PROJECT WORK IN SF2935:
MUSIC CLASSIFICATION

1. Group organisation and guidelines

The project work is to be done in groups of 2–5 students - exceptions from this must be cleared by the instructor *before* the project work begins. Each group must send an email to Pierre via Canvas no later than **September 24**, containing

- a list of group members (first and last name)
- the name and email of a group representative

The group representative will be responsible for all correspondence between the instructors and the group. Information to the group will only be sent to the group representative, who in turn is responsible for relaying relevant information to the other group members. Each group will be given a group ID that should be used when submitting the final report.

1.1. **Grading.** All members of a group will receive the same grade (pass/fail) for the project. All members of a group are equally responsible for the work of the group and are obliged to equally contribute to it. If a group member is of the opinion that this principle does not apply to their group, please contact the instructors.

2. Project : Music classification

The aim of the project is to classify songs in a test set (available later in the course) according to whether Pierre would like them or not. To aid in this task you have a training set with close to 500 songs, each which has been labelled '1' (Like) or '0' (Dislike). Neither dataset contains explicit information about the songs, such as artist, title or genre, but rather high-level features associated with the songs and available via Spotify's web-API[1]. The high-level features include acousticness, danceability, valence, tempo etc.; the description in Table 1 is available form Spotify's documentation.

2.1. **Data sets.** The training data is available on Canvas as `project_train.csv`. The test set will be made available later on in the course as `project_test.csv`. In both cases the columns in the tables correspond to the high-level audio features described in Table 1 and specified by the headers in the files. The labelling with '0' and '1' corresponds to 'Dislike' and 'Like', respectively.

2.2. **Specific tasks.** In this project you will make use of several of the classification methods discussed in the course. You must use **at least** three different methods. To distinguish between different methods, you can pick from the following 'families':

- logistic regression,
- discriminant analysis (LDA, QDA),
- *k*-nearest neighbour,
- support vector machines,
- tree-based methods (classification trees, random forests, bagging),
- neural networks.

[1] You can try this feature out for yourself to get a sense of what features different songs have

Name	Description
danceability	Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.
energy	Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.
key	The key the track is in; integers map to pitches using standard Pitch Class notation (e.g. 0 = C, 1 = C major/D minor, 2 = D etc.)
loudness	The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typical range between -60 and 0 db.
mode	Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived.
speechiness	Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.
acousticness	A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.
instrumentalness	Predicts whether a track contains no vocals. ”Ooh” and ”aah” sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly ”vocal”. The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.
liveness	Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.
valence	A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).
tempo	The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.

Table 1. Description of high-level audio features from Spotify’s documentation

Within the families you choose, you are to test at leat one method.

For each method you choose, you are to implement the method and tune it to perform as well as possible. The implementation can be done by writing your own code or using existing packages (any programming language may be used). Once this is completed for each method, you are to **select one** to be put ‘in production’. That is, you select one method to run on the test data and report the results in Canvas. This is done by submitting a string of 1s and 0s according to the predicted labels.

For groups that want to take part in friendly competition within the course, there will be a leaderboard on Canvas showing which group has achieved the best prediction up to that time. You are only required to submit your results once. However, if you notice that other groups are outperforming you and want to try to improve, you are allowed to submit new results *at most once every 3 days* until the deadline—on the final day all groups are allowed a final submission regardless of when the previous set of results were submitted.

The group with the leading classification score at the deadline for the project will receive a prize for their efforts. Additionally, the ”SF2935 Spotify Playlist”, containing the tracks that are used to create the test set, will be released at the deadline as well.

3. Report

The work for the project is to be summarised in a report that is then graded (P/F-basis) by the instructors. This report should include the following:

- A brief problem statement and introduction.
- A brief description of the methods you consider and details on how they are applied to the specific problem under consideration. In particular, you should explain what any pre-existing methods that you use in your preferred programming language *actually do* (e.g., there are implementations of regressions in Python that applies a regularising term

when estimating the parameters, which is different from the standard approach). These descriptions should be your own—do not copy-paste—and be rather brief.

- A description of how the chosen methods are applied to the data. E.g., which inputs where used and how, how did you tune the parameters in your model etc., along with motivation for your choices.
- An evaluation of how each method performed on the training set.
- A statement and motivation of which method you select for going into production.
- The final result for your chosen method (obtained from instructors after your first submission of predictions)
- Conclusions.
- In an appendix, place all the code needed to reproduce your results.

The final report must be submitted as a PDF and be namned according to:

SF2935Project-Group-ID

where 'ID' refers to the ID you receive upon registering your group (see above).

3.1. **Submission.** The final report is to be submitted—by one group member—in Canvas as part of an assignment ('Project report'). Before the final report you must submit your predictions via email to Pierre. The subject line should read

SF2935 Project predictions Group-ID

where 'ID' is as above. You should submit your result well in time to finish your report before the deadline (see next section), however there is no additional deadline for submitting the predictions.

4. Schedule and deadlines

- September : Email group information (see above) **no later than September 24**.
- October: Deadline for submission of final report is **Friday October 22 (23:59)**.

Good luck!