Project Proposal

Class Title: Machine Learning 1

Professor: Dr. Casey Bennett

Group Number: 6

Members:

Bi Meng Zhou 2021038731

Milton Eduardo Rios Castillo 9047420222

Matan Nahmani 2021073763

Introduction

While discussing with our team members, we concluded that we all love anime. But we all face the same difficulty of finding a good show to fit our preferences.
As avid anime watchers, we find ourselves troubled while searching for a show, spending countless hours without success. We decided that we wanted to tackle this problem and create a tool that can be used not only by ourselves but by the anime community and newcomers.
Anime has many genres to choose from, and everyone has a different taste. As more and more shows emerge, finding the one show, you will fall in love with becomes more challenging. This project aims to develop an effective model for recommending individualized anime lists based on distinct characteristics. There is some study for music recommendation that has the same notion since music includes features such as genre and kind. The method used in the research is KNN, which stands for k-nearest neighbors algorithm, which is a form of an algorithm for comparing things. (1. G. Li and J. Zhang)

We plan to utilize the Kaggle data collection for this research, which comprises information on user preference data from 73,516 people on 12,294 anime. This data set is a collection of such ratings. Each user may add anime to their finished list and give it a rating. We'll propose anime depending on what the user has already seen. We'll train the model using the KNN model to inspect how similar the anime is to users and others from the dataset; we will also explain the process of KNN and why it matters. (2. Harrison)

Data Source:
https://www.kaggle.com/datasets/CooperUnion/anime-recommendations-database

---

Feature Explanation

There are two datasets inside the project:

1. Anime.csv - which includes the information on the anime
2. Rating.csv - which includes the rating of the anime from the user

Anime.csv:

1. Anime_id - myanimelist.net's unique id identifying an anime.
2. Name - full name of anime
3. Genre - comma-separated list of genres for this anime.
4. Type - movie, TV, OVA, etc
5. Episodes - how many episodes are in this show. (1 if movie).
6. Rating - an average rating out of 10 for this anime.

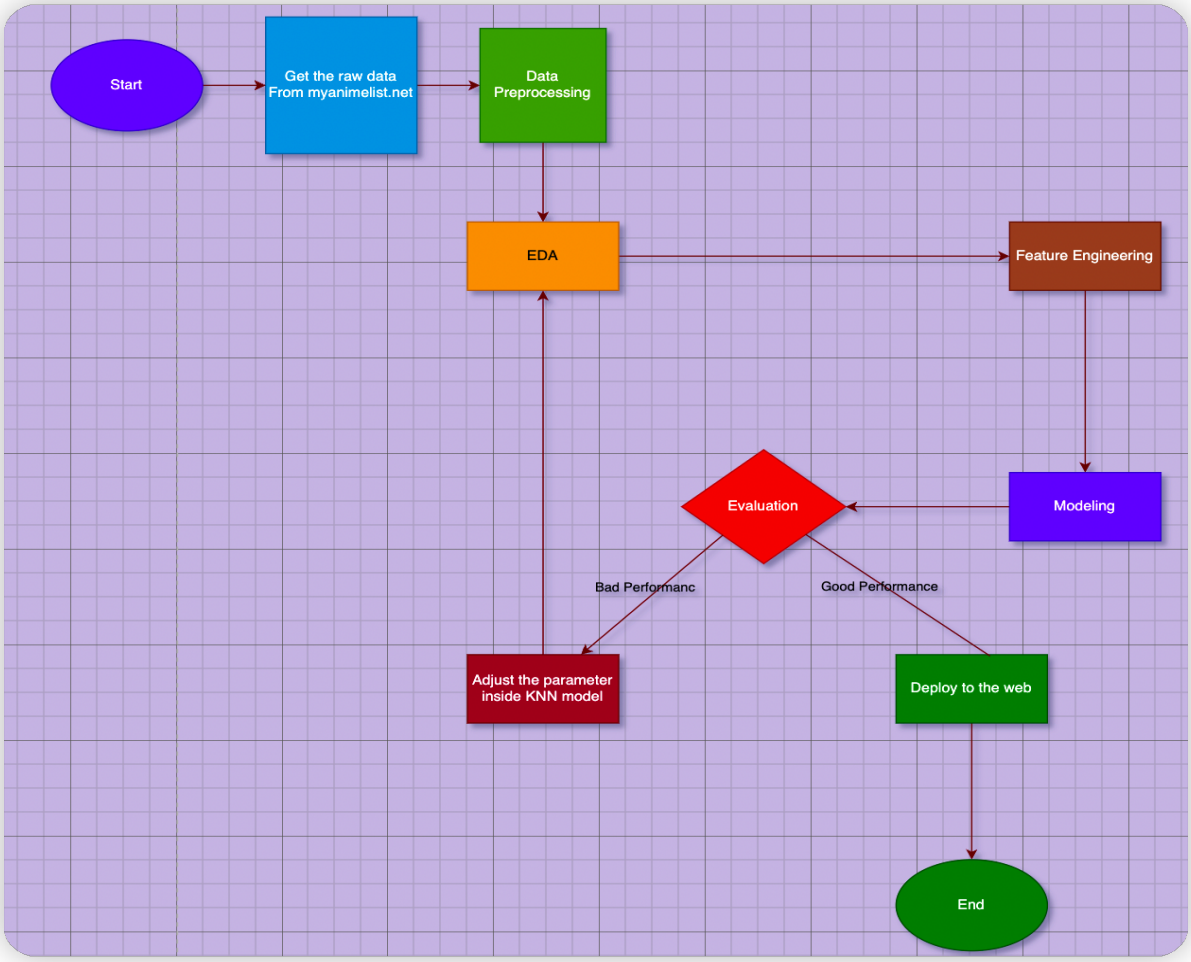
7. Members - number of community members that are in this anime's "group"

Rating.csv:

1. User_id - nonidentifiable randomly generated user-id
2. Anime_id - the anime that this user has rated
3. Rating - a rating out of 10 this user has assigned(-1 if the user watched it but didn't assign a rating)

Methodology

Plan Map



Data Preprocessing

Data Cleaning

We start detecting and removing corrupt or inaccurate records from our database. In this case, we will only choose the important ones. If we have a few missing data, we could drop them, and if we have a lot, we can do an average.

After cleansing, a data set should be consistent with other similar data sets in the system. The inconsistencies detected or removed may have been initially caused by user entry errors, corruption in transmission or storage, or by different data dictionary definitions of similar entities in other stores. Data cleansing differs from data validation in that validation almost invariably means data is rejected from the system at entry and is performed at the time of access rather than on batches of data.

Data Transformation

In over way of converting our data from one format to another. The steps of data transformations are converting raw data into a clean and usable form and converting CV. During data transformation, an analyst will determine the structure, perform data mapping, extract the data from the source, execute the transformation, and store the data in an appropriate database.

Data Reduction:

This has a capacity optimization technique in which data will reduce to its simplest possible form to free up capacity on a storage device. There are many ways to reduce data, but the idea is very simple—squeeze as much data into physical storage as possible to maximize capacity.

EDA

We are using two data sets in this project, the anime.csv, which includes the information on the anime. The second one includes the rating of the anime from the user. We will find the hypotheses between each feature from a different dataset during the EDA process. A box plot (or box-and-whisker plot) shows the distribution of quantitative data in a way that facilitates comparisons between variables. The box shows the quartiles of the dataset while the whiskers extend to show the rest of the distribution.

The box plot (a.k.a. box and whisker diagram) is a standardized way of displaying data distribution based on the five-number summary: Minimum, First quartile, Median, Third quartile, and Maximum.

Feature Engineering

This project will use K-NN as predictors/classifiers in the machine learning field. K-NN works with only numerical values. Any other types of data, such as factor variables, should be converted to an appropriate format via, for example, one-hot encoding. K-NN is a non-parametric and, as mentioned, a statistical learning model. So it is ideal for our data set.

The most important characteristics of a K-NN model are how the similarity is defined and how many neighbors you will exploit in predicting a new record. In linear regression, the K-NN uses the neighbors' average, while in classification methods; it uses the majority rule (probability of cut-off of 0.5). However, we can use any cut-off threshold to our benefit to fit the application. For example, in imbalanced data, we can set the cut-off point to the frequency of the rare data.

Modeling

We use the python language to implement the K-NN to model from the data sets we processed before (the cleaned dataset). The data set can be found here. We aim at using the Sci-kit Learn in python library. We adopt the anime dataset anime and recommendations. The full dataset can be downloaded from Kaggle. The dataset consists of 7. The target response is unsurvived. Please note that the factor variables that take a limited value level have already been converted via one-hot encoding. In addition, we will look at the parameters inside the KNN model from scikit-learn API and do some background research to have a deep comprehension of the parameter and try to deploy on the model and use the evaluation to receive the performance and do the comparison.

Evaluation

In this part, we use cross-validation to verify whether the model is generalized well or not.

Cross-Validation in Sklearn is very helpful for us to select the correct Model and Model parameters. By using that, we can intuitively see the effect of different Models or parameters on structural accuracy. The first way should be to use knn.score() to see the accuracy; then using for classification, the original sample is randomly partitioned into k equal size subsamples. The third way we are going to use is neg_mean_squared_error, seen as good for regression.

Future Plan

In this project, we plan to create a dynamic recommendation system that can be expanded in the future to make more accurate recommendations based on attributes like past shows (personalization) and feature selection by the user.

(ex: allowing to exclude categories from search and search only specific attributes like airing year, myanimelist rating range, and more)

We took inspiration from a few sites: (3* Collaborative Recommendation System in Users of Anime Films)
AniBrain RandomAnime

Reference

1. G. Li and J. Zhang, "Music personalized recommendation system based on improved KNN algorithm," 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2018, pp. 777-781, doi: 10.1109/IAEAC.2018.8577483.

2. Harrison, O. (2019, July 14). *Machine Learning Basics with the K-Nearest Neighbors Algorithm*. Medium. https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

3. Collaborative Recommendation System in Users of Anime Films https://iopscience.iop.org/article/10.1088/1742-6596/1566/1/012057/meta