Bachelorarbeit Johannes Effenberger

1 Introduction

Microbial community compositions in der Form von OTU Tabellen geben detaillierten Aufschluss über die Taxonomische Aufteilung in einem Biosample. Datenbanken wie Mgnify oder ENA enthalten tausende von Samples sowie die dazugehörenden Metadaten. So gibt es Alleine auf MGnify ca. 350.000 Samples mit ca. 450.000 Analysen welche fast 500 Biomen zugeordnet werden. Da es sich bei solchen Tabellen schnell um große Datenmengen handeln kann, ergibt sich hier eine Gelegenheit mithilfe von Maschinellem Lernen verschiedene Datengruppen zu klassifizieren. Eine solche Klassifizierung hat viele Anwendungsmöglichkeiten, wie etwa das automatische einschätzen neuer Samples, das unterscheiden von Biomen oder Herkunft eines Samples oder das Erkennen von Krankheiten anhand von kleinen Unterschieden in Proben von gesunden und erkrankten Menschen.

Ein Beispiel für einen Klassifizierung von Erkrankungen auf mikrobiom Daten ist DeepMico(REFERENZ PAPER DM), welches auch die Grundlage für diese Arbeit bietet. In der Klassifizieung von Erkrankten mithilfe von DeepMicro wurde preprocessing(++WAS GENAU PAPER) der Daten angewendet. Zudem hat es sich bei den Genutzen Daten um Gesunde sowie Erkrankte Menschen gehandelt. Da es sich bei Metagenomischen Datensätzen oft um große Datenmengen handelt steht die Vermutung da, das ein stärkeres Preprocessing der Daten für bessere Ergebnisse sorgt. In dieser Arbeit wird der Einfluss von Spezialisierung auf bestimme Taxonomische Level sowie der Einfluss von verschiedenen normalizierungs Möglichkeiten dieser gezeigt. Somit wird in dieser Arbeit wird Festgestellt, welche Preprocessing Methoden, sowie welche Klassifizierungs und Encoding Methoden sich am besten eignen um ein Klassifizierungs Model für zwei unterschiedliche Biome zu erstellen, und wie sich die Ergebnisse der Methoden von der Ähnlichkeit der zur Eingabe gegebenen Biome abhängt.

2 Ressources

2.1 MGnify

MGnify ist ein Teil vom European Bioinformatics Institute (EBI), erlaubt das Analysieren von Samples. Mgnify gibt nicht nur eine Möglichkeit zur Analyse von Samples, sondern fungiert auch als öffentliche Datenbank für

Samples und deren Analysen. In der Kombination mit dem Eurpean Nucleotide Archive (ENA) lassen sich zusätzliche Metadaten für Samples abfragen. Auf MGnify ist jedes Sample einem Biom zugeordnet und per API abfragbar. Darduch bietet sich MGnify sehr gut als Startpunkt der Daten dieser Arbeit an.

Aus den Aufbau von Mgnify folgt, das Samples von unterschiedlichen Studien Kommen sowie von unterschiedlichen Menschen zu Unterschiedlichen Zeiten entnommen wurden. In Kombination mit den Konstanten entwicklung der MGnify Pipeline und Unterschiedlich genauen Biomen kann es zu unterschieden in den Ausgangsdaten kommen.

Dennoch denken wir das MGnify eine gute Quelle für die Ausgangsdaten ist. Zum einen, da wir durch Preprocessing die möglichkeit besitzen, Unterschiede in den Samples zu reduzieren. Zudem erlaubt die einbindung von MGnify, das diese Arbeit auf alle Samples anwendbar ist, welche in Zukunft zur Datenbank hinzugefügt werden.

2.2 Galaxy

Galaxy Europe (Galaxy) ist eine Open-source Platform welche es Nutzern ermöglicht mit hilfe von Tool Workflows zu bauen und diese auf einem Server auszuführen. Einer der Vorteile von Galaxy ist es, dass Workflows und Tools von Leuten genutzt werden können, welche kein Fachwissen mit den jeweiligen Programmiersprachen der Tools haben. So ist unter anderem der Zugang zu Machile Learing Methoden für nicht Informatiker gegeben, was ein aktuelles Problem in Biologischen Bereichen darstellt [Quelle].

Da Galaxy auf Servern Läuft, ist es möglich Workflows zu bearbeiten, auszuführen und nachzuvollziehen, ohne selbst die benötigte Hardware zu besitzen. Vor allem im Bereich des Machine Learing, in welchem wir uns befinden, gibt es Höhere Anforderngen an CPU und GPU welche somit leicht erfüllt werden können.

3 Related Work

4 Methods

4.1 Gathering data from MGnify

4.2 Preprocessing

Preprocessing beschreibt das bearbeiten von Rohdaten mit der Intention, die Daten für die Verarbeitung mit Machine Learing Algorithmen vorzubereiten. Dabei kann der Einfluss von preprocessing unterschiedliche Auswirkungen haben. Wir wenden in dieser Arbeit verschiedene Preprocessing Methoden an, um festzustellen, welche Kombination aus verschiedenen preprocessing schritten die beste Klassifizierung zweier Microbial community compositions erlaubt. OTUs haben dabei, wie viele andere Analysen von Samples in der Biologie, sehr datailierte bestimmung, sodas es bei der Erstellung von Trainingsdaten oft daszu kommt, das es mehr unterschiedliche Features als Samples gibt. Zudem Stammen in userem Fall die daten von Mgnify. Dabei stammen die Samples von verschiedenen Studien und basieren auf 2 Unterschiedlichen versionen der MGnify Pipeline. Mithilfe von Preprocessing reduzieren wir die Anzahl von Features, indem wir die Samples auf bestimmte Taxonomische Ränge beschränken. Zudem Normalisieren wir die Samples, um Unterschiede in Herkunft sowie Erstellung zu reduzieren, sowie die Vergleichbarkeit der Samples zu erhöhen.

4.2.1 Taxa selection

Die Tabellen für Biom1 und Biom2 erreichen mit einer Anzahl von jeweils etwa 500 Samples 2500 bzw. 2000 Features. Daraus ergibt sich das Problem, das ein Großteil dieser Features keine Aussagekraft für die Klassifizierung tragen. Eine möglichkeit den Featurespace zu reduzieren ist das anwenden von Autoencodern. Zusätzlich zum Autoencoding lässt sich durch das einschränken der Taxonomischen Tiefe die Anzahl der betrachteten Features reduzieren.

++++++++++++ Dabei kann das fokussieren auf ein Taxonomisches Level nicht nur die Anzahl reduzieren, sondern auch die Qualität der Features erhöhen. Das ist darin begründet, das sich Unterschiede in Counts abhäning von der Relation der Biome auf unterschiedlichen Taxonomischen leveln ausprägen. Desto ähnlicher 2 Biome sind, desto ähnlicher sind die ersten Taxonomischen ränge. Das fokusieren von Taxonomischen rängen macht erst ab Phylum Sinn, da es bei Superkingom sowie Kingdom nur auf einen Einstel-

ligen Feature bereich reduziert wird, welcher bei den Meisten Biomen sogut wie Überinstimmt +++++++++

4.2.2 Table Normalization

Das normalizieren von Tabellen ermöglicht es, bessere ergebnisse bei Klassifizierung zu erhalten. Dabei gibt es mehrere aspekte, welche von der Noralizierung beinflusst werden. Zum einen ist das die verglichparkeit verschiedener Samples. Datenbanken wie Mgnify enthalten tausende von Samples. Diese werden oft über Jahre angesammelt und Stammen aus verschiedensten Studien/Experimenten. Somit kann es unterschiede in der Sammlung der Probe, der Art der analyse sowie der tiefe der Analyse geben. Einer der größten daraus entstehender Unterschiede zwischen verschiedenen Samles ist die anzahl der Totalen Reads. Normalizierungsmehtoden können genutzt werden um Samles zu standatiesieren und somit vergleichabrkeit zwischen allen Samples schaffen. Ein anderer anwendungszweck von Noramlisierungsmethoden ist das beeinflussen von Datenverhältnissen. Abhänging von den Beobachteten Biomen sind die Unterscheidungsfaktoren stark oder schwach ausgeprägt. (VLLT BESIPEL GESUND KRANK => FEINE UNTERSCHIEDE, ERDE WASSER => GRÖSSERE UNTERSCHIEDE)

Normlaization Relative Abundance

Relative abundance fungiert als Standart methode um andere einen vergleichswert für andere Noramlisierungsmethoden zu bieten. Relative abundance wird anstelle von den Raw Reads als vergleichswert genutzt, da somit jedem Taxa im Sample der prozentuale Anteil zugeordnet wird und somit alle Samples untereinander vergleichbar werden. Das verhindert den einfluss von Total reads auf das Ergebniss.

Normlaization Softmax

(WARUM SOFTMAX ?) Im Vergleich zu normaler Softmax Normalisierung wenden wird Softmax auf eine zuvor mit Relative abundance normalizierten Tabelle, welchen werte mit 100 Multipliziert wurden. Das vornormalizieren mit Relative abundance wird benötigt, da die Anzahl der Reads sehr von der Methode, mit welcher das Sample erstellt wurde abhängt. Somit gibt es oft sehr hohe Werte welche beim errechnen von Softmax für overflow Error sorgen. Das multiplizieren mit 100 wird genutzt, um den ausgangs Wertebereich vom [0,1] auf [0,100] zu Mappen und somit den Einfluss von Softmax darzustellen.

Normlaization CSS

CSS Normalisation basiert auf dem metagenomeSeq package von Bioconductor. Im unterschied zu Relative abundance und Softmax handelt es sich bei Cumulative Sum Scaling (CSS) um eine form der Library Normalization. Dazu werden alle samples nach Größe sortiert und Library weit wird eine quantile errechnet, danach wird diese Quantile genutzt um einen Normalisierungsfaktor für jedes der Samples zu berechnet. Somit hängt die Normalisierung des einzelnen Samples von der Verteilung der am wenigsten vorkommenden Taxa ab. CSS geht dabei davon aus, das diese verteilung über alle Samples hinweg ähnlich verteilt ist.

Normlaization RLE

RLE ist eine Form der Library Noramlization, welche auf der Annahme basiert, dass die ein Großteil der Assignments in den Samples keinen signifikanten Einfluss auf die Klassifizierung auswirkt. Dazu wird aus der Library ein Referenz Sample berchnet, welches auf dem Mean der Samples basiert. Der Median der Verhältnisse dieses Samples mit allen anderen Samples ergibt den Skalierungsfaktor, mit welchem am ende jedes Sample verrechnet wird.

Normlaization TMM

Genauso wie CSS und RLE handelt es sich bei Trimmed mean of M values (TMM) ebenfalls um eine Form der Library Noramlization. Dabei werden Noramlization Faktoren im verhältniss zu einem Referenz sample berechnet (unter Ausschluss der am stärksten vorhandenen Taxa). Und diese Noramlisierungs Faktoren werden dann mit den jeweiligen Samples verrechnet, um eine Normalisierte Matrix zu erhalten.

TMM sowie RLE stammen werden genutzt um Gene Libararys zu normalisieren. Dabei basieren beide auf der Annahme, dass es bei den meisten Genen keinen signifikanten unterschied zwischen zweier experimenteller Stadien gibt. Sie also nicht als differentially expressed (DE) gelten.

In dieser Arbeit werden beide Methoden benutzt, um Samples von Taxonomischen Zuordnungen zu Normalisieren, basieren auf der Annahme, das es viele Taxonomische Zuordningen gibt, welche insigifikant für die Klassifiezierung eines samples sind.

4.3 DeepMicro

Alle genutzten Encoding sowie Classification Methoden stammen aus DeepMicro(REFERENT PAPER DM). In welchem sie genutzt wurden, um Krankheiten bei Menschlichen Patienten vorherzusagen.

4.3.1 AutoEncoding

Autoencoders are used to encode an Input in a way, that an decoder can reconstruct the Input with as minimal loss as possible. In this process the encoded version of the input has normally a lower dimension compared to the input. In machine learing Auto encoders can be used to reduce the feature count, by using the encoded version of samples as input.

Autoencoders are a type of neural network, which can be trained on Input data to learn an effective encoding, usable for further usage. Beeing a neural network, there are many ways to influence such an encoding. DeepMicro comes with a shallow autoencoder(SAE), deep autoencoder(DAE), variational autoencoder(VAE) and convolutional autoencoder(CAE). All of them got optimized using gridsearch in the original DeepMicro paper[SOURCE].

AutoEncoding 1 SAE

A shallow auto encoder is constructed from an Input layer, latent layer and output layer. It learns a simple encoding, usable to reduce the featured of samples. Due to its small size, it is limited in encoding relations between features and mainly used for feature reduction.

AutoEncoding 2 DAE

A deep auto encoder is has, in addition to input, latent and output layers hidden layers in the encoding and decoding networks. These hidden layers allow the deep auto encoder to encode relations between features, resulting in the ability to learn a more complex encoding, compared to a shallow auto encoder.

AutoEncoding 3 VAE

Variational autoencoders (VAE) are a type of generative model, allowing generation of new samples based on a trained representation. Variational autoencoder work by learning a probabilistic representation of input samples through encoding of mean and variance, generating a destribution from which new samples can generated. The encoded mean and variance make up the latent space of the variational autoencoder, resulting in a lower dimansionality compared to the input samples.

AutoEncoding 4 CAE

Convolutional autoencoder differentiate themselves from other autoencoders by using convolutional layers instead of feed forward layers. In each layer a filter and local region of the layer before is used to calculate a part of

the new layer. Convolutional layers allow regional close features to influence each other and are often used in areas where proximity of features play an important role like image or sound generation.

Convolutional layers require a Matrix as input, instead of the feature vector. For that the vector is reshaped into a $d\ x\ d\ x\ 1$ matrix with $d = \lceil\sqrt{b}\rceil$, filling empty spaces with zero. For the decoding part of the autoencoder deconvolutional (convolutional transpose layers) are used.

In DeepMicro grid search was used to optimize the number of convolutional layers as well as the number of filters in the first convolutional layer. And the number of filters got halved each subsequent layer.

4.3.2 Classification

Classification 1 Support vector machine

Support vector machines are a Machine learing algorithm, which Takes N dimensional Input and tries to find a seperating hyperplane of dimension N-1 which seperates 2 Classes, allowing for classification in relation of an input to the hyperplane. Support vector machines are commonly used in classification biological Data[SOURCE]. Still, there are two problems with support vector machines. The first one is that SVM perform worse when there are a lot of features, wich do not have an infulence on classification. This problem should be reduced by Reducing the amount of Features through focosing on a specific taxonomic rank or the usage of Autoencoding methods. Secondly SVM are not able to learn complex relations between features.

In the Original Paper Gridsearch was used to tune penalty parameter C and the RBF kernel coefficent to optimize the method.

Classification 2 Random Forest

[Bootstraping]

Similar to Support vector machines, Random Forest(RF) is often used in Microbiom analysis [Source]. Random Forest is based on Decision Trees. A decision Tree is a mechine Learing algorithm where the Training data is used to Build a Decision Tree by splitting the Data on each node until there is a classification on each Leaf. Especially when using numbers as input decision trees tend to overfit. A Random Forest is using multiple Decision Trees to create a way of classifying Data. In a Random Forest each Tree is build by only using a Random choice of Input features resulting in a lot of different decision Trees. When classifying a new sample, the sample is classified by each Decision tree and the Final classification is based on the majority of trees.

In Deep micro grid search was used to tune number of trees, minimum number of samples in a leaf node as well as the Split criteria.

Classification 3 MLP

Multilayer Perceptrons(MLP) is a feedforward arifical neural network. A MLP is constructed from an Input layer, one or multiple hidden layers and an Output layer. At each layer Output from the layer bevore and weights are used in a activation function to calculate a new Output. The First layer(Input layer) takes fatures as input and the last layer (Output layer) outputs a classification. In case of DeepMicro only a few hidden layers where used. Increasing the number of hidden layers allows the artifical neural network to learn complex relationships between input features. In case of microbiom composition in form of Taxonimic assignments, a shallow neural network is enugh to classifiy different bioms. Increasing the number of hidden layers is likly to result in overfitting. Due to the limited relation between diferent features [Only thought maybe change or remove later]

In DeepMicro Gridseach was used to optimize Number of hidden layer, number of hidden units in the first layer, dropout rate and number of epochs.