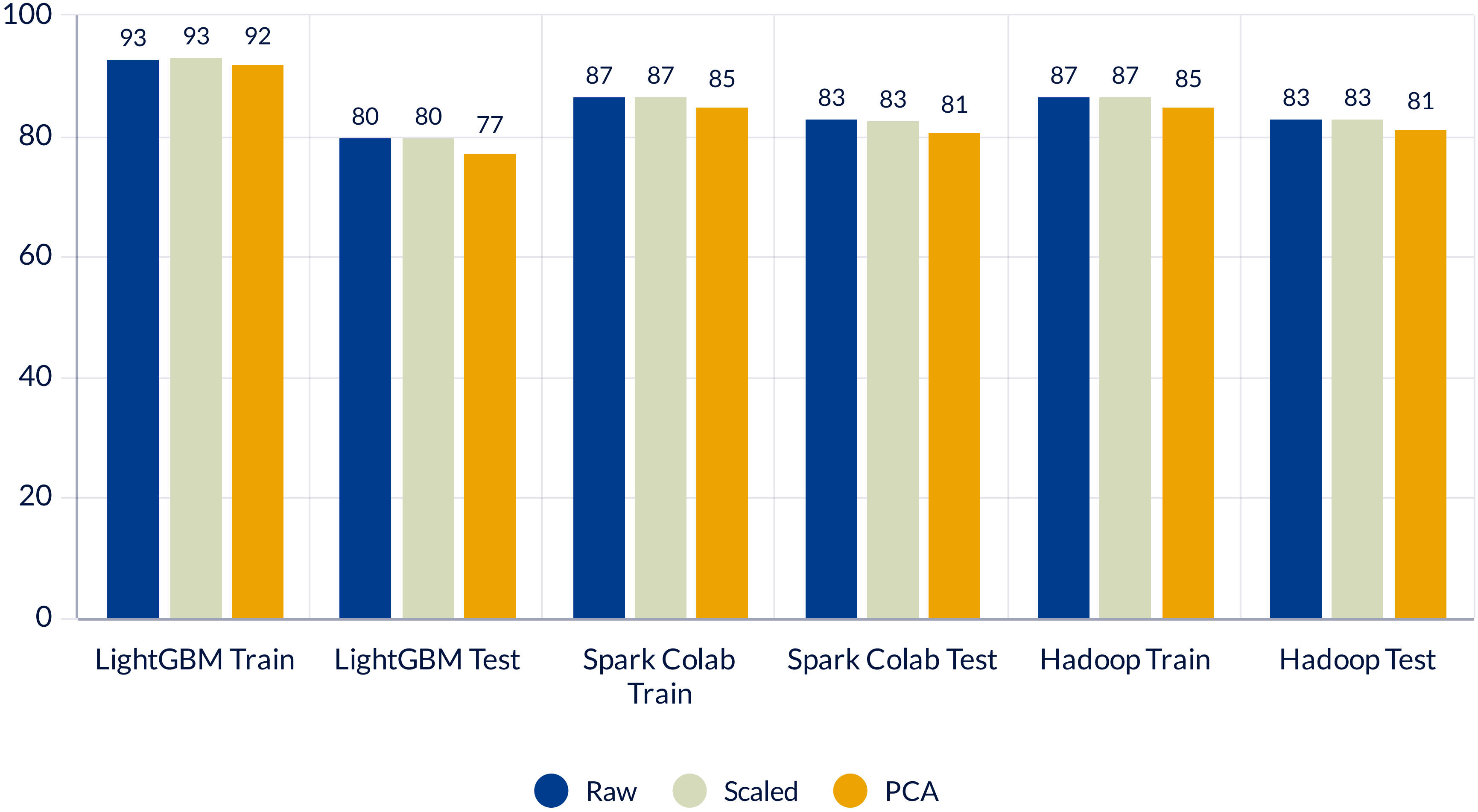
LightGBM (100 trees) overfit more than Spark (20 trees)

Raw or scaled doesn't matter much for a tree

PCA is always hurting. Makes sense for GBM

## RESULTS - ACCURACY

Train vs. Test and Raw vs. Scale vs PCA




MADE WITH beautiful.ai

# RESULTS - CONFUSION MATRIX

LightGBM results

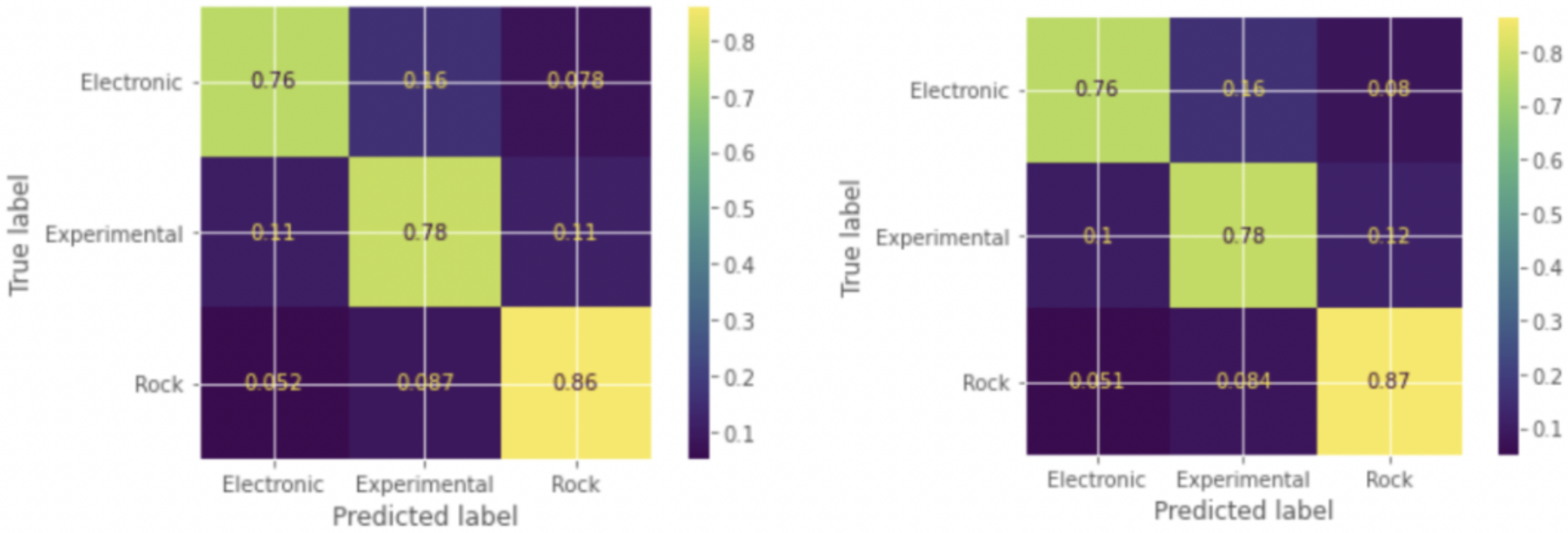



Figure 5: Confusion matrices for raw features (left) vs. scaled/normalized (right)

# WHAT ELSE?

A CLOSER LOOK:

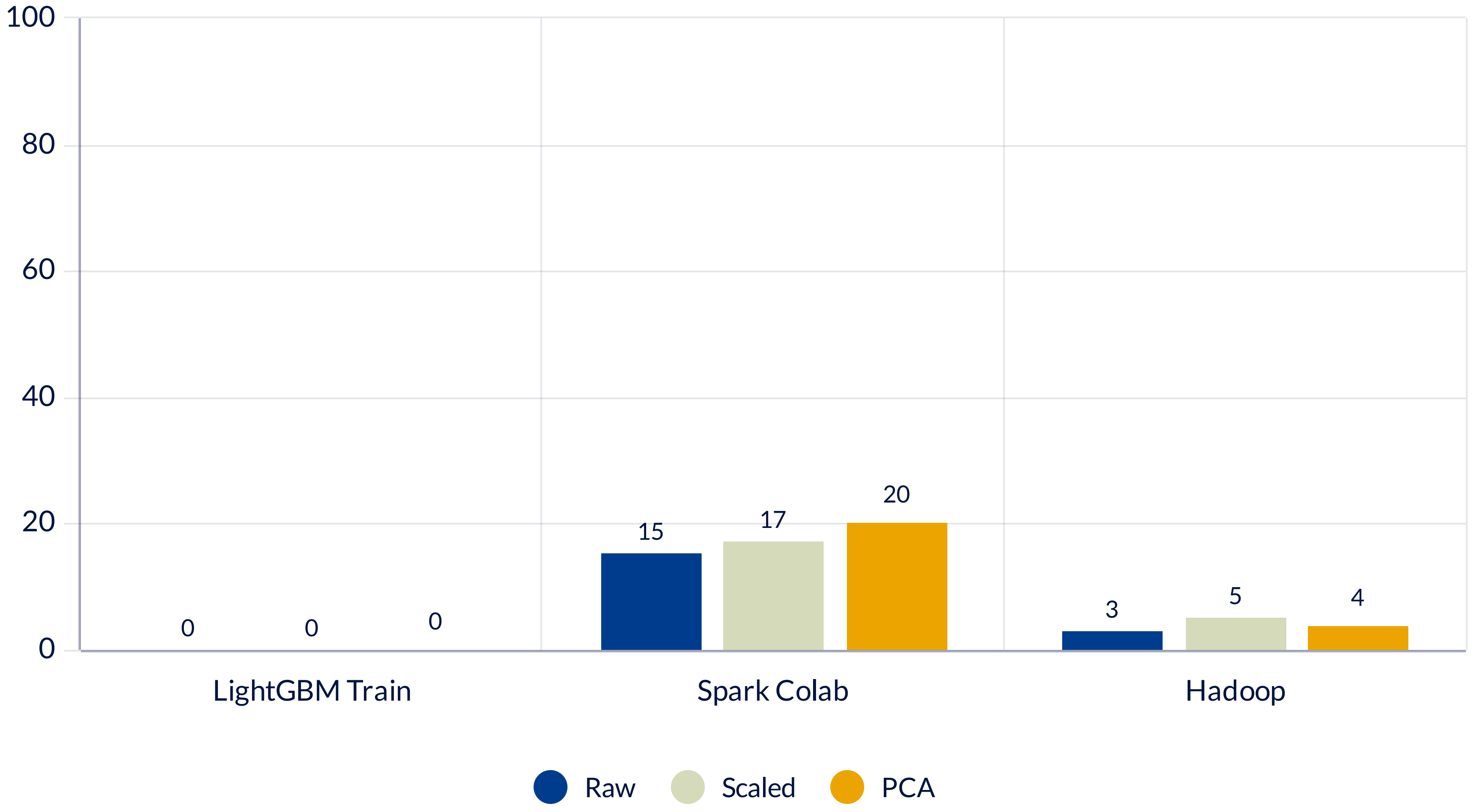
LightGBM (100 trees) but waaay faster

Colab is free but certainly not overperforming...

Hadoop takes still quite a bit.. Maybe random forrest better for distributed computing

## RESULTS RUN TIME

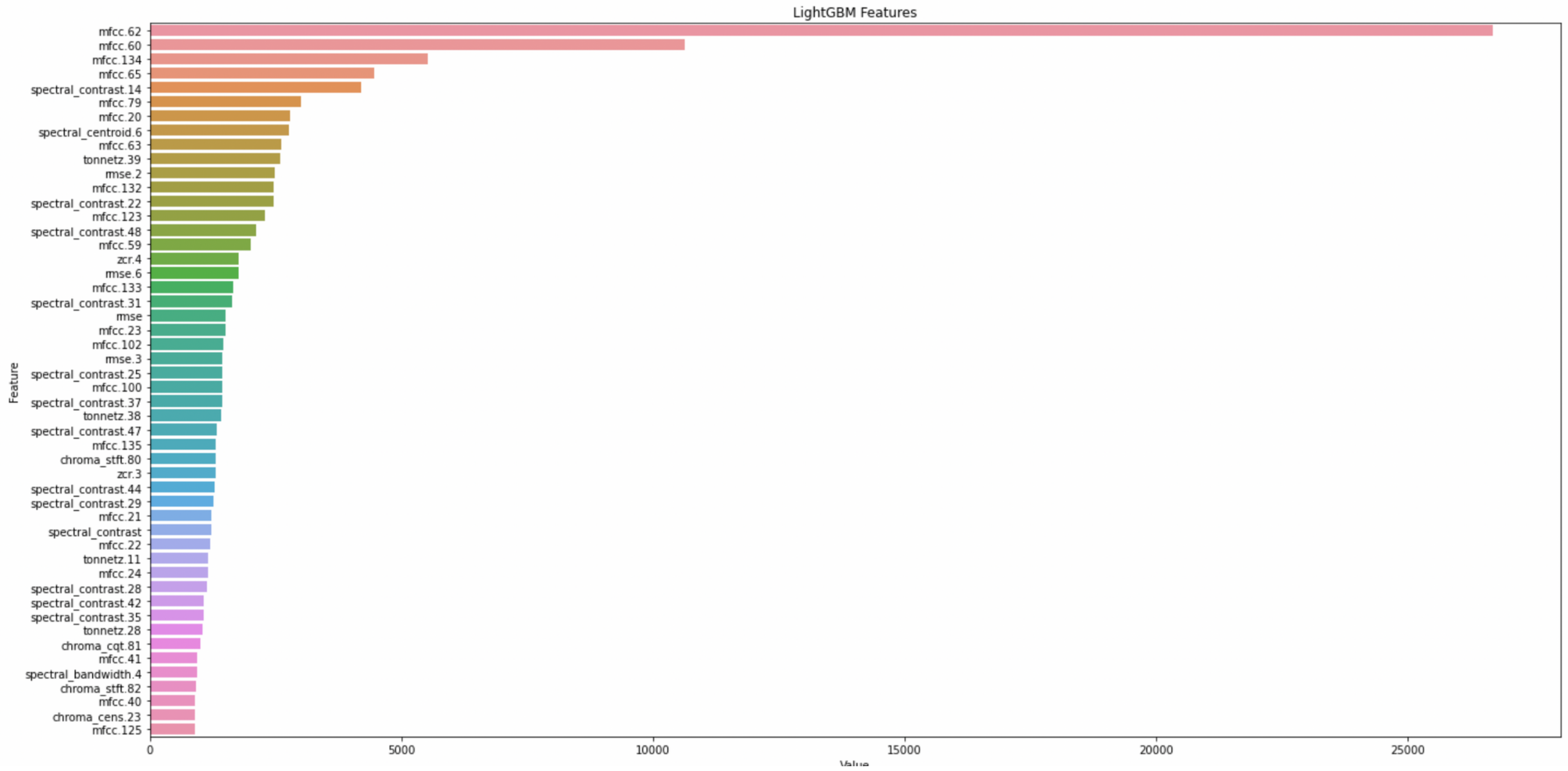
timing matters... values in minutes




MADE WITH beautiful.ai

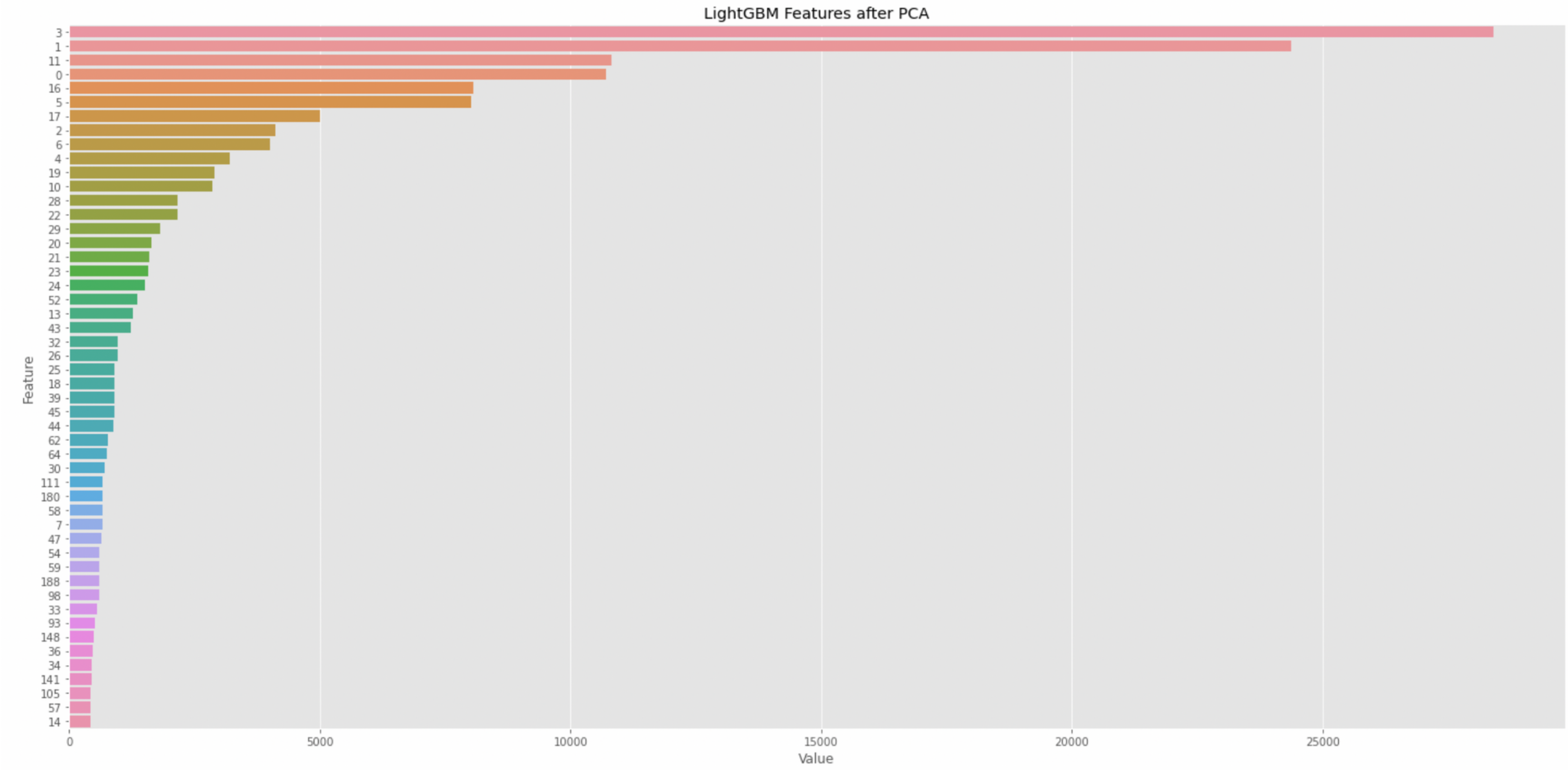
# RESULTS FEATURE IMPORTANCE

## Feature importance - raw features



# RESULTS FEATURE IMPORTANCE PCA

## Feature importance - principal components



# CONCLUSIONS


a highly optimized algorithm on a desktop might actually out-compete an “aged” implementation on a cluster in terms of runtime


Point #2

A cluster might come in handy when the task is highly parallelizable. E.g. random forest vs. GBDT or hyper-parameter optimization vs. single model.


The principal components with the largest eigenvalue might not at all be the most important feature.

That depends on the data set and class distribution in input space and the classification algorithm.mendations

MADE WITH beautiful.ai

# QUESTIONS?

MADE WITH
beautiful.ai