Tampere University

Rasmus Ryyppö

RESIDUAL NEURAL NETWORK IN THE IDENTIFICATION OF RARE DISEASES

Faculty of Information Technology and Communication Sciences
M. Sc. Thesis
November 2021

ABSTRACT

Rasmus Ryyppö: Residual neural network in the identification of rare diseases
M.Sc. Thesis
Tampere University
Master's Degree Programme in Computing Science, Data Science
November 2021

Rare diseases are difficult to identify and it may require years and multitude of different doctors to reach the correct diagnosis. Machine learning could be the answer to get correct diagnosis faster and that way improve patients' quality of life.

In this thesis, the objective was to implement residual neural network model that could assist in the early identification of the vasculitides, myositides and glomerulonephritides. Average prevalence in 100000 people for vasculitides is 5 persons, for myositides it is 10.5 persons, and for glomerulonephritides it is 1.6 persons. If we would pick randomly people for testing, estimated probability for vasculitides would be 0.005%, for myositides it would be 0.0105% and for glomerulonephritides it would be 0.0016%.

We found that for this purpose and with the data that Helsinki University Hospital provided, the most suitable model was the Inception type model. The constructed Inception type model performed robustly and over the expectations. The best results obtained showed that it could identify individual rare diseases from each other and from healthy controls almost six years before the initial diagnosis. Positive predictive value in this model for vasculitides was 0.2% and myositides and glomerulonephritides had 0.3%. Situation where patient has any of these diseases, positive predictive value was 6%. Classification accuracy of vasculitides was 42%, myositides was 45%, glomerulonephritides was 65%, and the control was 80%.

The Inception type model was compared against state-of-the-art models such as ResNet18, InceptionTime and XGBoost. Resnet18 and InceptionTime models were not suitable for this situation. This could be due the combination of sparse data and global pooling method. XGBoost on the other hand produced excellent results. The best results for early identification were received with one month before the initial diagnosis. Positive predictive value for vasculitides was 0.6%, myositides was 1% and in glomerulonephritides 0.5%. In the situation where patient has any of these diseases 8.8% was reached. Classification accuracy of vasculitides was 35%, myositides was 30%, glomerulonephritides was 83% and controls was 89%.

In the end, the early identification of rare diseases could be reached considerably earlier with either of these models. Since all these diseases are potentially rapidly fatal, and since once complications start to appear these rapidly lead into expensive supportive measures like intensive care, home care and dialysis, the potential individual and societal effect of reaching earlier diagnosis could be mutually beneficial to all, patients, society and healthcare.

Key words and terms: machine learning, rare disease, residual neural network, diseases detection, vasculitides, myositides, glomerulonephritides, XGBoost, Inception model

TIIVISTELMÄ

Rasmus Ryyppö: Residuaalineuroverkko harvinaissairauksien tunnistamisessa
Pro Gradu -tutkielma
Tampereen Yliopisto
Master's Degree Programme in Computing Science, Data Science
Marraskuu 2021

Harvinaiset sairaudet ovat vaikeita tunnistaa, oikean diagnoosin tunnistaminen voi kestää useita vuosia ja vaatia monia eri lääkäreitä. Koneoppiminen voisi olla ratkaisu nopeampaan diagnosointiin ja siten myös potilaan elämänlaadun parantamiseen.

Tämän pro gradu -tutkielman tavoite oli implementoida residuaalineuroverkko -malli, joka voisi avustaa vaskuliittien, myosiittien ja glomerulonefriittien varhaisessa tunnistamisessa. Keskimäärin esiintyvyys 100000 ihmisen joukossa vaskuliiteilla on 5.0 henkilö, myosiiteilla se on 10.5 henkilöä ja glomerulonefriiteillä se on 1.6 henkilö. Jos valitsisimme satunnaisesti henkilöitä testeihin, arvioitu todennäköisyys vaskuliiteille olisi 0.005%, myosiiteille 0.0105% ja glomerulonefriiteille se oli 0.0016%.

Havaittiin, että tähän tarkoitukseen Helsingin yliopistollisen sairaalan tarjoaman datan kanssa soveltuvin malli oli Inception -tyyppinen malli. Rakennettu Inception -tyyppinen malli suoriutui todella hyvin ja yli odotusten. Parhaimmat saavutetut tulokset näyttävät, että malli onnistui tunnistamaan harvinaissairaudet toisistaan ja terveistä henkilöistä noin kuusi vuotta ennen alkuperäistä diagnoosia. Positiivinen ennustearvo tällä mallilla oli vaskuliiteille 0.2%, ja myosiitteille ja glomerulonefriiteille saatiin 0.3%. Tilanteessa, jossa tarkasteltiin, että olisiko potilaalla jokin näiden sairaustyhmien sairaus, positiivinen ennustearvo oli 6%. Luokittelutarkkuus vaskuliiteilla oli 42%, myosiiteilla 45%, glomerulonefriiteillä 65% ja kontrolleilla 80%.

Inception -tyyppistä mallia vertailtiin yleisesti parhaimmiksi katsottuihin malleihin kuten ResNet18, InceptionTime ja XGBoost. ResNet18 ja InceptionTime mallit eivät soveltuneet tähän tarkoitukseen, mikä saattoi johtua harvan datan ja globaalin yhdistämismenetelmän yhdistelmästä. XGBoost toisaalta tuotti huomattavan hyvät tulokset. Parhaat tulokset aikaiseen tunnistamiseen saatiin kuukausi ennen toteutunutta diagnoosia. Positiivinen ennustearvo vaskuliiteille oli 0.6%, myosiiteille 1% ja glomerulonefriiteille 0.5%. Tilanteessa, jossa tarkasteltiin, oliko potilaalla jokin sairaus näistä sairausryhmistä, saavutettiin 8.8%. Luokittelutarkkuus vaskuliiteilla oli 35%, myosiiteilla 30%, glomerulonefriiteilla 83% ja kontrolleilla 89%.

Lopulta voitiin todeta, että harvinaissairauksien tunnistaminen onnistui paljon aikaisemmin näillä molemmilla malleilla. Näiden sairauksien ollessa potentiaalisesti nopeasti kohtalokkaita ja komplikaatioiden esiintyessä johtavat nopeasti kalliisiin tukitoimenpiteisiin, kuten teho-, koti- ja dialyysihoitoon, varhaisen diagnoosin potentiaaliset yksilölliset ja yhteiskunnalliset vaikutukset voivat olla molemminpuolisesti hyödyllisiä kaikille, potilaille, yhteiskunnalle ja terveydenhuollolle.

Avainsanat ja termit: koneoppiminen, harvinaissairaus, residuaalineuroverkko, sairauden tunnistaminen, vaskuliitti, myosiitti, glomerulonefriitti, XGBoost, Inception malli

PREFACE

This thesis was done while working at TietoEVRY Ltd in a Business Finland-funded research project eCare for Me coordinated by Helsinki University Hospital and its collaboration ecosystem CleverHealth Network. I would like to thank these organizations for enabling this work.

Adjunct Professor Mikko Seppänen has been the Principal Investigator and lead of the eCare for Me work package “Early Identification of Rare Diseases”. I thank him for his support in making my work possible as well as his forward-looking attitude towards the use of AI in healthcare.

Last but not the least I want to thank my thesis supervisors Professor Martti Juhola from Tampere University and PhD Miika Ermes from TietoEVRY Ltd, and the host of the seminar Professor Jaakko Peltonen from the Tampere University who gave me invaluable feedback and support during the time I worked on the thesis.

Helsinki 21.10.2021

Rasmus Ryyppö

Contents

1. Introduction

1.1. Rare diseases

There are 6000 to 8000 rare diseases in Europe and it is estimated that 30 million people are affected by those. According to the European Commission, a *rare disease* is a disease that occurs more seldom than one in two thousand (1/2000). [European Commission, 2021, Nguengang Wakap *et al.* 2020] In almost 85% of these rare diseases, prevalences are less than one person in one million (1/1000000) people [Nguengang Wakap *et al.* 2020].

In a survey conducted by Molster *et al.* [2016] in which about half of the undiagnosed patients who responded to the survey had to wait one year or more to get the right diagnosis. Also, about 65% of the respondents had to see three or more doctors, and in addition to that, 50% had received one or more incorrect diagnoses. [Molster *et al.* 2016]

This shows that rare diseases are hard to diagnose because there are so few patients. In addition, it is quite probable that a patient with an undiagnosed rare disease will get at least one incorrect diagnosis, and getting the right diagnosis might take a long time. The process might include many visits to the healthcare centers and many different doctors. [Jia *et al.* 2018].

1.2. Machine learning in healthcare

Use of *machine learning* (ML) is growing in healthcare sector, but it is still not commonly used. [Qayyum *et al.* 2021] The idea of ML and *artificial intelligence* (AI) is not to replace healthcare workers, but to be a tool that the healthcare workers can use to improve treatment and give new insight of the patient's condition [Krittanawong *et al.* 2017, Panesar 2020].

At the current state of healthcare, there is a lot of room for human error. Diagnosis can be dependent on the doctor's and patient's background. Also, the patient's description of symptoms can be unintentionally misleading by accident. ML can help here to diagnose patients, identify patients that are vulnerable to frequent illnesses, and direct patients to more suitable treatments. [Bhardwaj *et al.* 2017] This would be beneficial for the patients, healthcare and society. Patients could get the correct treatment faster and then prevent huge costs of different treatments and doctor visits [Bhardwaj *et al.* 2017]. The healthcare would not waste resources and the society would be able to divert such funds saved to more effective use of resources, without failure demand when people with illnesses would recover faster with better targeted therapies and treatment plans.

1.3. Machine learning in detection of rare diseases

Studies show that the most common aim of ML in the healthcare sector is in improving diagnosis and prognosis. In addition, most of these studies have revolved around topics of rare developmental defects during embryogenesis or rare systemic-, rheumatologic- or neurologic disease with available targeted treatments. [Schaefer *et al.* 2020]

In recent studies the most common detection methods have been used are *Support-Vector Machine* (SVM), *Random Forest* (RF), *Neural Network* (NN), *Deep Neural Network* (DNN), *K-Nearest Neighbor* (K-NN) classifier and *Naïve Bayes* classifier. There are several *Diagnosis Decision Support Systems* (DDSS), but only handful of those use AI or ML functionalities or methodologies. Even fewer of those DDSS are focused on rare diseases [Brasil *et al.* 2019]. For example, Jia *et al.* [2018] succeeded in developing *Rare Disease Auxiliar Diagnosis system* (RDAD) which is based on phenotypes. This DDSS reached up to 98% diagnostic precision [Jia *et al.* 2018]. The types of data used in previous research varies greatly, including for example images, electronic healthcare records, laboratory data, medication, etc. [Schaefer *et al.* 2020]

1.4. Focus of the research

In this research. I approach the identification of rare diseases by using *residual neural network* (ResNet) which has lately shown promising results in time series and medical image classification problems. The rare diseases which were selected for study in this thesis are vasculitides, myositides and glomerulonephritides. The aim is in the early diagnosis or identification of persons with high probability of carrying an undiagnosed rare disease, and to also compare the results obtained with the state-of-the-art residual neural network model to those obtained commonly used random forest-based classifier.

Myositides, or in other words, idiopathic inflammatory myopathies (IIM), is a rare disease group of rare autoimmune diseases. Common symptoms are, for example, raised skeletal muscle enzymes, muscle weakness, etc. Subclasses of myositis are polymyositis (PM), inclusion body myositis (IBM) and debilitating movement disabilities (DM). [Chinoy and Cooper 2018] Based on the rare disease database Orphanet [2021a, 2021b, 2021c] myositis prevalence varies between different subclasses, but prevalence rates of polymyositis and dermatomyositis range between 1-9 in 100000 people, while inclusion body myositis is much rarer with prevalence of 1-9 in 1000000 people. Polymyositis and dermatomyositis have occurrence peaks in childhood and with people aged between 30-50 years. Inclusion body myositis is mostly diagnosed in patients over 50 years of age. [Chinoy and Cooper 2018]

Vasculitides are the inflammation of blood vessel walls. It varies from mild blood vessel inflammation to organ- or even life-threatening rare disease. It has subgroups of large vessel vasculitis (LVV), medium vessel vasculitis (MVV), and small vessel

vasculitis (SVV), and whether mostly arteries, veins or both become inflamed. [Davey and Sprigings 2018] the combined prevalence of vasculitides range between 1-9 in 100000 people [Orphanet, 2021d]. Different vasculitis diseases are common in different ages. Kawasaki disease peaks for children averagely in the age of five, and it is more common in Asia, for example in Japan and Korea prevalence is 218.6 and 113.1 in 100000 people, compared to Europe, where for example in the United Kingdom or Finland it is 8.4 and 7.2 in 100000, respectively. Giant cell arteritis is more common in older people in the age of 45-80. [Ball *et al.* 2014]

Glomerulonephritides are rare inflammatory diseases affecting renal glomeruli, which participate in filtering foreign substances and metabolic end products in the kidney. If the inflammation in the kidneys does not affect glomeruli, one may also speak of nephritis. Often and here also the term glomerulonephritis is used for both glomerulonephritides and nephritides combined. Usual symptoms for acute (glomerulo)nephritis include, for example, fluid retention and hypertension, and there is possibility for non-specific systemic symptoms such as fever and weight loss. [Davey and Sprigings 2018] Glomerulonephritides has prevalence of 1.6 in 100000 people. [Woo *et al.* 2010] Diseases under the glomerulonephritides class can occur in different ages, and some might be more common with a specific gender. Lupus nephritis usually affects younger people in the age of 10-40 years while Anti-GBM disease (Goodpasture's disease) can occur with any age. Anti-GBM is more common in younger males, and while for females its incidence increases with advancing age. [Davey and Sprigings 2018]

2. Methods

In this chapter I will introduce some of the important concepts and methods used in this thesis. I am inspecting residual neural networks with different architectures, and their performance in detecting rare diseases. Very important part of the residual neural network is the knowledge of convolution. This method is used in the neural network layers, and it will be introduced to give a necessary understanding of the neural network used in this thesis.

2.1. Patient data

The rare disease project which my thesis is about, is part of bigger CleverHealth Network's eCare for me project entity. Three major questions stand up in this project. Firstly, how can we identify diseases at an early stage to improve patient care and avoid costly processes of delayed and/or misdirected treatments and wrong medication. Secondly, how can we become better at automating diagnostics and treatment selection based on detailed and comprehensive disease characterization. Thirdly, how can we enable a more comprehensive outpatient and home care using novel innovations in logistics, biomonitoring and patient-reported and -measured outcomes. Helsinki University Hospital (HUS) has unique position in the crossroad of research and clinical care to address these three questions by their technological advances. Provided data was pseudonymized, so individual patients could not be identified and in addition, the data was handled in a secure environment.

The research was approved by the ethical review board of HUS and for this project vasculitis, myositis and glomerulonephritis were chosen, because they were the commonest rare disease groups in HUS, with highly significantly delayed diagnoses and associated failure demand. Also, once they have become severe, these rapidly cause the need for expensive supportive therapies. This ensures sufficient data and has a real-world value. We aim to reach the diagnosis earlier. This would result in less suffering and earlier targeted therapy, with associated cost-savings.

RARE DISEASE	N	PREVALENCE
VASCULITIDES	2919	0.00005
MYOSITIDES	942	0.000105
GLOMERULONEPHRITIDES	11036	0.000016
CONTROLS	100000	

Table 1. Number of patients in the rare disease and control groups.

Table 1 shows the number of patients in the individual rare disease group and control group. The patient data includes mostly numerical bioinformation data and numerical knowledge of situations if bioinformation value is outside of the reference value range. There are 3930 combined numbers of features, these include 1965 bioinformation values and 1965 attributes of out of range values. The most common feature across the groups is the knowledge of the *hemoglobin* (B-Hb) if it is out of reference value range. Averagely this test is obtained from the control patient seven times and from patients with rare diseases 45-71 times. Next there is knowledge if the test results of *leukocyte* (B-Leuk), *red blood cell count* (B-Eryt) and *hematocrit* (B-Hkr) are out of reference range. The control patient has all of those tests taken averagely seven times. A patient with vasculitis has received about 51 results of B-Eryt and B-Hkr, and B-Leuk for 50 times. Single patient with Myositis has about 46 results of B-Leuk and 45 results of B-Hkr and B-Eryt. Glomerulonephritis patient has average of B-Eryt and B-Hkr tests 70 times and B-Leuk 62 times.

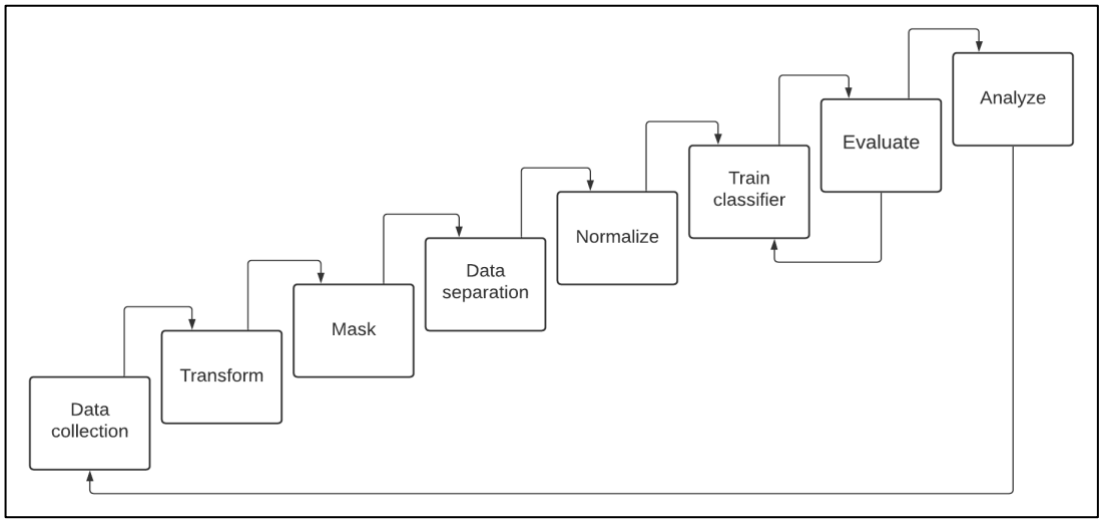

Figure 1. Data processing steps.

In the test setup there are different processes for data to go through before it can be used for training the model. Some of the steps are commonly used in knowledge discovery and data mining, such as *transformation*, *preprocessing* and *normalization* [Han *et al.* 2011]. There are also steps that are used specifically in the ML, such as separation of dataset, training classifier, and so on. [Kumar 2019]. All the test setup steps can be seen in Figure 1.

When data is collected, it is not in a perfect format, and it may come from multiple different sources, so it needs to be preprocessed to make it compatible with the neural network. Preprocessing is done by following the principles of tidy data. This principle says that every column is a *variable*, every row is an *observation* and every *cell* is a single

value. [Wickham 2014] Data transformation of data includes pivoting of the data. In this case, raw data format is long, which is not compatible, and needs to be changed to wide format. Firstly, data dates are transformed to respond the timeline of the patient, so every observation is a day in a patient's life. Secondly, variables are collected, which are, for example, different bioinformation tests. Thirdly, values are added to the cells of the corresponding variables and observations.

After that, the data is masked as requested, so the diagnosis and everything after that is removed, as well as the requested number of days before the diagnosis. Also, the data is pseudonymized, so identification of patients is not possible. For the last steps before training the classifier, the data is divided to training set, validation set, and test set, and normalized. This is because the result would not be valid if only one dataset would be used, and it would not simulate the real situation as performance would be tested with the same data as the model is trained already. Normalization is done, because feature ranges are very different and to increase performance.

Training of the classifier takes multiple epochs, and evaluation of the model is done after each epoch. Training is physically done with two GPUs of NVIDIA Tesla V100s which will make it a lot faster. Still, it takes hours or even days to train a proper model. If it is recognized early that a model is not learning, we can stop the training – and the other way around, if we notice that we are underfitting, we can continue training. After training a model and evaluating with the validation dataset that it is a good model, analysis of the results is done. It is done by using the test set and multiple metrics.

2.2. Technologies

TensorFlow is a deep learning framework created by Google. Google released it in 2015 for the public and uses it by itself in their projects, such as Google Assistant and Google Search. TensorFlow offers parallelism for models and data. Also, it allows to run models in the CPU or GPU. [Manaswi 2018] It is beneficial to run models on GPU as it has been proven that it can be four to five times faster than running in the CPU. [Buber and Diri 2018]

Keras is a high-level API implemented to the TensorFlow. It allows also to run models in the CPU or GPU. Keras support various model architectures for neural networks, such as recurrent- and convolution networks. [Baranwal *et al.* 2019] Keras has everything we need, because Keras provides input layers, convolution layers, output layers, activation functions, a learning procedure, and so on. [Moolayil 2018] In addition to that, Keras is easy to use because of the modularity it provides, and because it has comprehensive documentation. [Baranwal *et al.* 2019]

Python is a programming language which supports object-oriented-, functional-, imperative- and procedure programming paradigms, and it is easy to work with as its

syntax is pseudocode-like. [Marvin *et al.* 2018] In addition to that, it provides lots of different data science third-party libraries and has various support in specialized software. These are the reasons that keep python one of the most used programming languages. [Cielen *et al.* 2016]

2.3. Convolution theory

Before understanding the convolution, we need to know what kernels are. Kernels are sometimes also called filters. *Kernels* can be identified as matrices that usually contain small integers, as seen in Figure 2. [Michelucci 2019]

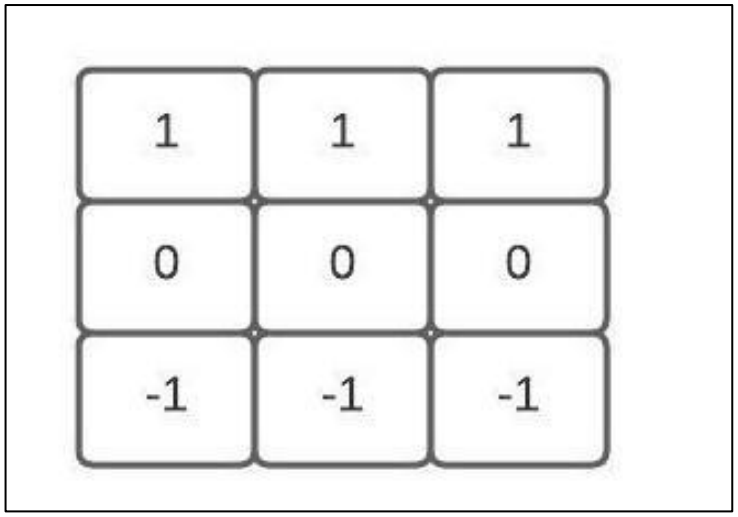

Figure 2. Example kernel matrix B.

In *convolution* kernels are moving over the inspected matrix and creating new matrix. New matrix is constructed through modified matrix multiplication rule which will follow the formula:

$$C = (A * B)_{i,j} = \sum_{m=0}^{n_k-1} \sum_{f=0}^{n_k-1} A_{i+m,j+f} B_{i+m,j+f}.$$

The main matrix is in the formula A, and the B is the kernel matrix. [Michelucci 2019]

In Figure 3 we can see an example convolution which has 4×4 matrix A and 3×3 kernel matrix. Stride in the example is one. *Stride* describes how many columns and rows kernel moves when it slides over the input. With this we will have 2×2 result matrix C. If stride would be two the size of the result matrix C would be 1×1, because kernel matrix B would not fit the matrix A more than once horizontally. In Figure 4, we can see that result matrix C is smaller than in Figure 3, because the stride is two in that example, and the kernel does not fit there twice. In this case there were no padding added and it

would not inspect the last column at all, and therefore the data in the last column would be missed. *Padding* means that zeros are added around the input. [Michelucci 2019] TensorFlow provides padding types of same and valid for convolution layers. *Valid* padding is presented in Figure 4, so basically it means that zero padding is not added around the input, and some columns may be ignored. *Same* padding adds zero equally around the input when needed. [Géron 2017]

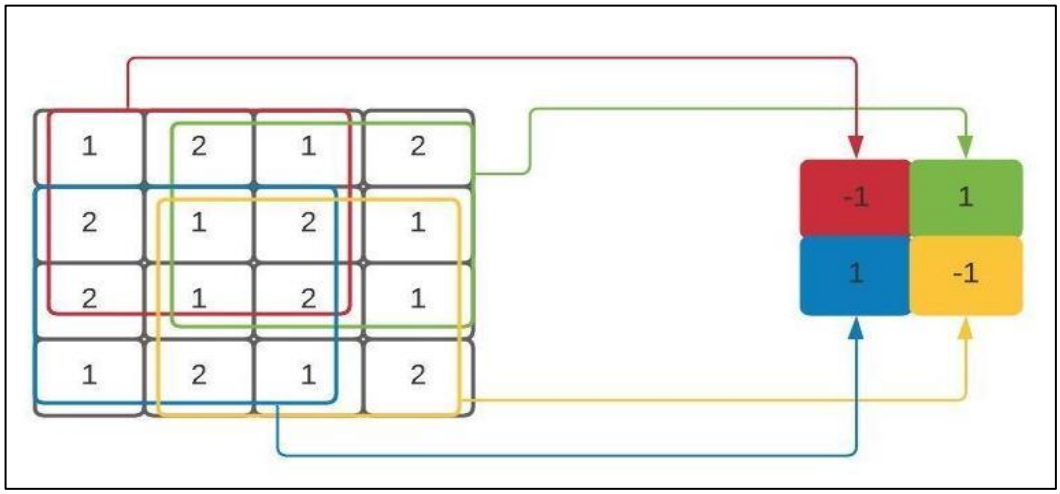

Figure 3. Example convolution over matrix A with example kernel matrix B and result matrix C.

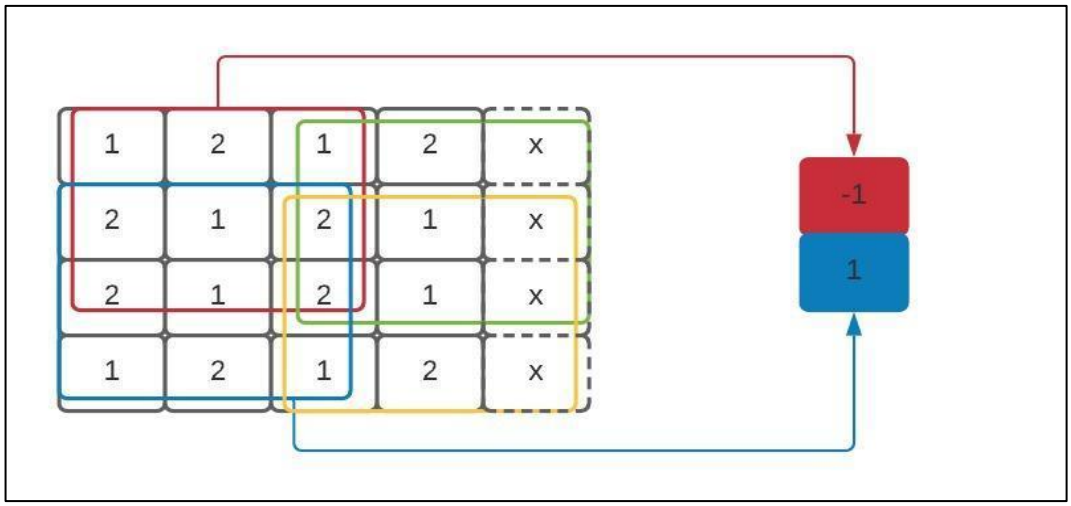

Figure 4. Example convolution that would need padding, smaller kernel or smaller stride so the whole data would be inspected.

2.4. Pooling

Pooling is an important operation when using convolution and neural networks. There are few different pooling methods, such as max- and average pooling, which are the most

commonly used. [Michelucci 2019] In addition to that, there are also global functions available for both. [Lin *et al.* 2013, Michelucci 2019]

When convolution theory is understood, this concept is easier to comprehend as the same principle applies here, but the pool is just a scope that is inspecting the matrix. In the max pooling, inspection is done in size n_k area. The outcome is the max value in the inspected area, and an example can be seen in Figure 5. Same thing is done in the average pooling, but the average value is calculated as in the example Figure 6. [Michelucci 2019] Global pools in the other hand are returning only one value for each feature map. This is at the same time an advantage, but also a risk, as a whole input is squeezed to one value and great knowledge of situation of the convolution is needed. [Lin *et al.* 2013, Michelucci 2019]

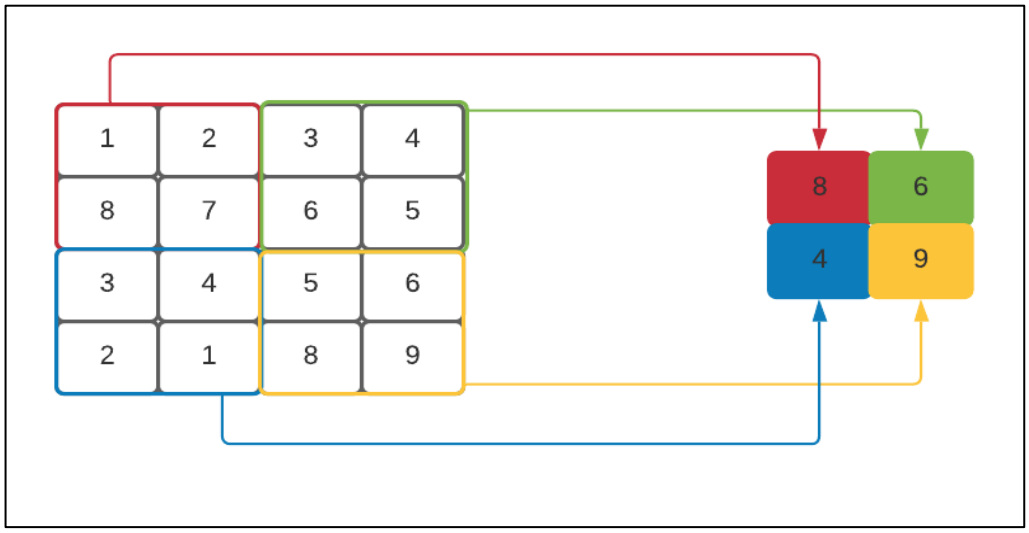

Figure 5. Example of max pooling with stride value 2.

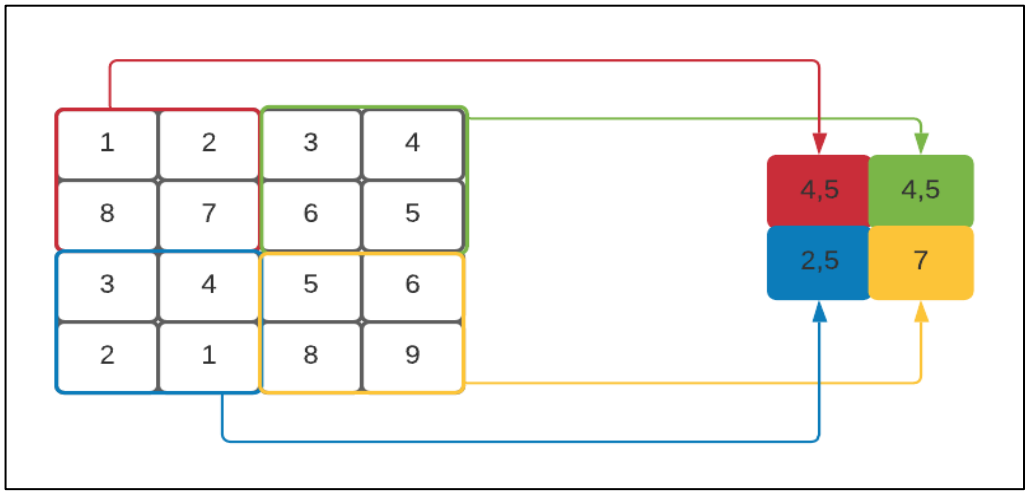

Figure 6. Example of average pooling with stride value 2.

Global average pooling was introduced by Lin *et al.* [2013], because fully connected layers are vulnerable for overfitting, and this strategy was meant to supplant fully connected layers. The advantage global average pooling method does not have optimizable parameters and it is enforcing correspondences among categories and feature maps. Lin *et al.* [2013] Used global average pooling as part of their deep network model called "Network In Network" (NIN). This model was classifying images content and were tested against SVHN, MNIST, CIFAR-10 and CIFAR-100 datasets. This model was outperforming the other model in CIFAR-10 and CIFAR-100 datasets at that time, but it did not perform as well with the SVHN and MNIST dataset where it was supposed to recognize numbers. The comparison of the global average pooling impact was tested, and it did perform better than fully connected layers or fully connected layers with dropout layer. [Lin *et al.* 2013] The global average pooling became a very commonly used method, especially after He *et al*. [He *et al.* 2016] introduced their residual neural network as they were using the global average pooling in their model's last layer.

For our data, max pooling was the best option, as the average values were all getting too close to zero, so identification was not successful. Also, global versions were not suitable as they did group up the values to one value too much and missed lots of important values. This is quite unorthodox for residual neural networks as global average pooling is getting more attention at the current time. Our data is max pooled twice. At the start we are reducing the timeseries to groups and at the end getting max values of each feature.

2.5. Masking

Masking the data can be done with different techniques and for different reasons. Masking is also sometimes referred to as obfuscation, pseudonymization or tokenization. The idea of it is to hide data, and it is widely used to hide sensitive data, such as *social security numbers* (SSN) and first and last names. Masking can be, for example, pseudonymization or randomization. They differ from each other in a way that in pseudonymization it is possible to recover the masked data by decrypting the encryption, which is possible if the encryption key or salt is known. In randomization this is not possible. [Emam and Arbuckle 2013] In addition to that, masking can be done by nulling, substitution and shuffling. These are a one way to get more realistic data for testing and training. When using nulling, data is removed completely. In substitution, data is replaced by unrelated data, which is similar to the original data, such as all SSNs would be changed to "xxxxxxxxxxx", so that every person would have the same SSN. In addition, the values in the data Shuffling are derived from the original value, so it is a bit similar with the substitution technique. [Anjaiah Gujjary and Saxena 2011]

If we think about masking and the need for it from the aspect of this research, masking is needed to pseudonymize part of the data, such as SSNs and names, so that identification of patients is not possible. In addition to that, masking is needed for getting realistic data which resembles real situations. It would not be predicting, if the data contains information of diagnosis which is being predicted. In addition, mask size can increase, which is important in the early diagnosis, because the aim is to find indications of rare diseases earlier than it is at the moment. When the masking size increases, the training of the neural network is gets more difficult.

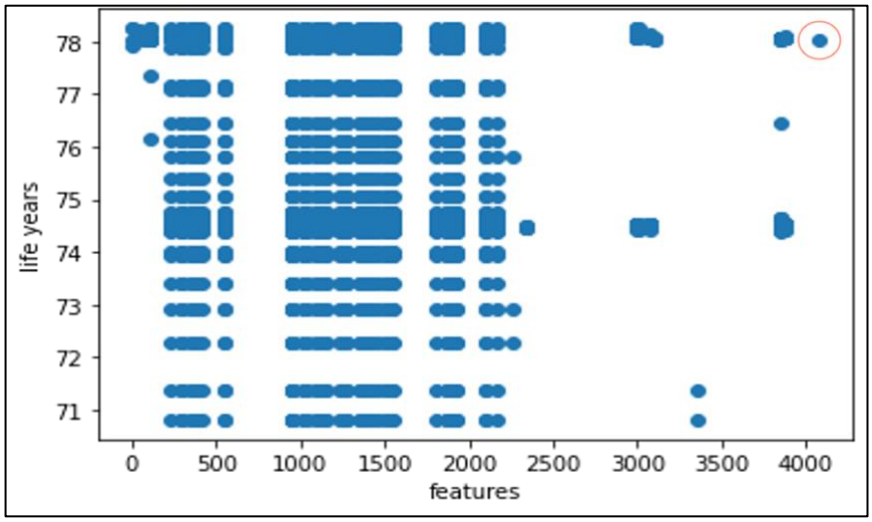

Figure 7. Hypothetical patient with rare disease identified and marked with red circle. Vertical axis has patient's age and horizontal axis includes feature numbers.

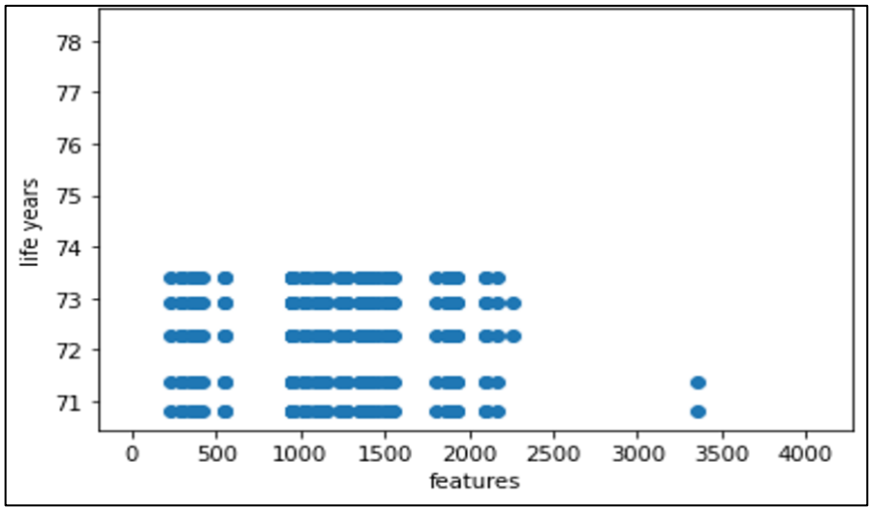

Figure 8. Hypothetical patient with masked data by using nulling technique. Vertical axis has patient's age and horizontal axis includes feature numbers.

From Figures 7 and 8 we can see what tests had been made to the hypothetical patient and what was the age of the patient at that time. The features are mostly bioinformation, but it includes diagnoses and other features as well. As seen in Figure 7, the hypothetical patient has indication of a rare disease and a clear pattern is possible to see, but in practice this example data cannot be used, because the indication of the rare disease makes the data in Figure 7 unrealistic. Therefore, masking is needed to simulate a realistic situation, as demonstrated in Figure 8. Nulling technique is used in Figure 8, and it removes everything after the initial diagnosis is given and four years before that.

2.6. Activation function

There are several different activation functions that can be used in the ML models, and it is very important to select the correct one to prevent issues such as vanishing gradient. Most commonly used are *sigmoid*, *identity*, *hyperbolic tangent* (tanh) and *rectified linear unit* (ReLU). [Di *et al.* 2018] Also there are *noisy*-, *leaky*-, *parametric* ReLU and *exponential linear unit* which are variations of ReLU and should be mentioned [Ahmad 2020, Di *et al.* 2018, Vasilev *et al.* 2019]. Softmax activation function use case is differs from the others, as it is used in the output layer to provide probability distribution for multiple classes [Ahmad 2020]. A few activation functions and their formulas which were considered can be seen below:

Identity:

$$f(x) = x$$

Sigmoid:

$$f(x) = \frac{1}{1 + e^{-x}}$$

tanh:

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

ReLU:

$$f(x) = \max(0, x)$$

Leaky ReLU:

$$f(x) = \begin{cases} x, x \geq 0 \\ 0.01 * x, x < 0 \end{cases}$$

Parametric ReLU:

$$f(x) = \begin{cases} x, x \geq 0 \\ \alpha x, x < 0 \end{cases}$$

Softmax:

$$P(x_s) = \frac{e^{x_s}}{\sum_{i=1}^{n} e^{x_i}}$$

Tanh and sigmoid activation functions are both affected by a vanishing gradient problem. [Di *et al.* 2018, Kumar 2019] Vanishing gradient means the gradient of the loss is getting smaller and close to zero, which makes the training harder. Basically, when the vanishing gradient problem affects the backpropagation of the network, it slows the learning of the initial layers which should learn basic features and the later layers starts to learn faster, and these should learn more complex features. [Kumar 2019] This is not a problem in the identity- and ReLU activation functions. Despite not having the vanishing gradient problem, the ReLU has disadvantages, such as dead neuron problem. This means that the neuron is unusable for future data points, and it will not get activated, because weights are updated in such way by a large gradient. For this reason, leaky- and parametric ReLU were introduced, which will prevent dead neuron problem, but are still having a problem with inconsistency. [Di *et al.* 2018] These two activation function formulas have quite similar formulas, but they work very differently. The parametric ReLU lets neural network choose the slope α value [Ahmad 2020]. Softmax is an output layers activation function, and it is used mostly in situations where there are more than two classes in classification. It provides probabilities for each classification class. [Ahmad 2020]

I choose to use tanh functions as it fitted the best for our data. In addition, it is easier to model positive, negative and neutral values and because residual connections help with the issue of vanishing gradient problem. Identity activation function could have been chosen, but it lacks back-propagation because of its derivative function which is constant, and due to this, relation to the input is lost. For this reason, it was not selected. Sigmoid activation function's issue was that it was not zero centered, and our data would have lots of neutral zero values. ReLU and its variants were also in the scope at the start as it is at the moment the most used activation function in residual neural networks, but ReLU does not comply with the negative values in our case. Leaky rectified unit on the other hand would have fixed the issue with negative values, but it did not process highly positive and highly negative values equally, as they can in our case have equal meaning to the outcome. Parametric ReLU would have weighted the values itself, but it provided too much parameter for our cluster and crashed due to insufficient memory. Softmax

activation function for our output layer was a natural choice as it provides probabilities for each class, and we have four classes.

2.7. Optimizer

Optimizers are crucial part of the neural network learning as it is the algorithm that updates neuron's weights in the training process. There are, for example, Adaptive moment estimation (Adam)-, Stochastic gradient descent- (SGD) and adaptive gradient (Adagrad) optimizer. Adam might be one of the most used at the moment, as it has the best performance in deep neural networks. Stochastic gradient descent optimizer is mostly used in shallow neural networks, so it is not in our scope, nor is the Adagrad, although it is an adaptive optimizer. [Loy 2019] Both Adam and Adagrad keep the history about gradients, but Adam also saves exponentially moving averages of the gradients and the squared gradients. [Dawani 2020]

2.8. Residual neural network

Residual neural network was introduced by He *et al.* [2016] and it is a deep neural network. That model was named as ResNet. The difference to ordinary neural network is that it has extra connections, which makes it possible to skip layers. These connections are called skip-, shortcut- or residual connections [Chaweewat and Singh 2020, He *et al.* 2016, Ismail Fawaz *et al.* 2020]. It is said that ResNet can reach depth of over 100 layers [Wang *et al.* 2021]. Although He *et al.* [2016] noted a decrease of accuracy with 1202-layers of the network compared to their 110-layer ResNet, this is argued to result from overfitting. Figure 9 illustrates residual neural network block and residual connection to next block.

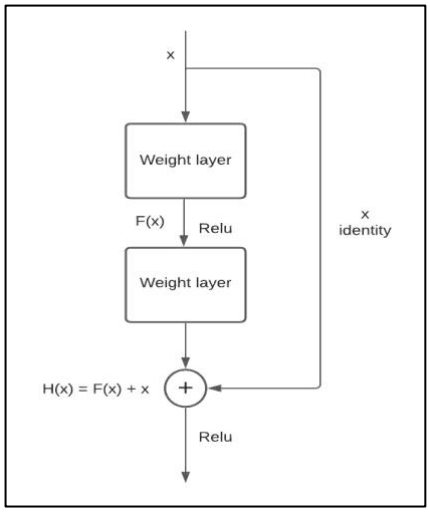

Figure 9. Example ResNet block and residual connection. [Erwandi and Suyanto 2020, He *et al.* 2016, Wang *et al.* 2021]

The technique of residual neural networks has been most commonly implemented in tasks which are based on image classification. For example, Erwandi and Suyanto [2020] researched breast cancer classification with biomedical images, such as mammograms and histopathology, and Jung *et al.* [2017] researched vehicle classification through a traffic surveillance system. In addition to those, InceptionTime deep learning model, which is a residual neural network model, has been used in time series classification [2019, 2020].

2.9. Metrics

For the metrics, *true positive* (TP), *true negative* (TN), *false positive* (FP) and *false negative* (FN) are used, and by calculating ratios of those, it is possible to calculate *receiver operating characteristics* (ROC) with *area under the curve* (AUC), *positive predictive value* (PPV) and *negative predictive value* (NPV). In the training phase, *categorical cross entropy* (CCE) and *accuracy* (ACC) were the best metrics available.

	NULL HYPOTHESIS IS FALSE	**NULL HYPOTHESIS IS TRUE**
REJECTED NULL HYPOTHESIS	True positive (TP)	False positive (FP)
FAILED TO REJECT NULL HYPOTHESIS	False negative (FN)	True negative (TN)

Table 2. Confusion matrix of hypothesis testing results. [Salkind 2010]

If the hypothesis is that a patient has a rare disease, then the null hypothesis is that patient does not have a rare disease. As Table 2 shows, when null hypothesis is rejected and it is actually false, TP is received. For example, a patient with rare disease is classified with a rare disease. On the other hand, if the null hypothesis is rejected, but null hypothesis is applying, result is FP. For example, a patient is classified with a rare disease and in reality, the patient does not have a rare disease. In addition, when the null hypothesis is failed to reject and null hypothesis is indeed true, TN is received. For example, a patient is not classified with a rare disease and in reality, the patient does not have it either. When the null hypothesis is failed to reject and it is really false, result is FN. For example, a patient with a rare disease is classified without rare disease. [Salkind 2010]

Ratios of those TP, TN, FP and FN are telling how many of the results are in a specific category. TPR indicates how much there are positive results caught from the overall positive cases. TNR tells how many negative results are given correctly from all of the

negative cases. FPR describes how many falsely diagnosed positives are given within the negative cases. FNR then tells how many falsely given negatives are received from the positive cases. [Walter 2005] Values can be received with the following formulas:

$$True\ Positive\ Ratio\ (TPR/sensitivity) = \frac{TP}{TP + FN} = 1 - FNR$$

$$False\ Positive\ Ratio\ (FPR) = \frac{FP}{FP + TN} = 1 - TNR$$

$$True\ Negative\ Ratio\ (TNR/specificity) = \frac{TN}{TN + FP} = 1 - FPR$$

$$False\ Negative\ Ratio\ (FNR) = \frac{FN}{FN + TP} = 1 - TPR$$

There are limitations to the ratios of the TP, TN, FP and FN, because there is no hands-on use for them in an estimation of a disease for individual patients. That is why predictive values are calculated. These metrics tell how likely it is that patient has a disease when tested positive, or does not have a disease when tested negative. [Akobeng 2007] Metrics can be calculated with the following formulas:

$$Positive\ Predictive\ Value(PPV) = \frac{TP}{TP + FP}$$

$$Negative\ Predictive\ Value\ (NPV) = \frac{TN}{TN + FN}$$

If the disease prevalence is taken into account, in calculations ratios of the TP and TN are used instead of number of TPs and TNs. [Altman and Bland 1994] Calculation can be made with these formulas:

$$Positive\ Predictive\ Value\ (PPV) = \frac{sensitivity \times prevalence}{sensitivity \times prevalence + (1 - specificity) \times\ (1 - prevalence)}$$

$$Negative\ Predictive\ Value\ (NPV) = \frac{specificity \times (1 - prevalence)}{(1 - sensitivity)\ \times\ prevalence + specificity \times\ (1 - prevalence)}$$

ROC curve with AUC is widely used in ML problems, especially in the medical diagnostic ML problems. With this it is possible to receive measurement of performance

in the diagnostic rules and classification. The curve will visualize ratios of TPs and FPs, and that will indicate what FPR is expected with the specific TPR with different accuracy thresholds. AUC is a single value which will tell percentage of the space under the curve, and it is indicating the performance. [Hand 2009]

In the training of the model, CCE is used as loss functions, and ACC is showing the accuracy of classifications. TensorFlow's [2021] accuracy is a simple calculation which equals to how many labels are correctly classified and as a mathematical formula it can be presented:

$$Accuracy\ (ACC) = \frac{Number\ of\ correct\ classifications}{Total\ classifications}$$

CCE is multiclass loss function which calculates cross entropy between real labels and classifications, and the main objective is to minimize it in the training of the neural network. [Chen, C. *et al.* 2020] The metric can be calculated with the formula below, where n is the size of the dataset, m presents number of classes to classify, vector d contains binary values where one presents correct label, and incorrect labels are zeros in other words vector d is one-hot encoded and y is probability vector containing probability of each class:

$$Catregorical\ Cross\ Entropy\ (CCE) = -\sum_{i=1}^{n}\sum_{j=1}^{m} d_{ij}\log(y_{ij})$$

3. Machine learning models

In this chapter I will introduce my method, comparing it to other methods. It will be compared with residual neural networks ResNet18 and InceptionTime, and with boosted tree classifier XGBoost.

3.1. My model

My model contains input layer, normalization layer, concatenate layer, max pooling layers, convolution layers, flattening layer, output layer, dropout layer and add layers which combine residual connections to neural network. Full architecture can be seen in Figure 10, which will show the order of the layers. As my model is convolution neural network with residual connections, so it makes it residual neural network.

Input layer controls the input shape of the data. TensorFlow has a few different ways to do this. In this model SparseTensor is used with the DenseTensor. SparseTensor is used, because the data contains lots of zero values, and the training, validation and testing would be slower as more data is transferred to the GPU. Missing values would have been replaced with zeros. SparseTensor contains only the information of the real values and indices of the values, and the dense shape, which means the shape of the original data given in the input layer.

Normalization layer is a built-in layer by TensorFlow, and it normalizes data's individual features by their mean and variance. Concatenation layer is needed as the data is received from multiple systems, and there are few different feature maps which needs to be combined so there will be just one dataset per patient. Max pooling is done before the convolution layers and after the convolution layers. The idea is to reduce data without losing important data, such as reducing the number of zeros, and squeezing the data from time series of daily data to 10-day time intervals, as it is not likely that a patient is tested very often with the same tests. Flattening is done, because it is easier for neural network, and the dense layer is expecting it in one dimensional vector form. *Flattening* means that the result matrix is converted to 1-dimensional array. The output layer is the dense layer, which will produce probabilities of the individual classes.

In the convolution blocks there are three different kinds of convolution layers. The filters of these layers are different. The first one has a shape of 10×1, the second has 1×10, and third has 1×1. The third one is used only in the convolution block B, where it is providing residual connection. Column convolutions are inspecting the individual tests if there are any changes in the timeseries or typical values to some rare diseases. Row convolution is inspecting if there are combination of tests done to the patient, which are typical for the rare diseases. Add layer combines the residual connection to the main path, which allows the skip of two convolution layers. Dropout layer at the end is used, because

when developing it, it was noticed that this model was overfitting and learning the training data really fast. Dropout layer did help to prevent this issue.

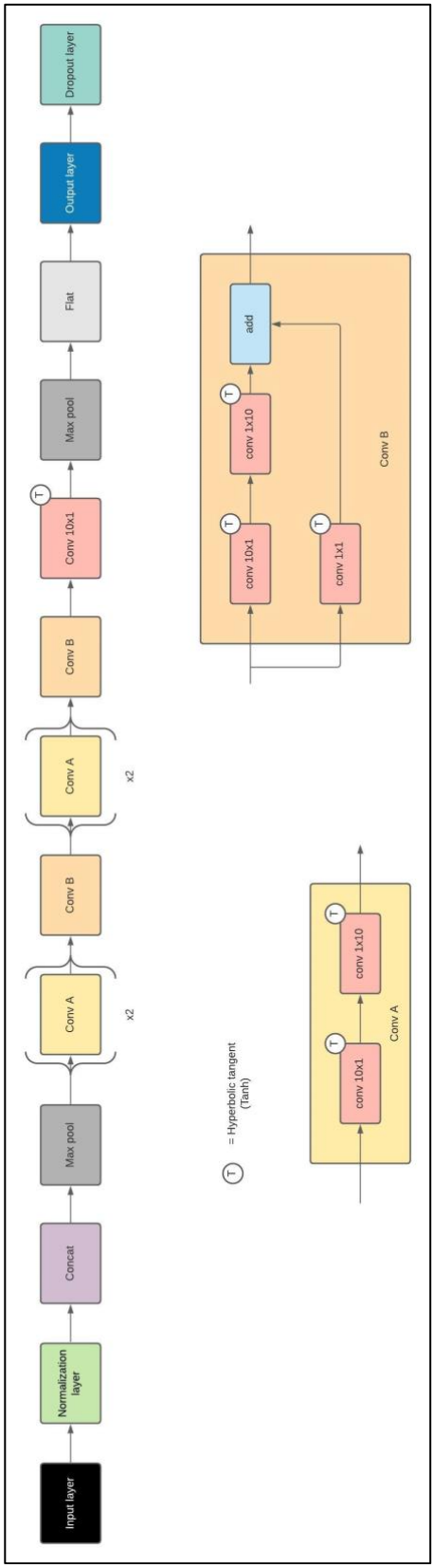

Figure 10. Architecture of my residual neural network.

3.2.ResNet18

ResNet18 is one of the suggested residual neural networks He *et al.* [2016]. Due the limitations of our system's performance it was not possible to use ResNet34 or higher, which would have been more ideal. As it was found out that the ResNet performed better with more layers, and the 34-layer was the divider where performance increase started to show significantly [He *et al.* 2016].

ResNet was originally tested with CIFAR-10 dataset, which is an image dataset, so the use purpose of ResNet is very different than in this thesis' use case. [He *et al.* 2016] Despite that, it is the state-of-the-art method and the original residual neural network, so it could be used as a baseline of residual neural networks. Data used in this thesis is very unconventional for residual neural networks, so expectations are not very high as the model should be developed based on the data and not the other way around.

3.3. InceptionTime

Ismael Fawaz *et al.* [2019] introduced the state-of-the-art *InceptionTime* residual neural network model, because the timeseries classification performance is still falling behind the image recognition performance. Compared to earlier models, InceptionTime is faster and highly scalable. HIVE-COTE model was considered the best in the accuracy in this field, but rarely implemented to the real-world applications. This was due slowness in training, which is characterized in time complexity as $O(N^2 * T^4)$. [Ismail Fawaz *et al.* 2020]

Due to the limitation of the system, it was not possible to run the InceptionTime with two residual connections as Fawaz *et al.* [2020] originally stated. In this thesis, the InceptionTime was tested with only one residual connection, as we had only two GPUs in comparison to Fawaz *et al.* [2020] having remote cluster which contained more than 60 GPUs. This method is expected to have some performance over this data, because it has similarities with timeseries data, although quite sparse one, which may have some effect on the performance.

3.4. XGBoost

XGBoost is a state-of-the-art ML method. It is tree boosting system which is highly effective with sparse data, and can be used to solve many different problems. XGBoost has been proven to be highly used in the field of data science, because its scalability and minimal resource usage. [Chen, T. and Guestrin 2016]

Data used in this thesis being very sparse this is very promising ML method and it is selected for comparison based on that. In addition, promises of XGBoost having minimal

usage of resources, it should not have the same system limitation issues as residual neural networks as stated earlier.

4. Results

In this chapter the results are shown and briefly introduced. I will go through the results and select a few ML methods to provide a deeper comparison. In comparison, different sizes of masks are used, so we can see how much earlier models could get indications of rare diseases.

4.1. Overall learning of the neural networks

Figures 11 and 12 are showing that the models of the InceptionTime and ResNet18 are not learning from the data. My model on the other hand is learning. ACC is calculated with its formula, and the loss is calculated with CCE formula. These values are calculated after each epoch, and the training time was 10 epochs. The same data source was used in this test as in the other tests, but here the dataset size was a bit smaller, so that the models which are learning could be identified early, and to make sure they are productive to inspect more closely. The smaller dataset was the same with all of the models.

In Figure 11, the loss is not decreasing for the InceptionTime or ResNet18 in the training or validation. It is decreasing only for my model for both training and validation. Validation of ResNet18 shows some movement in the loss, as it rapidly increases for the epochs eight and nine. It is unclear what happens, as the accuracy for that is not changing significantly. My model shows that it is not overfitting or underfitting. This is because the validation and training loss is close to each other.

In Figure 12 the accuracy is not increasing for the InceptionTime or ResNet18. Some change is visible for the validation of InceptionTime, but it is not increasing significantly. My model shows increase of training and validation. Validation is a bit lower than the training, which is normal behaviour. My model has accuracy of about 55% and the training is getting close to 65%.

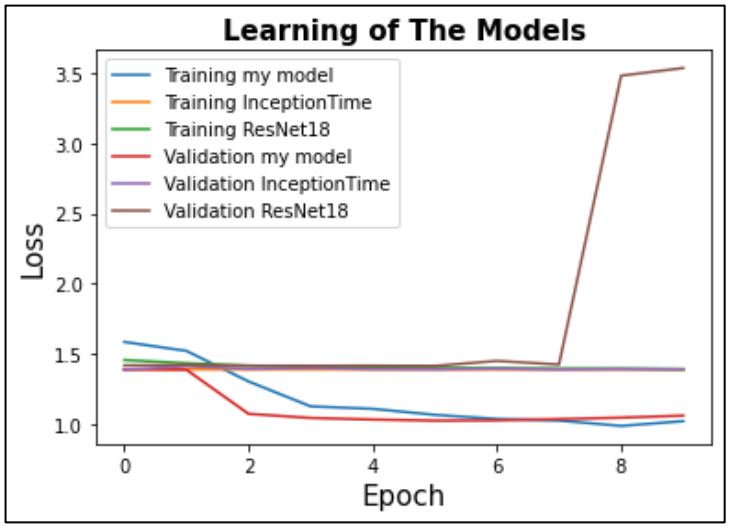

Figure 11. Loss of the training and validation of the models.

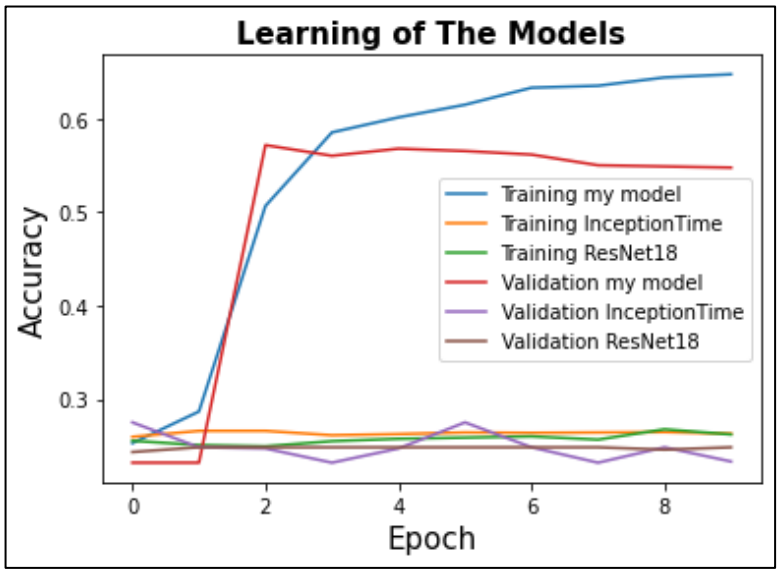

Figure 12. Accuracy of the training and validation of the models.

4.2. 30 days masking

In this the data is masked with 30 days, which means that classifications are done 30 days before the doctor gave the initial diagnosis of the rare disease.

4.2.1. My model

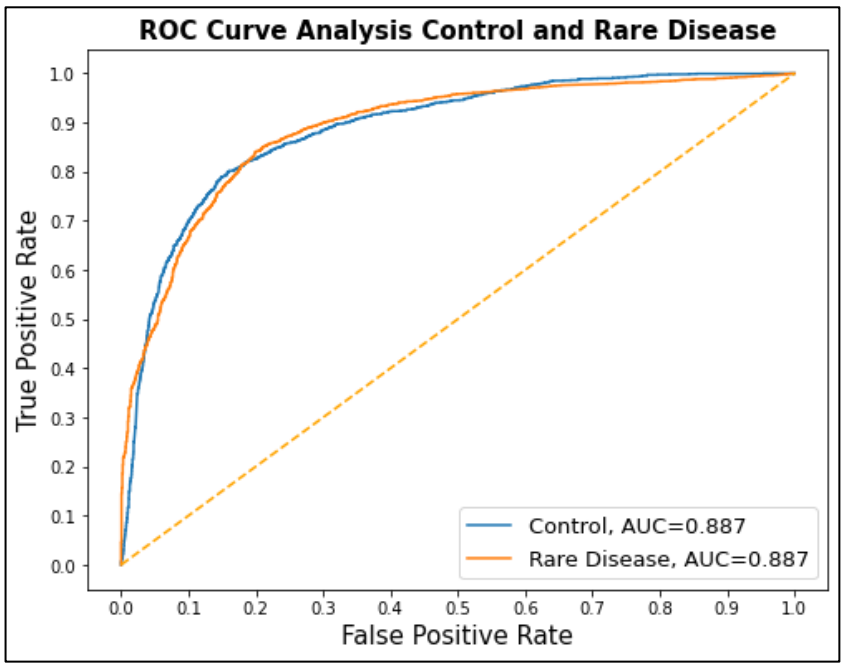

Figure 13. ROC curve of control and rare disease classifications with AUC value.

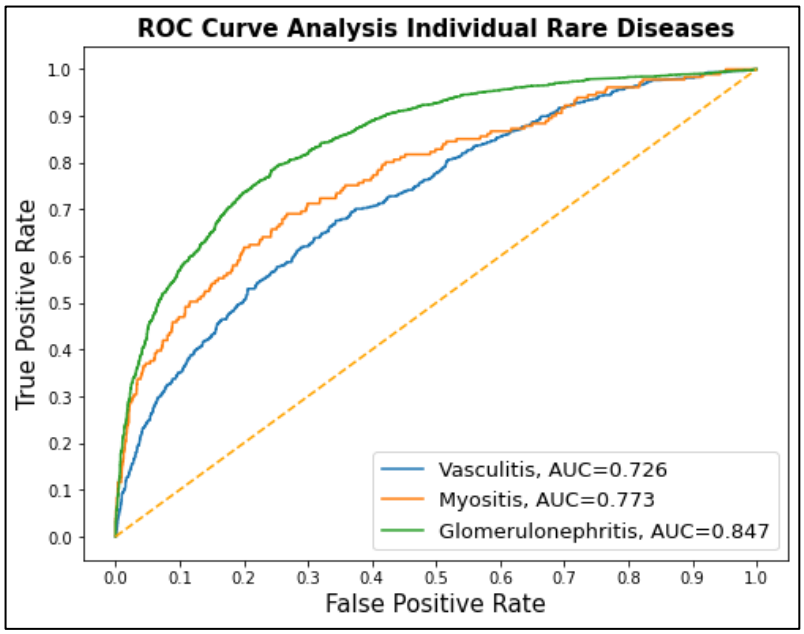

Figure 14. ROC curve of individual rare diseases with AUC value.

Figure 13 and 14 shows ROC curves with AUC value of control-, rare disease- and individual rare disease classification.

In Figure 13, the control and rare disease AUC values are for both 88.7%. This indicates very good performance of the classification. The optimal value for both would be in theory close to 80% in the TPR. It is the closest to the top left corner.

Figure 14 show a decrease of the AUC values of individual diseases compared to Figure 13 rare disease curve. This is quite expected as the individual disease is harder to classify and separated from each other. Still Glomerulonephritis classification is very good with the value of 84.7%. Myositis with value 77.3% and vasculitis with value 72.6% are good results. Good thresholds in theory for these three would be something between 70% and 60% of TPR. At the same time, FPR would be about 20% for glomerulonephritis, 25% for myositis and vasculitis would have a bit below 30%.

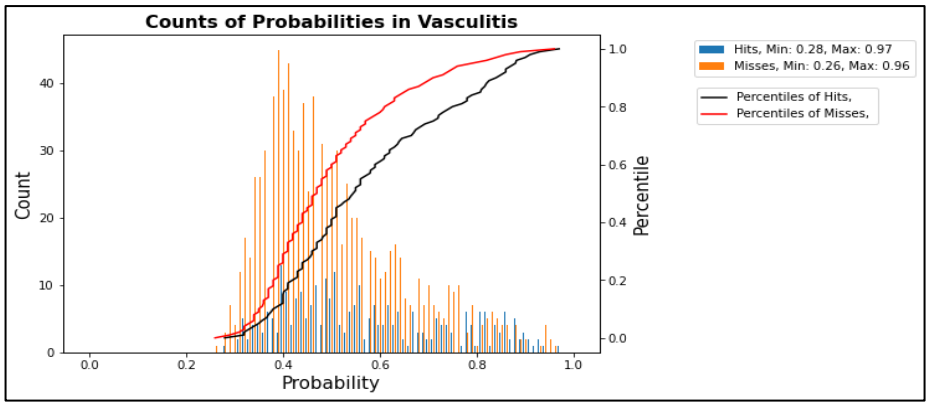

Figure 15. Hits and misses of vasculitis by their probabilities and percentile showing how values are distributed.

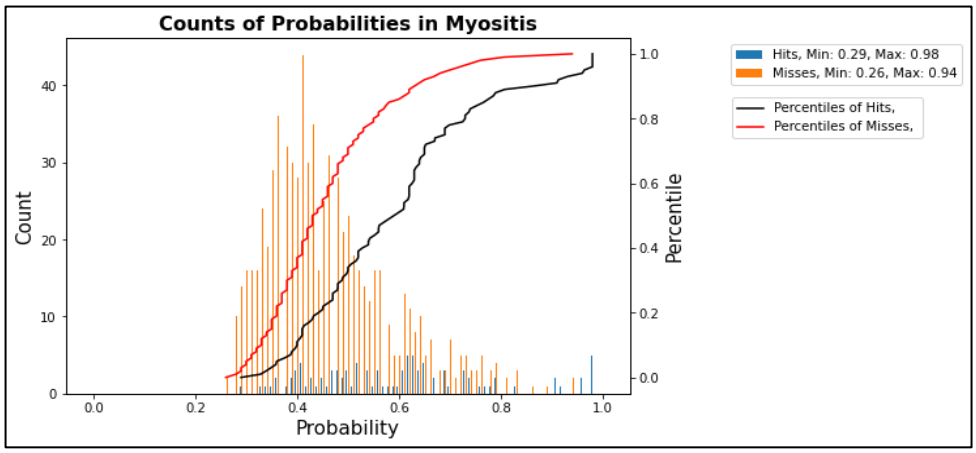

Figure 16. Count of hits and misses of myositis by their probabilities and percentile showing how values fall below specific probabilities.

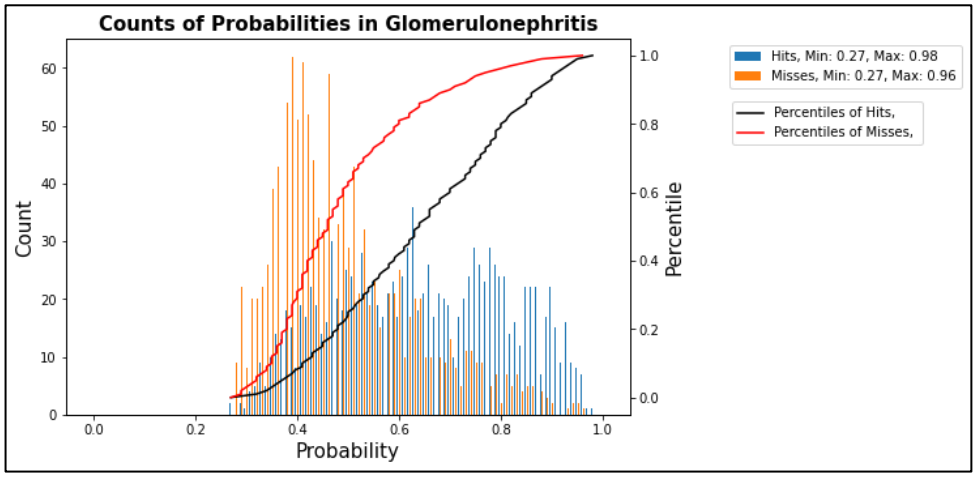

Figure 17. Count of hits and misses of glomerulonephritis by their probabilities and percentile showing how values fall below specific probabilities.

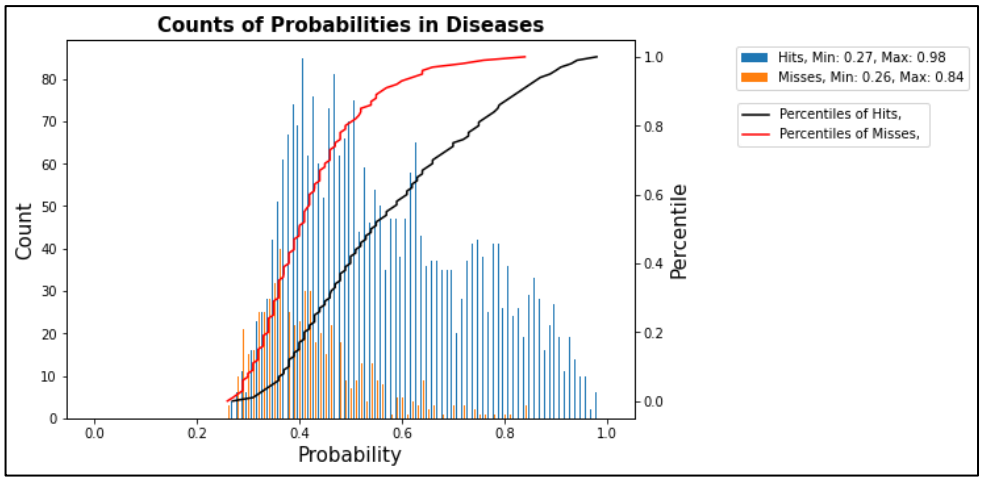

Figure 18. Count of hits and misses of diseases by their probabilities and percentile showing how values fall below specific probabilities.

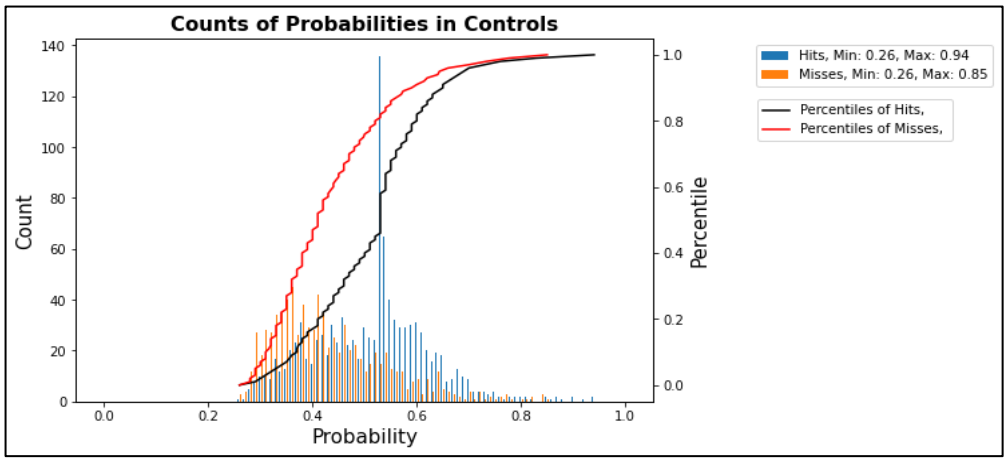

Figure 19. Count of hits and misses of control by their probabilities and percentile showing how values fall below specific probabilities.

Figures 15 to 19 are showing the hits and misses count in vertical axis and probabilities in horizontal axis. Probability of each hit and miss is the highest probability of the patient's probability distribution which was received from the prediction. Probability values are rounded to the nearest percent, so the values are between 0% and 100% with incrementation of 1%, so there are 101 possible probabilities in this graph. Percentiles of hits and misses are shown as well, and it helps to see how much values are falling below the different probability thresholds. In addition, min and max probabilities of hits and misses are given. The hits include only correctly classified labels, and the misses contain falsely classified labels in two ways: either correct label is falsely predicted to something else, or incorrect label is predicted inspected label.

Figure 15 shows the results of vasculitis. The minimum of hits is 28% and the maximum is 97%. For misses, the minimum is 26% and the maximum is 94%. Graph shows that vasculitis is very hard to classify correctly because of lots of misses. Hits and misses are both decreasing when probability goes higher, and misses are decreasing faster. In addition to that, the last miss comes in the 96% and the last hit came in the 97%, which is quite a small margin. Despite that, it would seem that almost for every threshold there are more misses than hits.

In Figure 16 there are the results of myositis. It shows similar result than the vasculitis, because there are more misses than hits. In addition, the misses start to decrease around probability of 45%. The minimum of hits is 29% and the maximum is 98%, and the minimum of misses is 26% and the maximum is 94%. This is a bit better than the vasculitis results in Figure 15, as the last miss and the last hit difference is 4%, which is a bit better. Still, this is quite a small margin overall.

Glomerulonephritis in Figure 17 shows better results than the other two diseases. The minimum value of hits is 27%, and the maximum is 98%. The misses have a minimum value of 27% and the maximum of 96%. The misses start to decrease around 50% and the hits have quite an even increase of percentile, which means that the count of the hits stays

stable. Also, after probability level of 60% there are more hits than misses in the individual probability thresholds.

Figure 18 presents great results of prediction of diseases. The hits have a minimum of 27% and maximum of 98%, and the misses have a minimum of 27% and maximum of 84%. There are significantly more hits than misses in across the threshold probabilities. Both hits and misses decrease on their way closer to probability of 100%. The difference of maximum hits and misses is significantly better than the individual diseases as it is 14%.

The control patient results in Figure 19 show good results, but not as good the diseases result in Figure 18. The maximum of hits is 94% and the minimum is 27%. For misses, the minimum is 26% and the maximum 85%. The decrease of misses can be seen in the graph, and the hits will pass misses in around the probability of 50%. The hits start to decrease after 60%. The hits maximum is 9% bigger than the misses maximum.

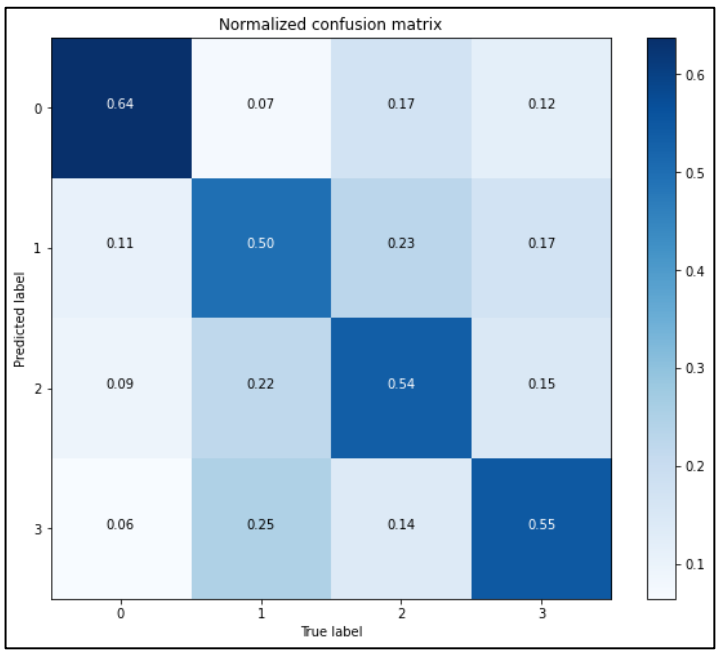

Figure 20. Confusion matrix of the predicted labels and true labels. Class zero is control, one is vasculitis, two is myositis and three is glomerulonephritis.

Figure 20 is confusion matrix of predictions and true labels, and it is showing how hits and misses are distributed with different classes. This shows that every class gets at least half of the predictions correct. In addition, some difference can be seen because more miss classifications happen when control patients are falsely classified with a disease, than other way around, but with an exception of vasculitis. Most of the disease prediction errors are made by predicting it to other disease. Notable is also that the control is a bit more often predicted with a disease as compared to the disease's predictions to the control. Vasculitis is an exception here.

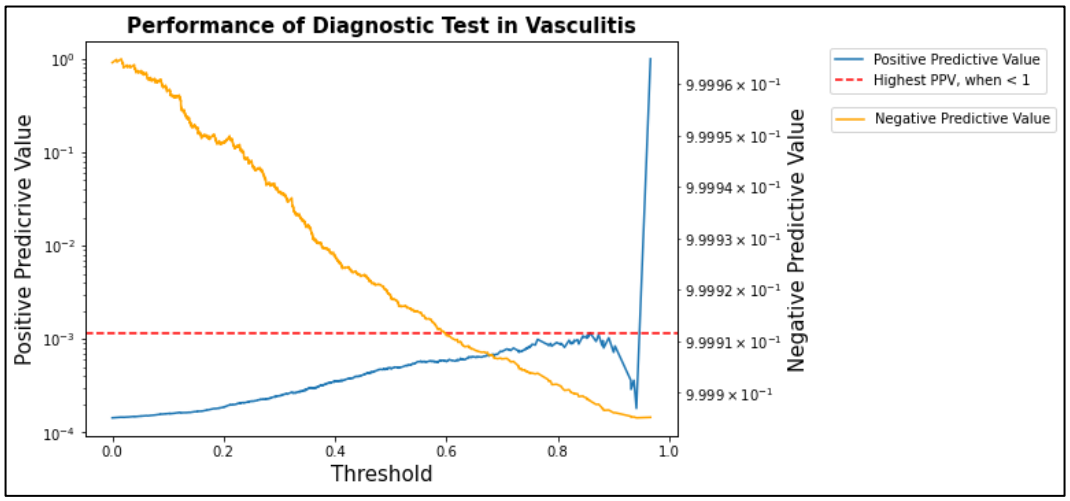

Figure 21. PPV and NPV for the vasculitis.

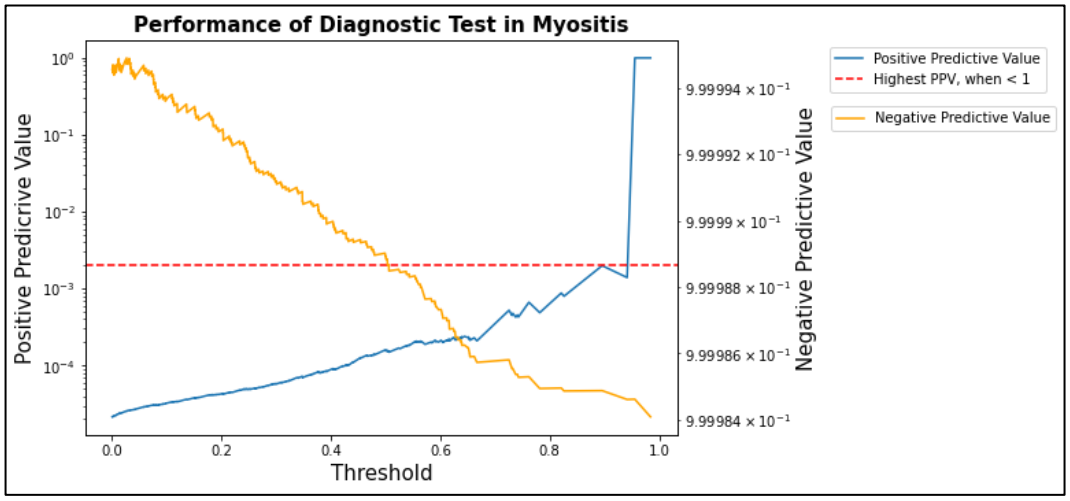

Figure 22. PPV and NPV for the myositis.

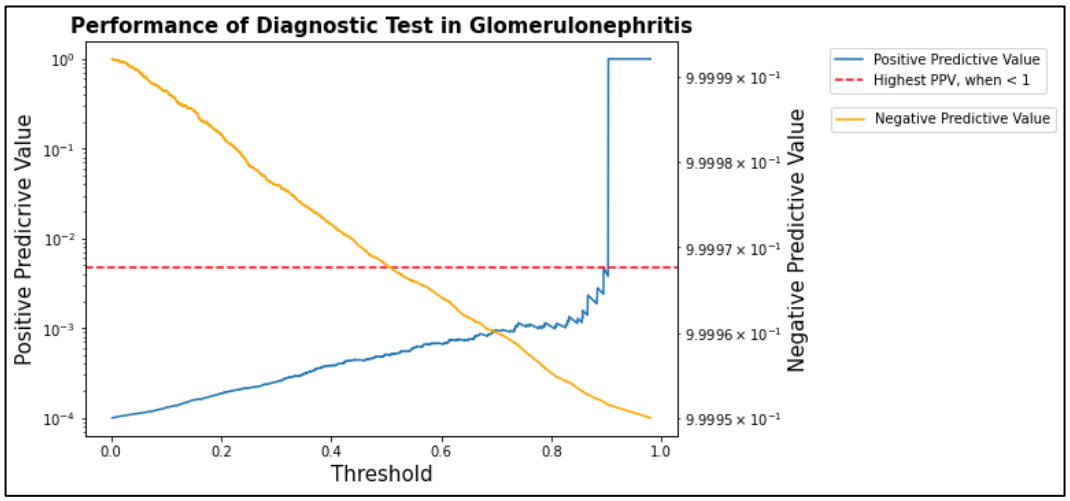

Figure 23. PPV and NPV for the glomerulonephritis.

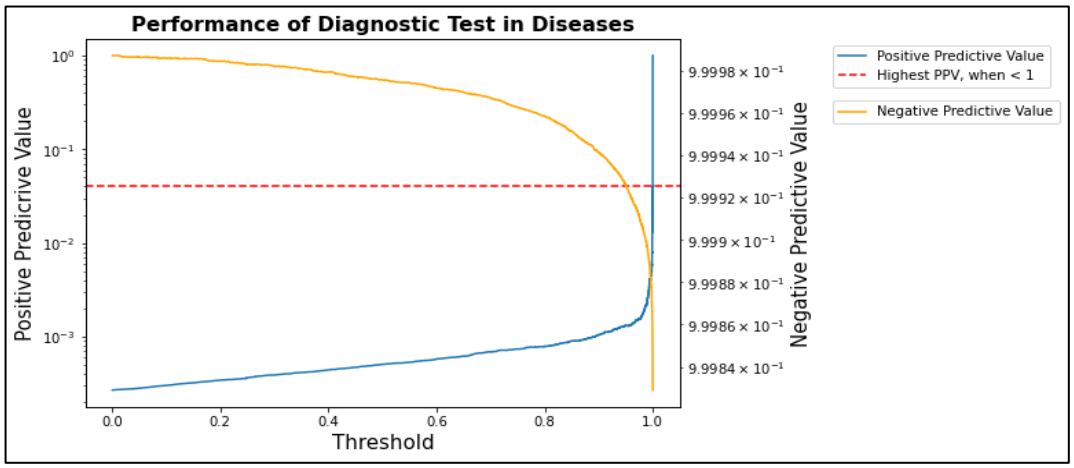

Figure 24. PPV and NPV for the diseases.

	PPV	NPV	THRESHOLD
VASCULITIS	0.1%	99.99%	86%
MYOSITIS	0.2%	99.998%	90%
GLOMERULONEPHRITIS	0.5%	99.995%	90%
DISEASES	4%	99.985%	99%

Table 3. Highest score of PPV which is below one and corresponding NPV and threshold value for that.

Figures 21 to 24 are showing the PPVs and NPVs, which will help to determine the best threshold for neural network classification, as it can be used to calculate the ratio of expected TPs and FPs, and TNs and FNs. Table 3 shows the highest peak of the PPV and corresponding values for NPV and threshold values. The threshold is probability value which works as a limit, and when the prediction's probability falls below that, the class is changed to control. In practice it is a certainty threshold, for example, if the prediction's certainty for a disease is not high enough, the patient cannot be said to have a disease. The threshold can be changed over the time and it is highly dependent of resources of the healthcare centers and hospitals.

Figure 21 shows that vasculitis NPV stays above 99% for all the thresholds, and the PPV increases till 86% threshold, where there is the highest peak of 0.1% and after that it is decreasing. This means that for every TP it is expected to have 999 FPs and with NPV value of 99.99% we could expect one FN for every 9999 TNs.

The myositis results are presented in Figure 22. PPV is increasing till 90% of threshold. With this threshold, the peak of PPV of 0.2% was achieved, and this value indicates that the expected number of FPs for one TP would be 499. NPV value in the

peak point of PPV is 99.998%, which indicates that the expected value of FNs would be one in 49999 TNs.

Figure 23 for glomerulonephritis disease's PPVs increases the most out of the individual diseases. It increases till 90% of threshold and gets peak of 0.5%. This value would expect that for every TP there is 199 FPs. NPV is 99.995% for the best PPV, and with this it can be assumed that for one FN there would be 19999 TNs.

Figure 24 describes that the patient does have some of the three diseases. PPV increases till 99% and gets peak of 4%. We could assume with this value that for every TP there would be 24 FPs. As the NPV is 99.985%, we can estimate that for every FN there are about 6666 TNs.

4.2.2. XGBoost

XGBoost's ROC results with the AUC score can be seen in Figures 25 and 26.

Figure 25 shows great scores AUC of 96.4% for both control and disease. The best choice would be about 90% of TPR for both curves. In those situations, FPR would be quite minimal with below 10%.

In Figure 26, the highest score of AUC is for glomerulonephritis with 89.6%, then vasculitis with score 80.8%, and myositis has the score of 75.7%. In addition, myositis has a big dip starting at about 10% of FPR. Small dip can be seen for vasculitis after 25% of FPR. The best choice for the best result would be for glomerulonephritis about 80%, for vasculitis about 70%, and for myositis something between 55% to 60%. At the same time, FPR would be about 15% for glomerulonephritis, vasculitis would have about 20%, and myositis about 10%.

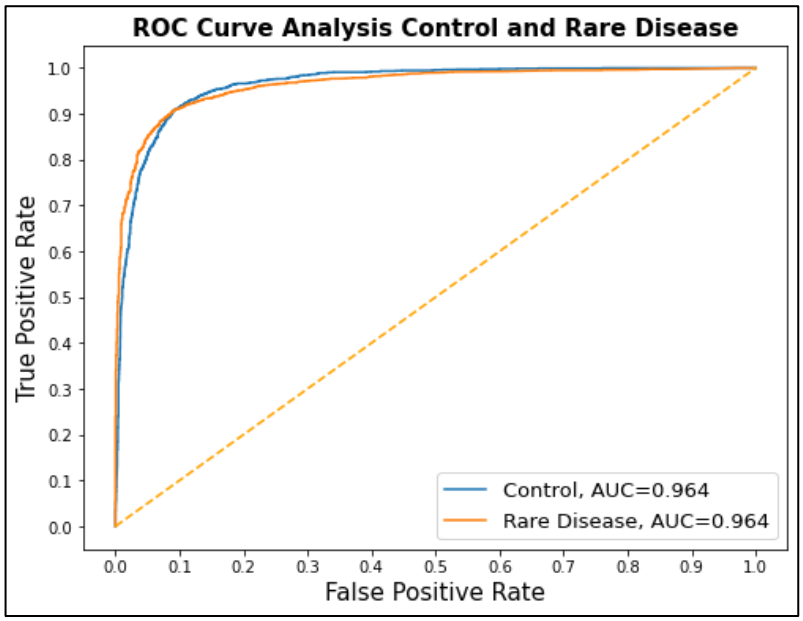

Figure 25. ROC curve of control and rare disease classifications with AUC value.

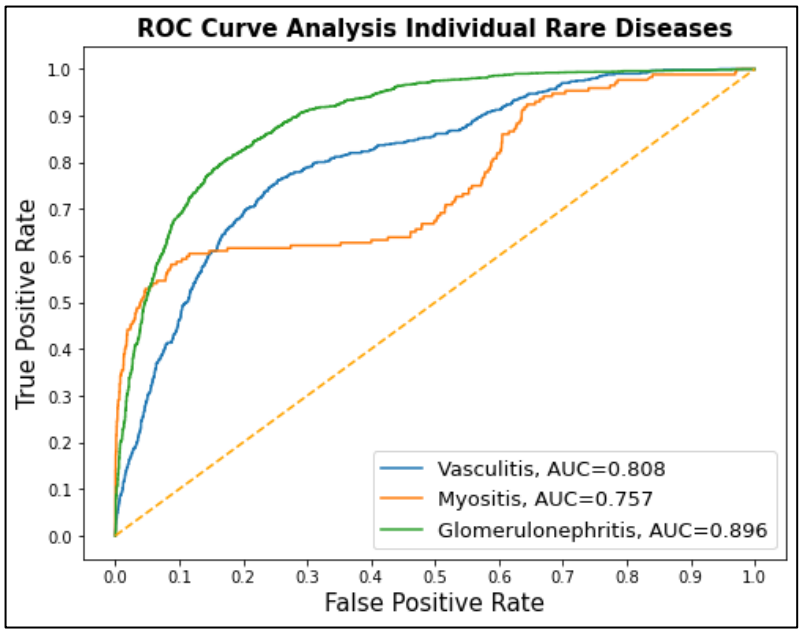

Figure 26. ROC curve of individual rare diseases with AUC value.

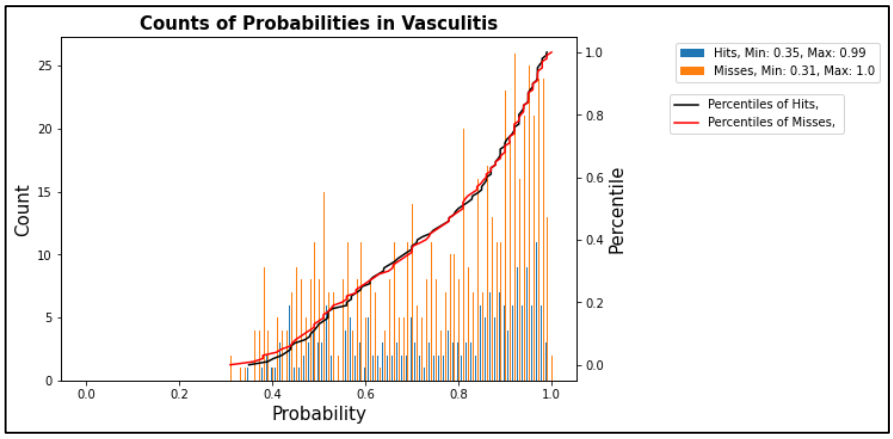

Figure 27. Hits and misses of vasculitis by their probabilities and percentile showing how values are distributed.

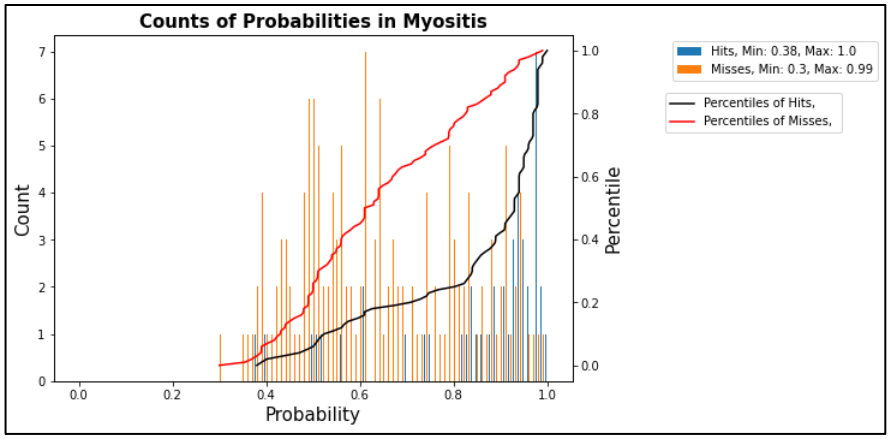

Figure 28. Count of hits and misses of myositis by their probabilities and percentile showing how values fall below specific probabilities.

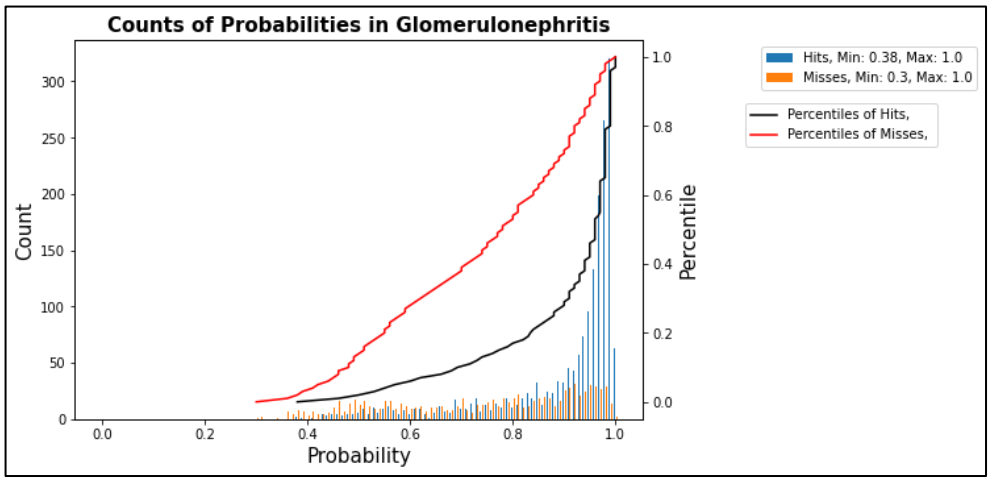

Figure 29. Count of hits and misses of glomerulonephritis by their probabilities and percentile showing how values fall below specific probabilities.

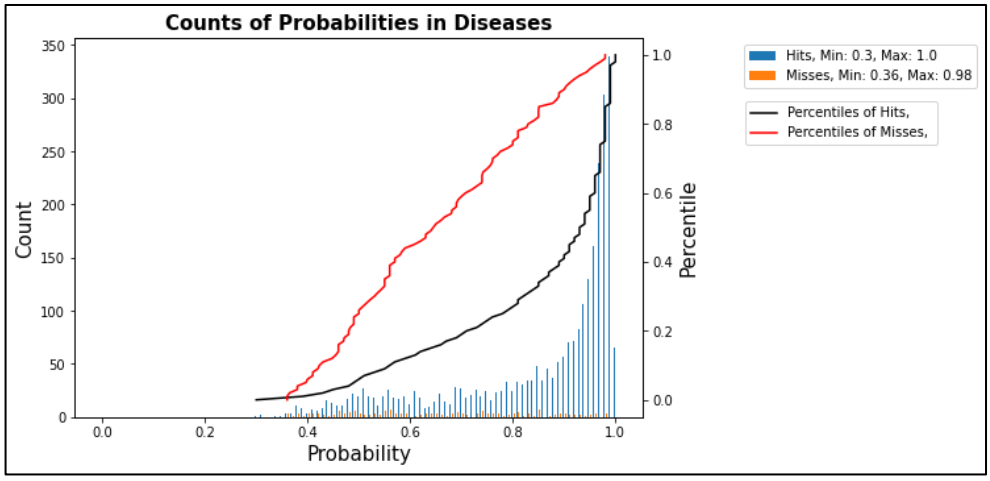

Figure 30. Count of hits and misses of diseases by their probabilities and percentile showing how values fall below specific probabilities.

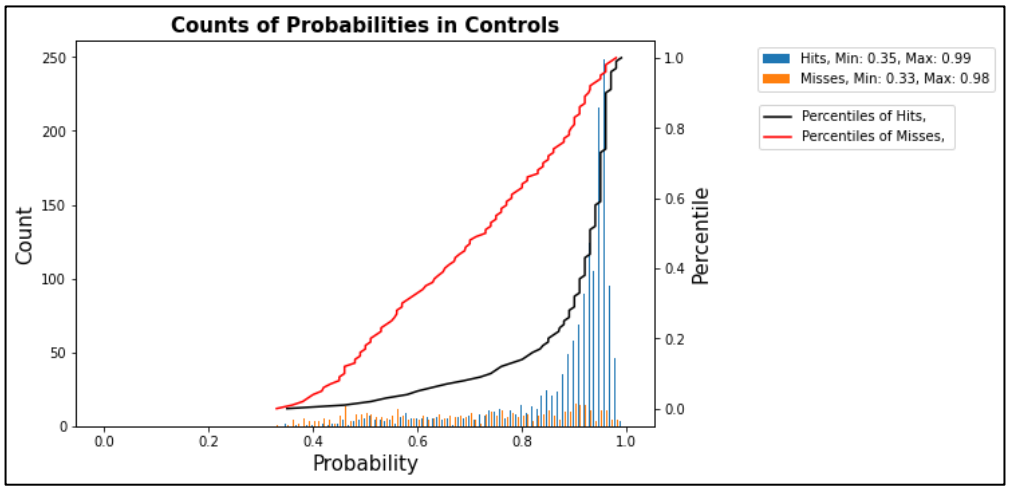

Figure 31. Count of hits and misses of control by their probabilities and percentile showing how values fall below specific probabilities.

Figures 27 to 31 are describing hits and misses of individual diseases, for all diseases and control, and the minimum and maximum values for hits and misses and percentile curves.

Figure 27 is showing vasculitis hits and misses. There are a lot more misses than hits, and in addition to that, the maximum of misses is bigger than hits when the misses have 100% and hits have 99%. The difference is 1%.

Myositis Figure 28 shows small improvement from vasculitis's results, but it is not good enough. Maximum of hits is 100% and for misses 99%. Difference is 1%. The hits are distributed better as it increases from 90% to 100%.

Figure 29 are the results for glomerulonephritis. Here the performance can be seen to improve significantly. The hits start to increase from 80% to 100%. Also, misses increase from the start till end, but there are a lot less misses than hits. The maximum for both hits and misses is 100%.

Figure 30 has results of the diseases' hits and misses, and it shows that it can classify diseases as one very well. The maximum for hits is 100% and for misses 98%. The reliability of decision-making increases towards the end.

Control Figure 31 shows similar results as glomerulonephritis and diseases. The control patient classifications are very successful, and the reliability is increasing towards maximum. The maximum value for misses is 98% and for hits 99%.

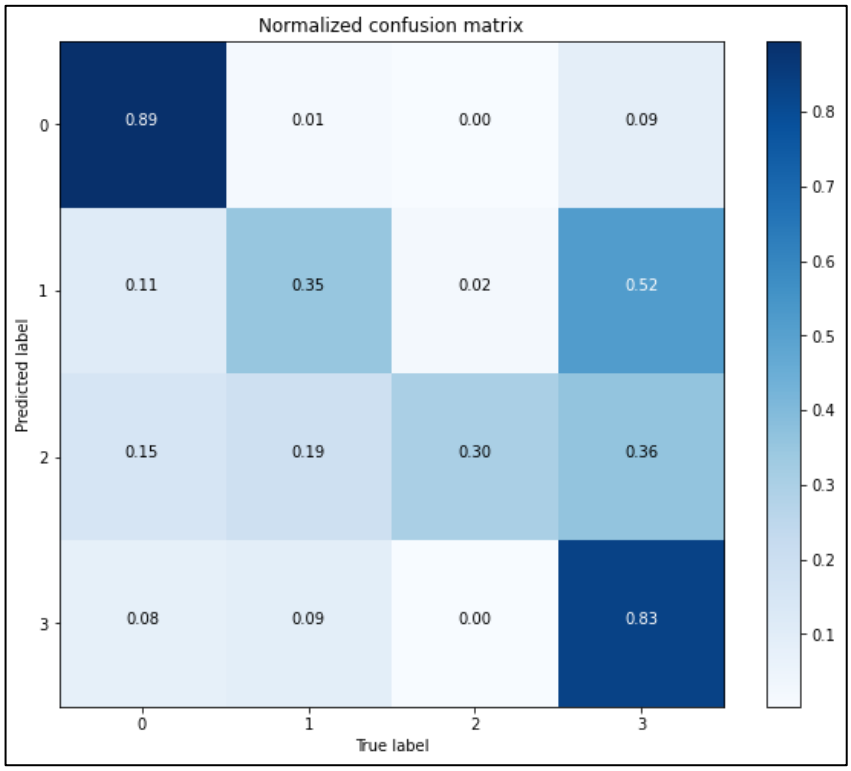

Figure 32. Confusion matrix of the predicted labels and true labels. Class zero is control, one is vasculitis, two is myositis and three is glomerulonephritis.

Figure 32 shows confusion matrix of the prediction- and true labels and how those are distributing among other classes.

Control is predicted very well with the highest accuracy of 89%, and when the control patient is classified, it is rarely classified wrongly. Then glomerulonephritis is also classified significantly better than the other diseases. It has an accuracy of 83%. Vasculitis is classified mostly wrongly. It is falsely classified to glomerulonephritis with an accuracy of 52%, and correctly classified with an accuracy of 35%. Myositis is also falsely classified to the glomerulonephritis with an accuracy of 36%. Correct classification is done with the accuracy of 30%. Myositis appeared to be the hardest to classify correctly, as other miss classifications happen also quite often. The control patients are rarely miss classified with a disease, and other way around the patients with disease are more often classified as control, with an exception of glomerulonephritis. When diseases are thought as one, it can be seen that works quite well. In addition, it interesting to point out that other diseases or control are rarely miss classified as myositis, but the other way around it performs the worst.

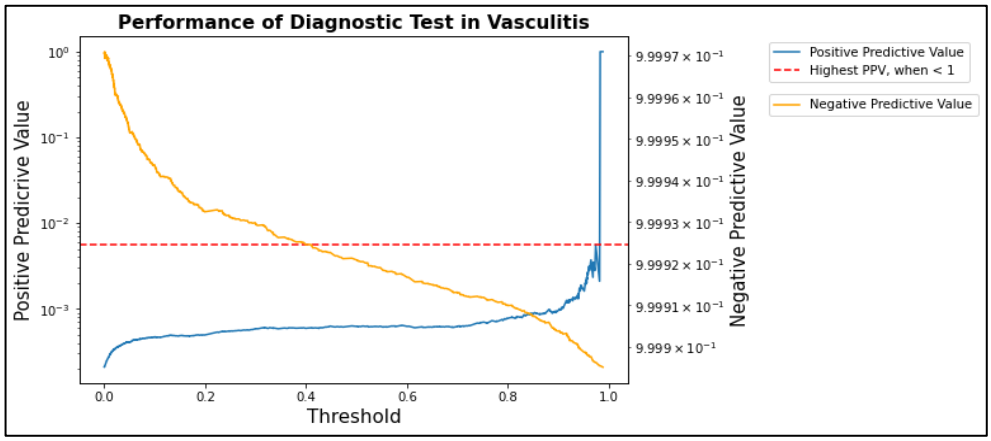

Figure 33. PPV and NPV for the vasculitis.

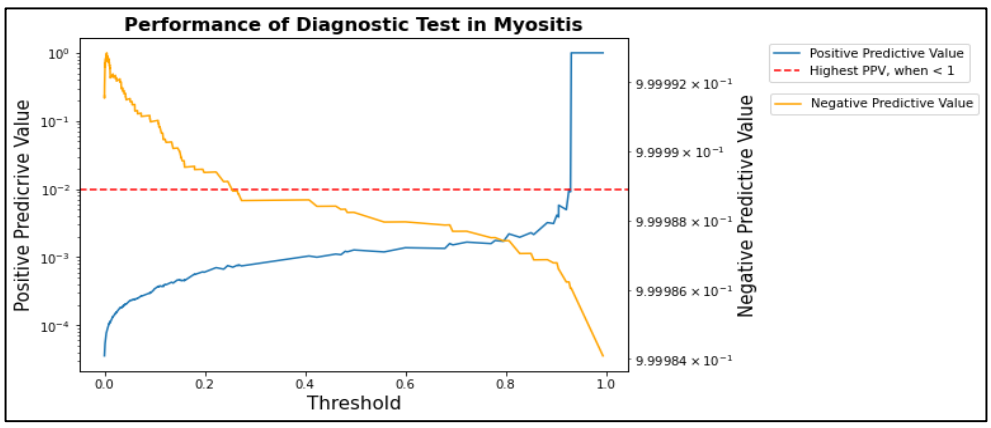

Figure 34. PPV and NPV for the myositis.

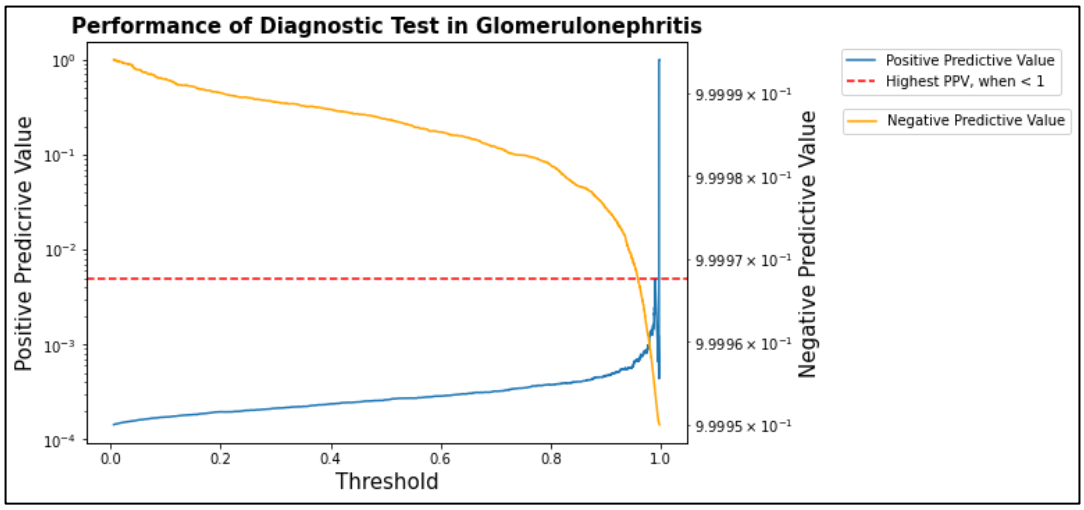

Figure 35. PPV and NPV for the glomerulonephritis.

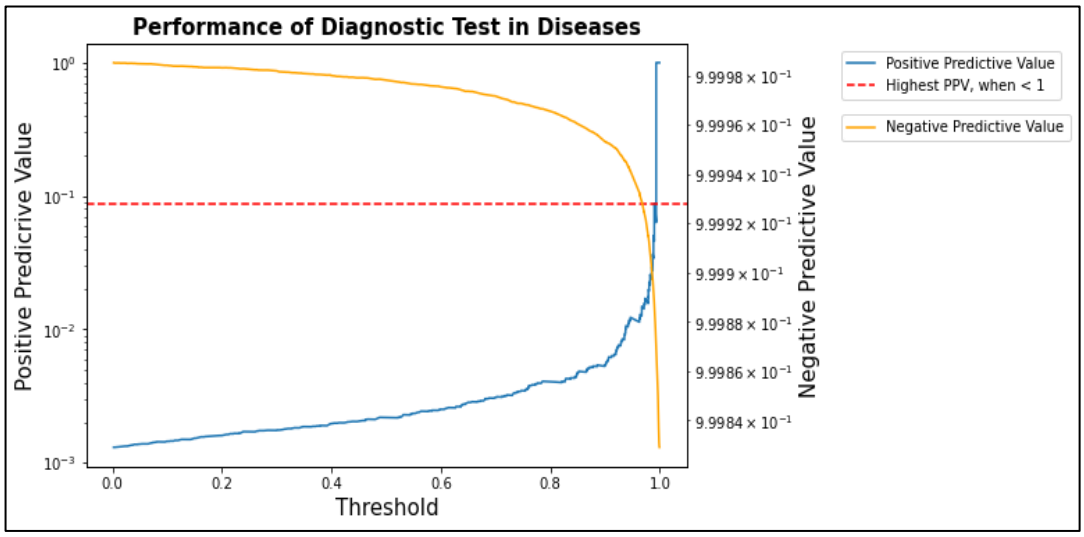

Figure 36. PPV and NPV for the diseases.

	PPV	NPV	THRESHOLD
VASCULITIS	0.6%	99.99%	97%
MYOSITIS	1%	99.999%	93%
GLOMERULONEPHRITIS	0.5%	99.996%	99%
DISEASES	8.8%	99.989%	99%

Table 4. Highest score of PPV which is below one and corresponding NPV and threshold value for that.

Figures 33 to 36 are presenting PPVs and NPVs with horizontal line indicating highest peak of PPV. In addition, there is Table 4 to show the exact peak values of PPVs and corresponding NPVs and thresholds.

Figure 33 shows vasculitis results and the highest peak of PPV of 0.6%. It determines that for every three TPs it is expected to have 497 FPs, and this is achieved with the best threshold of 97%. At the same time, the NPV is 99.99% which means that a patient is rarely given negative test result if a patient truly has a disease, and for exact estimation there would be 9999 TNs against one FN.

Myositis result in Figure 34 shows better result than the vasculitis as highest peak of PPV is 1%. This means there would be expected 99 FPs against one TP. This is reached at 93% threshold, and at the same time the NPV is 99.999%. For these NPV we could expect one FN in every 99999 TNs.

Figure 35 shows the results of glomerulonephritis, and its PPV peak of 0.5% is reached with threshold of 99%. At the same time the NPV is 99.996%, which means there would be expected one FN in every 24999 TNs. Also, for every TP there would be expected 199 FPs.

Disease's Figure 36 shows the best results of these, which is expected. The peak of the 8.8% is reached with threshold of 99%, and at the same time the NPV is 99.989%. PPV indicates that every 11 TPs it is expected to have 114 FPs. In addition, FNs are really rare – about one in every 9090 TNs.

4.3. 2160 days masking

In the following test the data is masked with 2160 days, which means that classifications are done almost six years before the doctor gave the initial diagnosis of the rare disease. The data of the initial diagnosis and six years before that is removed.

4.3.1. My model

Figures 37 and 38 are presenting the ROC curves of my model when the data is masked for 2160 days, which is about six years. These present the ratio of the TPR and FPR. Also, AUC score is given.

Figure 37 shows great results for control and disease. Both have AUC of 93%. In theory, the optimal value would be around 85% for both. This is because that point is the closest top left corner and has the optimal TPR, and FPR is not increasing too much. In that point FPR would be around 10%.

Figure 38 describes good results for individual disease detection. Glomerulonephritis has AUC score of 87.1%, which is the best among the three rare diseases. The second-best AUC score is in myositis with the score of 78.5%, and the third is vasculitis with the

score of 76.3%. A small dip can be seen in the myositis curve, as we did in the XGBoost Figure 26, but not as big. The optimal values for the individual diseases vary, because glomerulonephritis would have optimal TPR around 75%, and then FPR would be 15%. Then myositis would have TPR of about 65%, and FPR of about 20%. Vasculitis would have TPR a bit under 60% and FPR a bit over 20%.

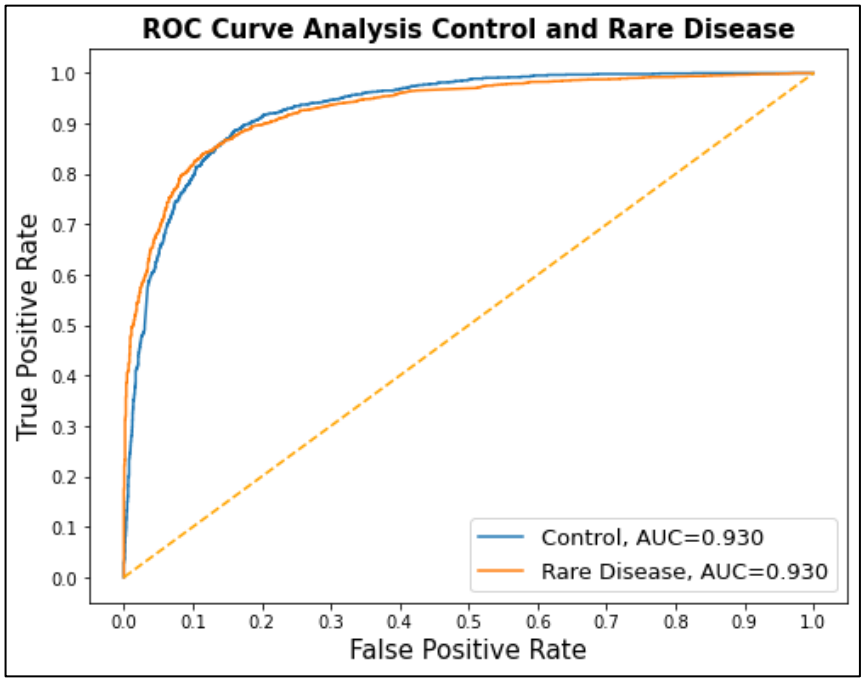

Figure 37. ROC curve of control and rare disease classifications with AUC value.

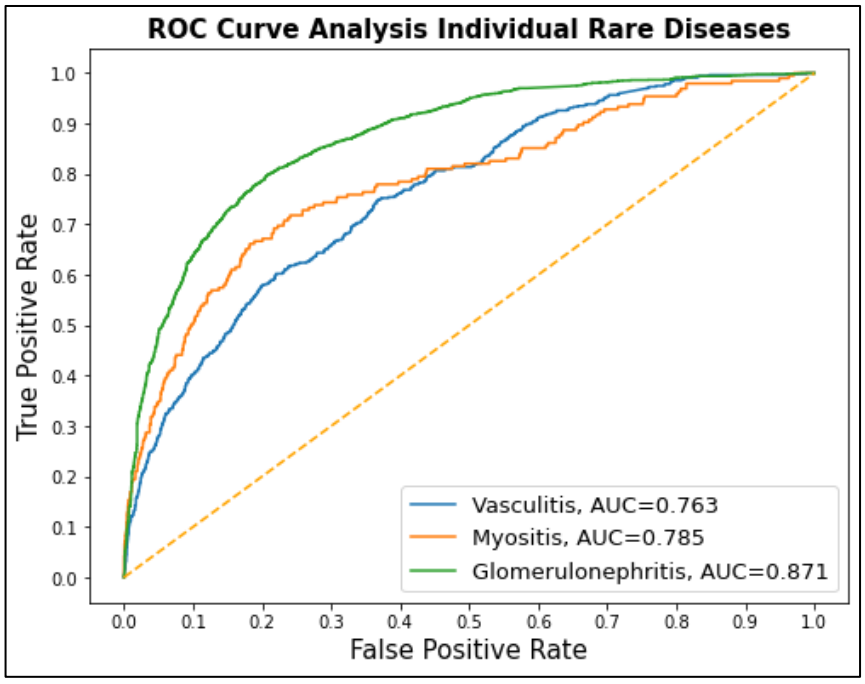

Figure 38. ROC curve of individual rare diseases with AUC value.

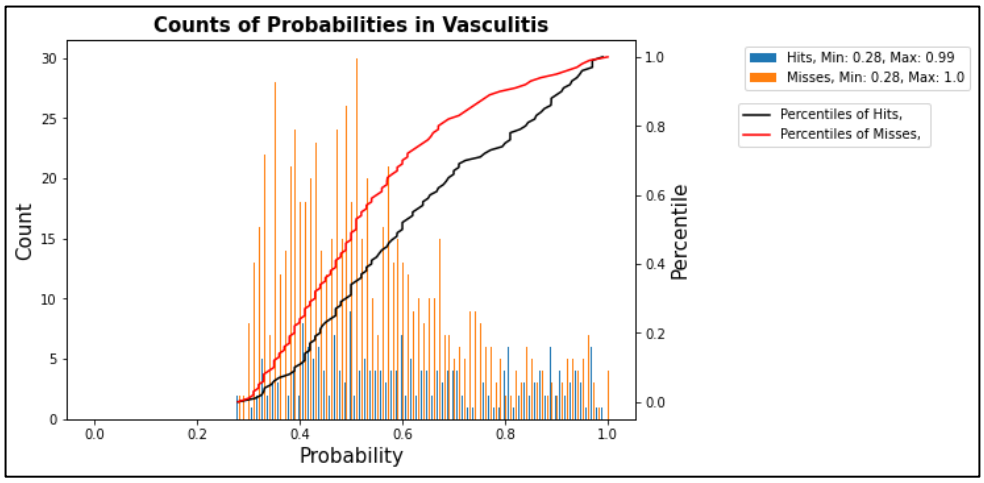

Figure 39. Hits and misses of vasculitis by their probabilities and percentile showing how values are distributed.

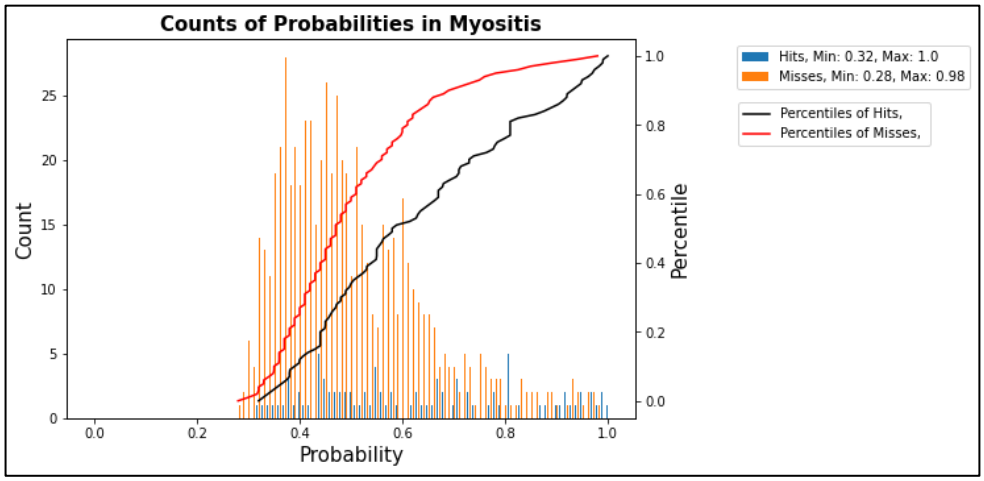

Figure 40. Count of hits and misses of myositis by their probabilities and percentile showing how values fall below specific probabilities.

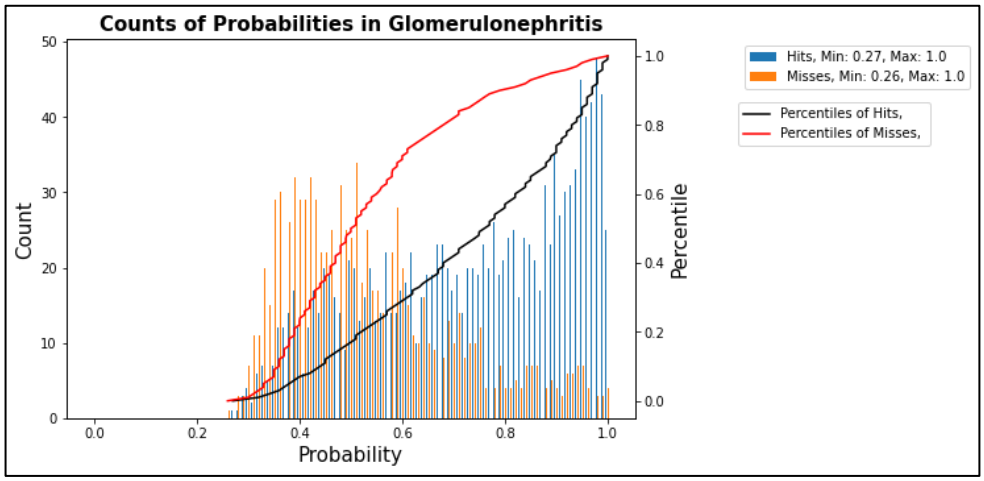

Figure 41. Count of hits and misses of glomerulonephritis by their probabilities and percentile showing how values fall below specific probabilities.

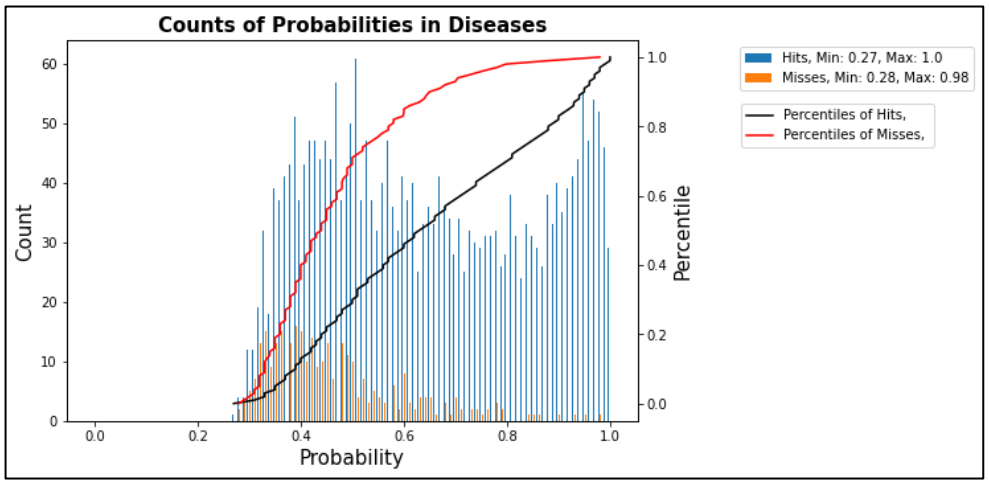

Figure 42. Count of hits and misses of diseases by their probabilities and percentile showing how values fall below specific probabilities.

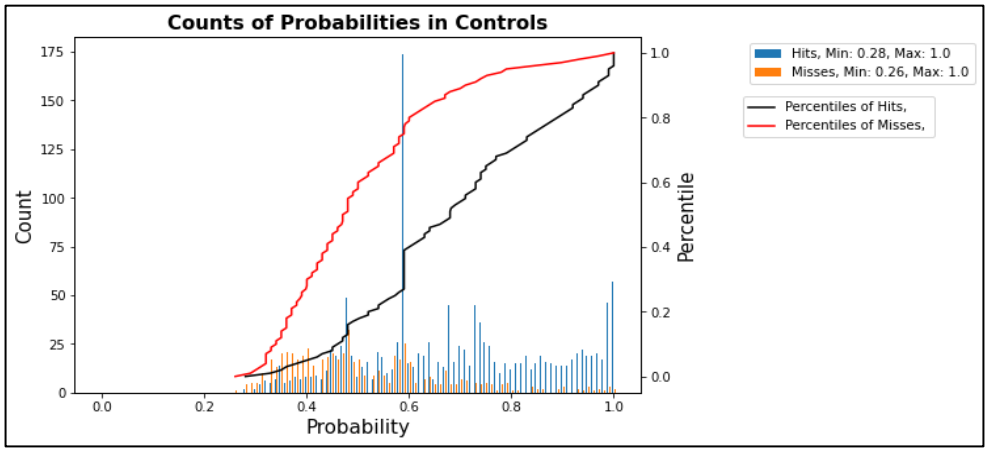

Figure 43. Count of hits and misses of control by their probabilities and percentile showing how values fall below specific probabilities.

Figures 39 to 41 present the individual disease's hits and misses, and Figure 42 presents hits and misses in a situation that a patient has some diseases, and the control Figure 43 describes the hits and misses for the classification of a healthy person. The hits and misses are in their dedicated probability place. The hits is the number of TPs and misses include all scenarios of false classification.

Figures 39 and 40 are showing similar results with the 30-day masking. There are lots of misses in the vasculitis and myositis classifications. The hits have a significant minority in the graph. In addition, in both cases the number of misses starts to decrease after the point of 60% probability. The number of hits stays about the same across the probabilities. Vasculitis has a maximum probability of hits in the 99% and misses in the 100%. Myositis has the maximum of hits in the 100% and misses in the 98%.

Figure 41 shows some improvement in comparison to Figure 17, where the mask was 30 days. The hits increase towards the end, and the misses decrease towards the end after an early rise. Both the hits and misses have a maximum of 100%, which means the model is certain of the results. In addition, it can be seen from the graph that after the probability of 60% there is more hits than misses in the individual probability frames, and before that, it is other way around.

Figure 42 presents misses and hits for combined diseases. It can be seen that hits have the majority of the graph area. The misses are decreasing after the increase in the start, and the decreasing begins to happen at around 50%. After 80% there are only a few misses. The maximum of hits is 100%, and misses has it at 98%.

The Controls' Figure 43 has a bit similar results as in the 30 days masking Figure 19. There are lots of hits close to the 60% probability, but the difference is that the number of hits increase towards the end maximum, which is 100%. The misses on the other hand decrease towards the end.

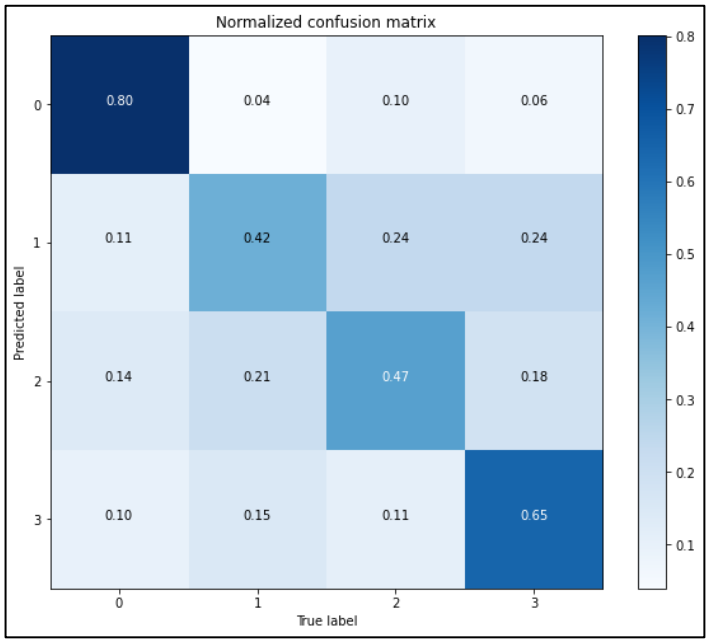

Figure 44. Confusion matrix of the predicted labels and true labels. Class zero is control, one is vasculitis, two is myositis and three is glomerulonephritis.

Figure 44 shows the confusion matrix of the predicted labels and true labels distribution. This helps to define what kind of errors the model made.

From the earlier 30 days masking Figure 20 we can see improvement in the control prediction and glomerulonephritis prediction. The control has the highest of accuracy of 80%, and the glomerulonephritis has the next highest accuracy of 65%. After that, myositis and vasculitis have some decrease, if the performance as myositis has the accuracy of 47% and vasculitis has the accuracy of 42%. In addition, it can be seen as a

small decrease of performance that there is a bit higher chance that a patient with a disease is classified as a control in comparison to the 30 days masking results. Despite these performance decreases, it can be seen that diseases are mostly mistaken to other diseases, not to control. The other way around the performance increased, so that control is more rarely mistaken to diseases.

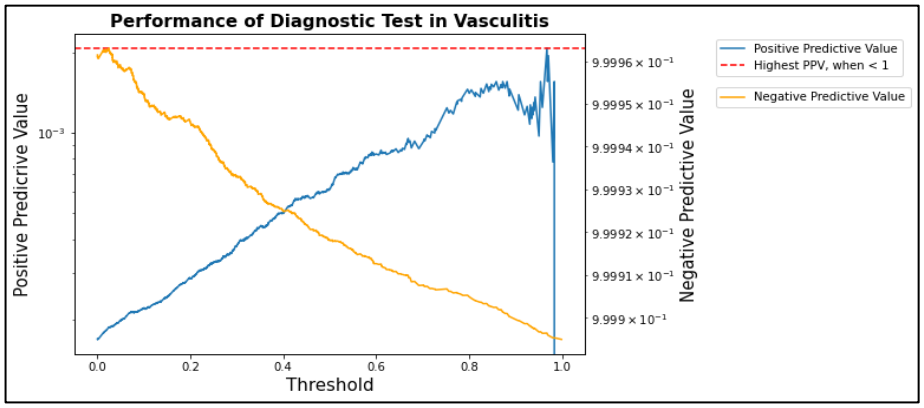

Figure 45. PPV and NPV for the vasculitis.

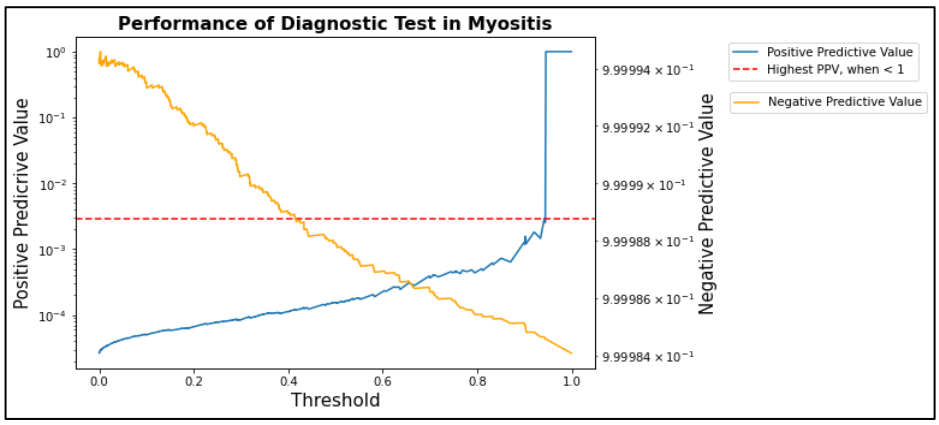

Figure 46. PPV and NPV for the myositis.

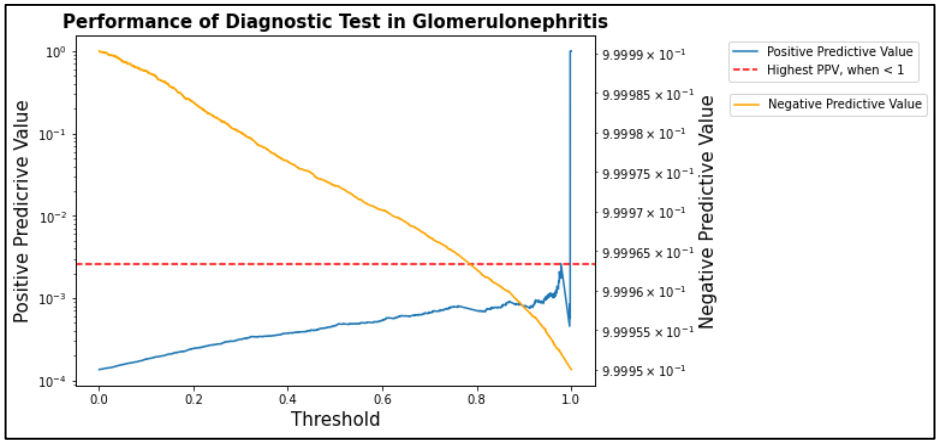

Figure 47. PPV and NPV for the glomerulonephritis.

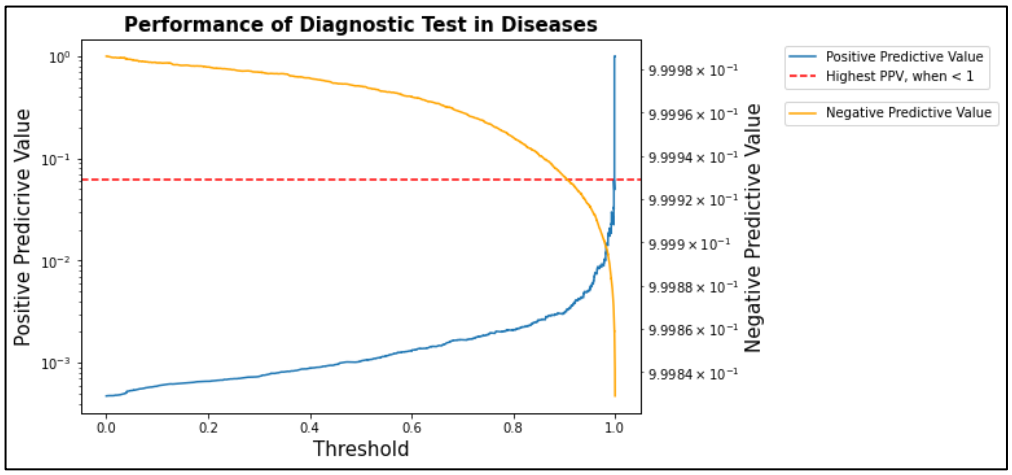

Figure 48. PPV and NPV for the diseases.

	PPV	NPV	THRESHOLD
VASCULITIS	0.2%	99.99%	97%
MYOSITIS	0.3%	99.998%	94%
GLOMERULONEPHRITIS	0.3%	99.995%	98%
DISEASES	6%	99.987%	99%

Table 5. Highest score of PPV which is below one and corresponding NPV and threshold value for that.

Figures 45 to 47 are showing the PPVs and NPVs with different thresholds, and Figure 48 shows same values but for combined diseases. The Table 5 shows the exact peak value of PPVs and corresponding NPVs and thresholds.

Figure 45 shows the results of vasculitis, and there the PPV peak of 0.2% happened in the threshold of 99%, and at the same time the NPV would be 99.99%. This would mean that for one TP there would be 499 FPs, and for every 99999 TNs there would be one FN.

The myositis results can be seen in Figure 46. The highest PPV peak of 0.3% happens in the threshold of 94%, and the NPV is 99.998%. The graph also shows that NPV is not decreasing rapidly. The PPV value shows that for three TPs there would be expected 997 FPs, or for one TP there would be about 332 FPs. Also, NPV value tells that for every 49999 TNs there would be about one FN.

Glomerulonephritis results in Figure 47 shows that PPV peak is 0.3%, and it happens in the threshold of 98%. At the same time the NPV is 99.995%. With these results we could expect for the same result as for myositis – for one TP there would be about 332 FPs –, but we would have a different result in the NPV as for every 19999 TNs there would be one FN.

Figure 48 shows the results of diseases when the highest peak of PPV 6% is reached, the threshold is 99%, and the NPV is 99.987%. These results help to assume that for every TP it is expected to have about 16 FPs, and for every 7691 TNs we could expect one FN.

4.3.2. XGBoost

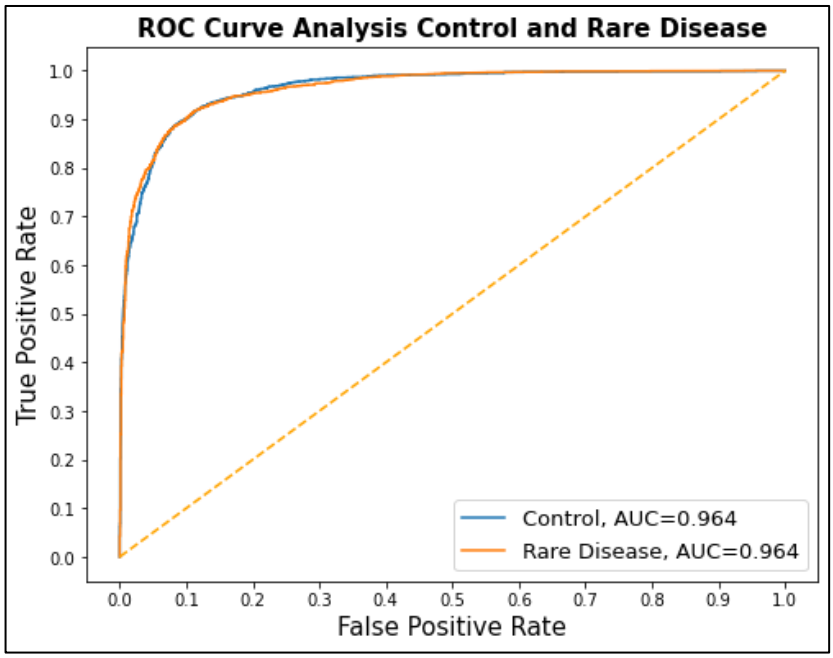

Figure 49. ROC curve of control and rare disease classifications with AUC value.

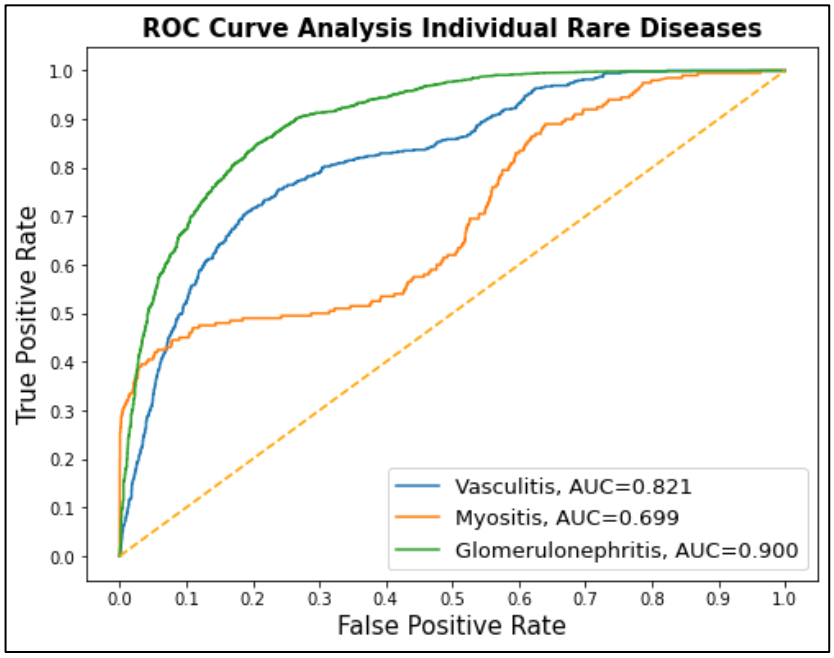

Figure 50. ROC curve of control and rare disease classifications with AUC value.

Figures 49 and 50 are showing the ROC curves with the AUC score of the classifications. The control and rare disease are in Figure 49, and the individual diseases are in Figure 50.

The control and rare disease Figure 49 has a great performance and identical AUC score of 96.4% as it was with the 30 days masking earlier. The optimal value from the graph would be about 90% of TPR, and then the FPR would be about 10%.

The individual disease Figure 50 shows some changes to the 30 days masking. Firstly, glomerulonephritis AUC score increases to 90%, and vasculitis' increase to 82.1%. Secondly, myositis has bigger dip and the AUC score decreases to 69.9%. These changes will change the optimal selected values from the graph. Glomerulonephritis' optimal value of TPR would be about 80%, and then the FPR would be about 15%. Vasculitis would have its optimal values of TPR of 70% and FPR of 20%. The optimal value of TPR myositis would have a bit over 40% and FPR of 5%.

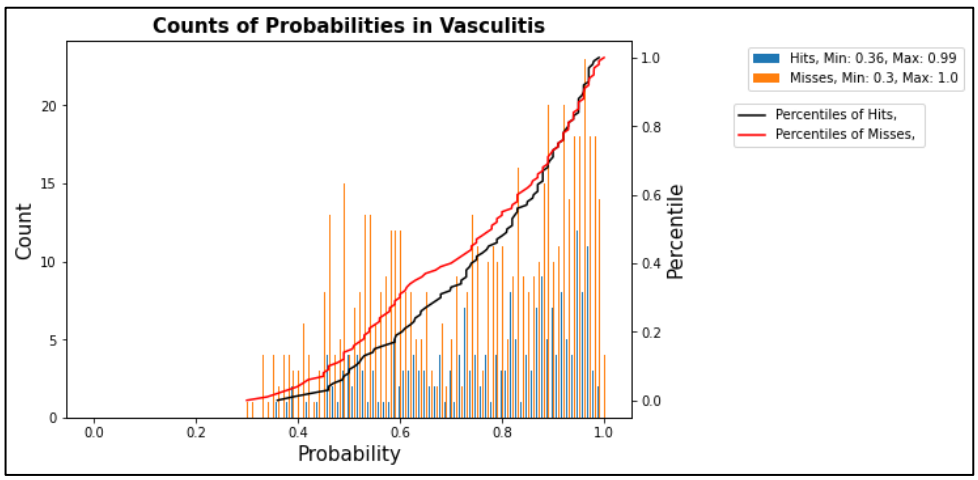

Figure 51. Hits and misses of vasculitis by their probabilities and percentile showing how values are distributed.

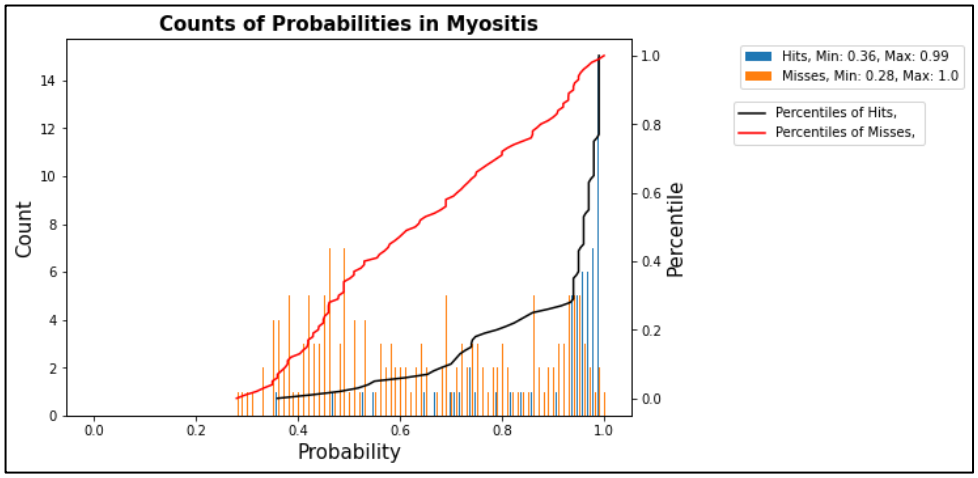

Figure 52. Count of hits and misses of myositis by their probabilities and percentile showing how values fall below specific probabilities.

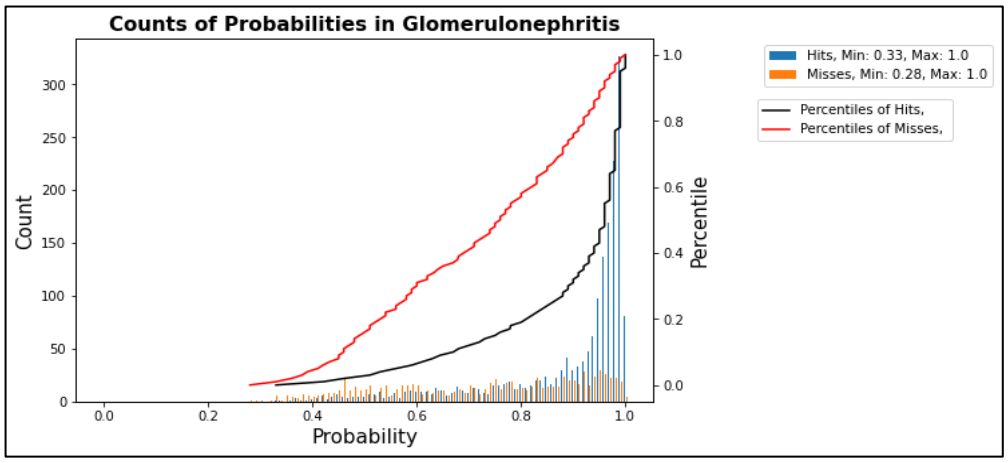

Figure 53. Count of hits and misses of glomerulonephritis by their probabilities and percentile showing how values fall below specific probabilities.

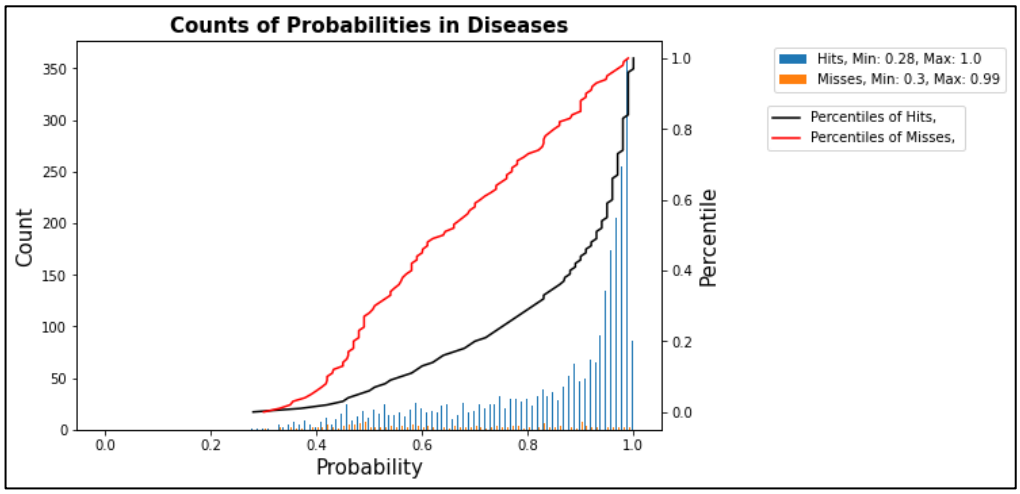

Figure 54. Count of hits and misses of diseases by their probabilities and percentile showing how values fall below specific probabilities.

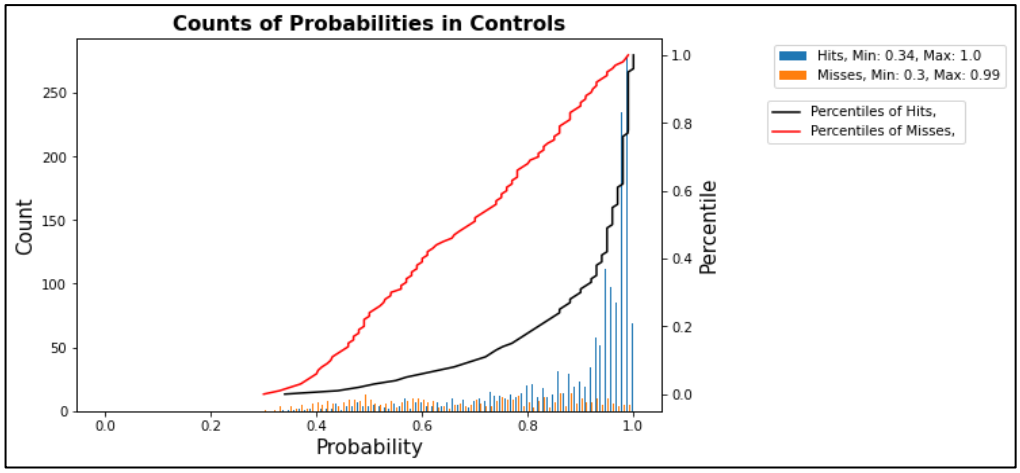

Figure 55. Count of hits and misses of control by their probabilities and percentile showing how values fall below specific probabilities.

Figures 51 to 53 are showing the misses and hits of the individual diseases. Figure 54 is showing the same results, but diseases are combined to one class. Figure 55 presents the results of control classification's hits and misses.

Figures 51 and 52 show similar results as with the 30 days masking, although the percentiles of hits slope increase steeper at the end in myositis. In this, both have a maximum of hits at 99% and misses at 100%55.

Glomerulonephritis in Figure 53 also shows similar results as they were in the 30 days masking. The maximum of hits and misses is equal to those results as well, as both have the values of 100%. The percentile curve of hits also steepens towards the end, and misses are quite linear.

The combined diseases have also steepened slope towards the maximum hits of 100% in Figure 54. The maximum of misses is at 99%. There are more hits than misses across the probabilities.

The controls in Figure 55 do not either show big differences in the slopes or in the maximum values. The maximum of hits increased 1%, from 99% to 100%, and misses from 98% to 99%. The slope of the percentiles of hits steepens gradually, but in the probability 90% it starts to increase faster.

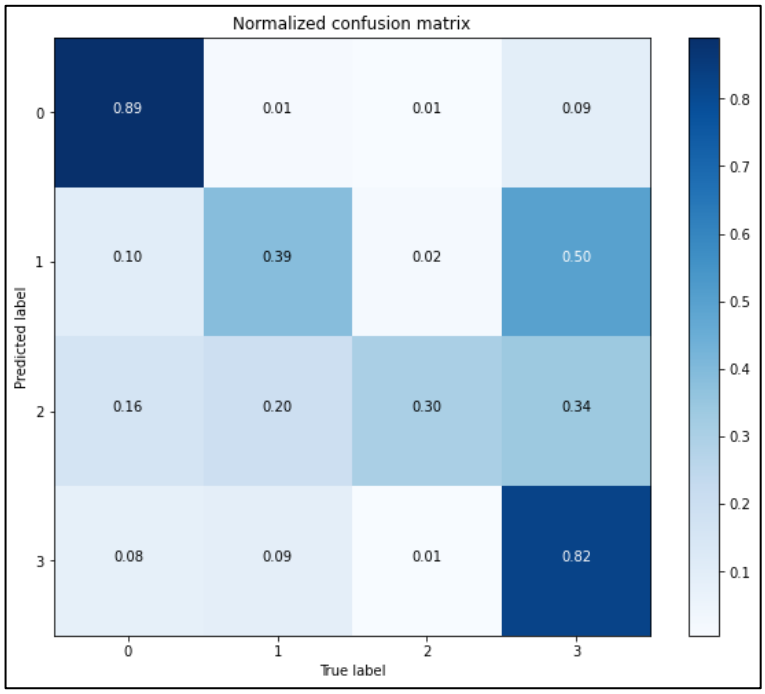

Figure 56. Confusion matrix of the predicted labels and true labels. Class zero is control, one is vasculitis, two is myositis and three is glomerulonephritis.

The confusion matrix Figure 56 describes the distribution of the correctly predicted labels and falsely predicted labels among the classes.

It can be seen that the results stay about the same, so there was no impact of the data reduction. Again, XGBoost predicts control and glomerulonephritis very well. Both have above 80% probability, as glomerulonephritis has 82% which is decreased by 1%, and the control equals the result of 89%. Then we have the same problem as with the 30 days masking, that the identification of vasculitis and myositis is not performing that well. Both are more probable to be miss-classified to glomerulonephritis. Vasculitis has that probability of 50%, which decreased 2% from earlier, and myositis 34% which also decreased 2%. Myositis' correct predictions stay the same 30%, and vasculitis has increased from 35% to 39%. It is notable that control is rarely predicted with diseases, and diseases are mostly mistaken with other diseases, although other classes are very rarely mistaken to myositis. Myositis itself is more probable to be mistaken with other classes.

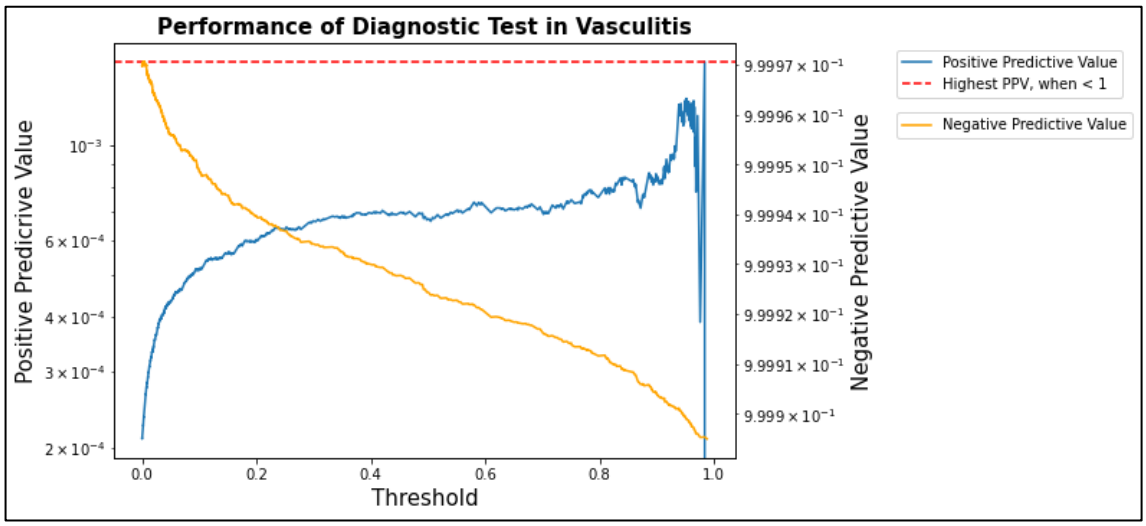

Figure 57. PPV and NPV for the vasculitis.

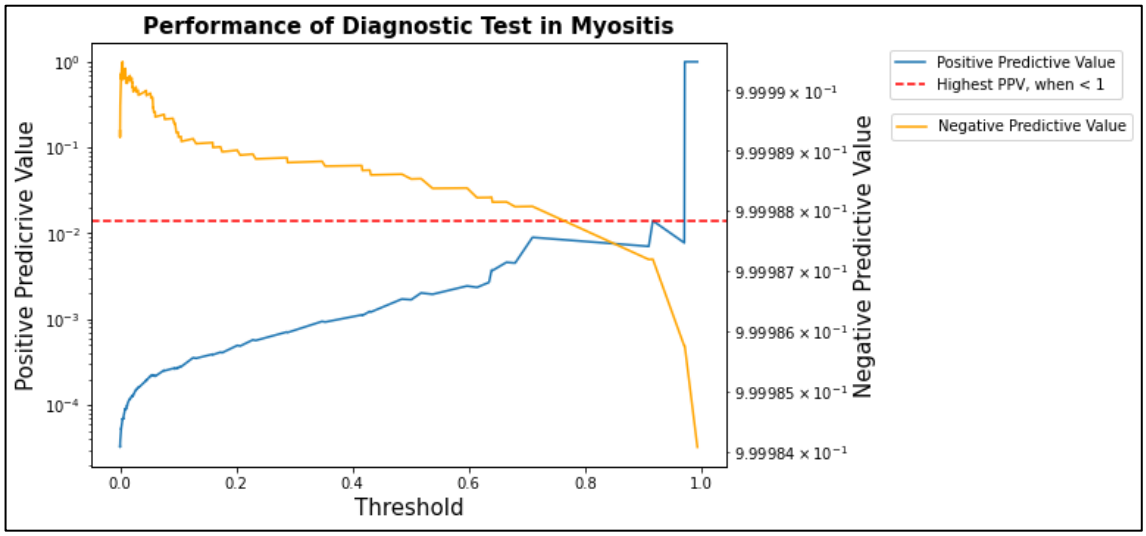

Figure 58. PPV and NPV for the myositis.

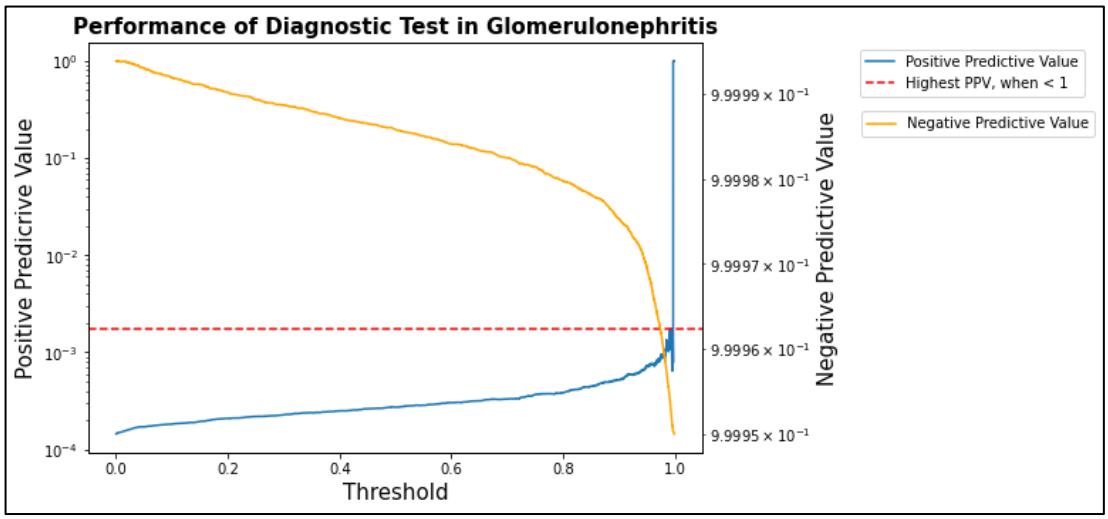

Figure 59. PPV and NPV for the glomerulonephritis.

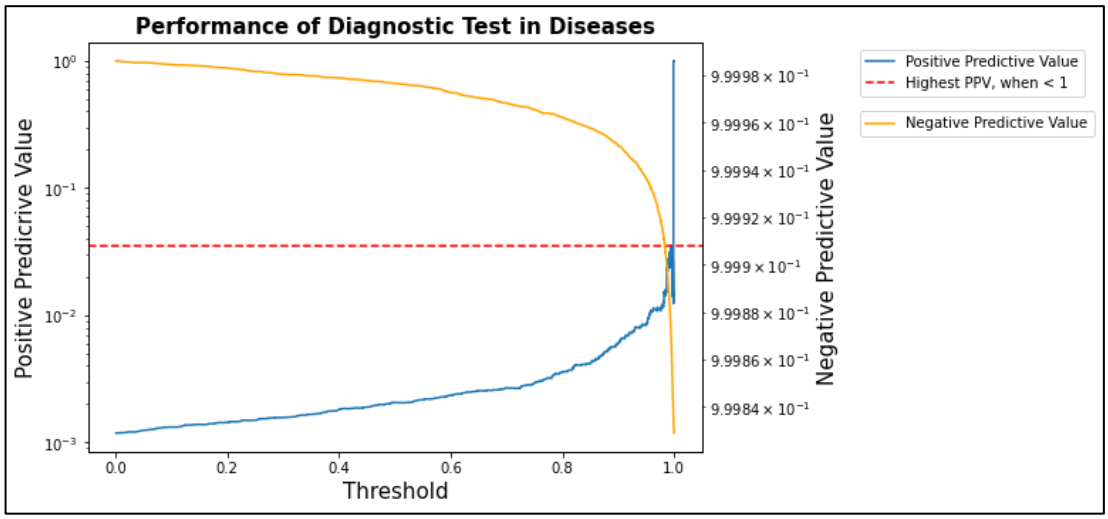

Figure 60. PPV and NPV for the diseases.

	PPV	NPV	THRESHOLD
VASCULITIS	0.2%	99.99%	98%
MYOSITIS	1%	99.999%	92%
GLOMERULONEPHRITIS	0.2%	99.995%	99%
DISEASES	4%	99.987%	99%

Table 6. Highest score of PPV which is below one and corresponding NPV and threshold value for that.

Figures from 57 to 60 present PPV and NPV results of individual diseases and combined diseases. The graphs have horizontal red line to mark the highest peak of the

PPV. There is also Table 6 which has the exact values of the highest peak value of PPV and corresponding NPV value in that threshold.

The vasculitis results are presented in Figure 57. NPV is not getting below 99%, and PPV peak of 0.2% is reached at the very end at the thresholds of 98%. NPV is in that same threshold 99.99%. These values indicate that in every TP there would be expected 499 FPs, and that for one FN it is expected to have 9999 TNs.

Figure 58 contains result for myositis, and in that the NPV really starts to decrease after the threshold of 70%. The decrease is not that significant, as the minimum of NPV is about 99.9984%. PPV peak of 1% is reached at the end at threshold of 92%. NPV in that same threshold is 99.999%. PPV value would expect that there will be one TP in every 99 FPs. NPV on the other hand expects that for on FN there is 99999 TNs.

In Figure 59 there are the results of glomerulonephritis. It possible to see that NPV starts to decrease faster after 80%. The peak value of PPV is at the 99% threshold, and the peak is 0.2% and corresponding NPV is 99.995%. These values tell that with optimal value it is expected to have one TP for every 499 FPs, and for one FN there is always expected to have 19999 TNs.

Figure 60 is combined diseases' graph, and there the NPV starts to decline faster after 90% of threshold. NPV value of 99.987% is reached at the peak of PPV, which is 4%, and the threshold in that situation is 99%. With these it is expected to have one FN against about 7691 TNs, and one TP with the expectation of 24 FPs.

4.4. 4320 days masking

In this last test there is 4320 days of masking used, which is about 12 years. That means that the decisions the models are making are done with reduction of initial diagnosis and 12 years of data before that breaking point.

4.4.1. My model

Figures 61 and 62 present the ROC curves and AUC scores of those curves. The control and rare disease results are in Figure 61 and the individual diseases in Figure 62.

Figure 61 shows improvement of the performance of this presentation from the earlier 30 days of masking and 2160 days masking. Both the control and rare disease have AUC score of 94.5% and therefore the theoretical optimal value from the graph would be about 85% of TPR, and then the FPR would be about 10%.

The individual disease Figure 62 indicates some improvement from the previous results of 30 days and 2160 days masking. The vasculitis AUC score of 76.3% equals the 2160 days masking result, and is better than the 30 days masking result. Myositis AUC is here 82.2% and it is an improvement from both of the earlier tests. The glomerulonephritis AUC of 87.1% equals the result of 2160 days masking, and it is an improvement from the

30 days masking result. The theoretical optimal values for glomerulonephritis would be TPR of about 80%, and FPR about 20%. Myositis would have a TPR of about 70% and FPR of 20%. Vasculitis' TPR could be about 65%, and FPR 30%.

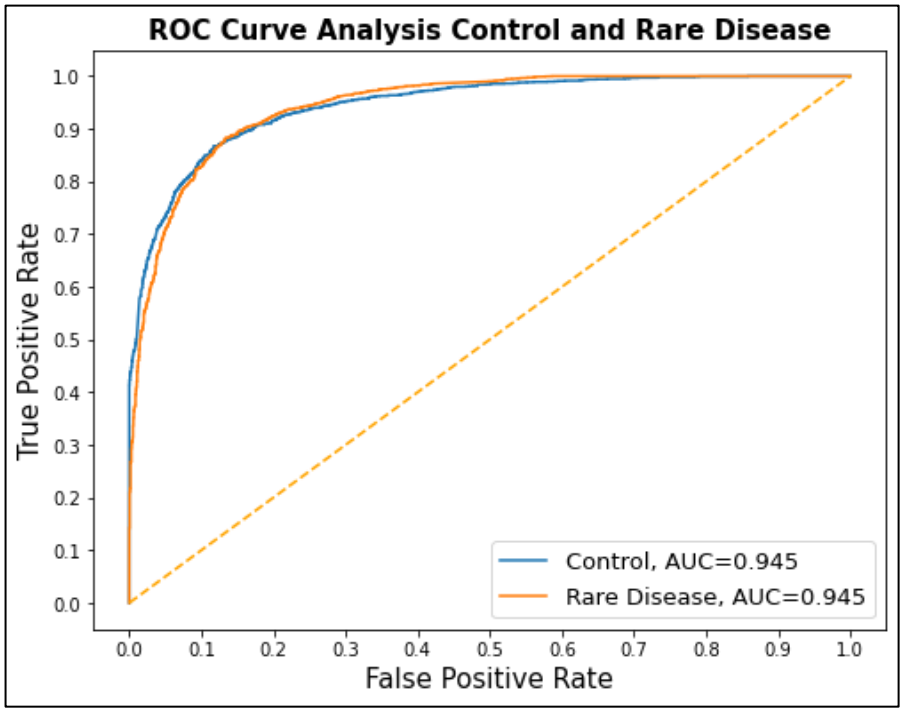

Figure 61. ROC curve of control and rare disease classifications with AUC value.

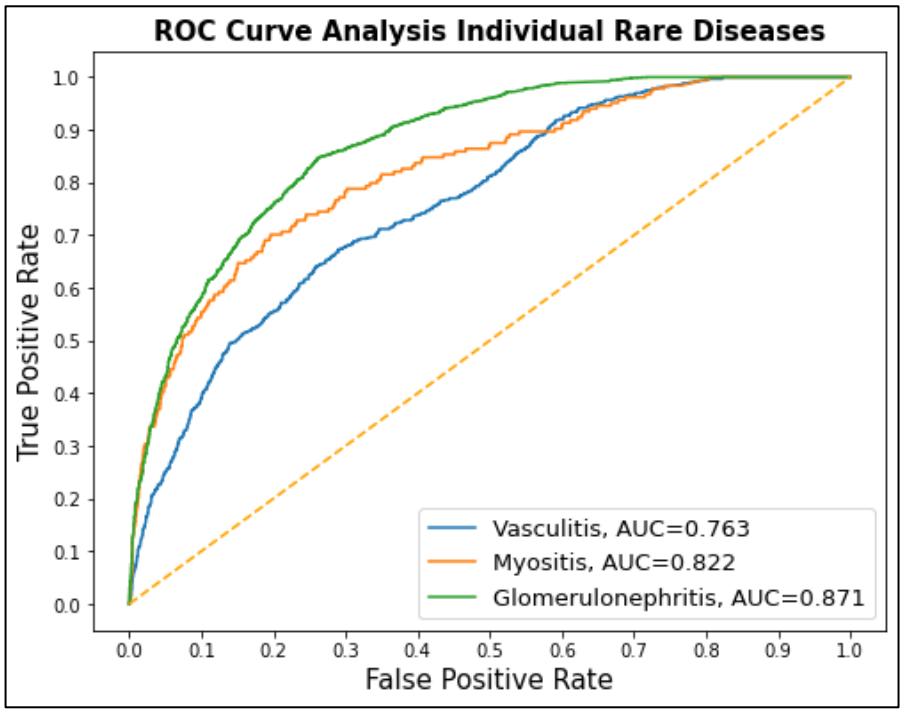

Figure 62. ROC curve of control and rare disease classifications with AUC value.

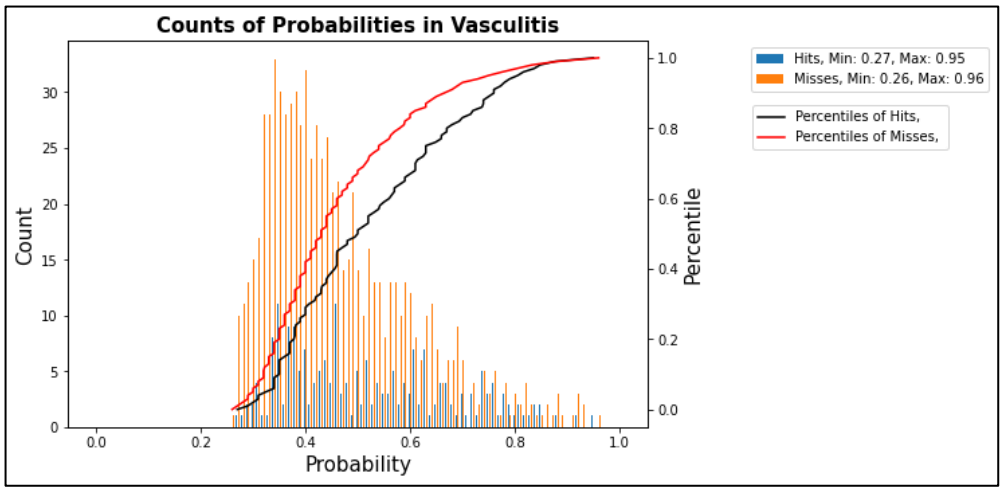

Figure 63. Hits and misses of vasculitis by their probabilities and percentile showing how values are distributed.

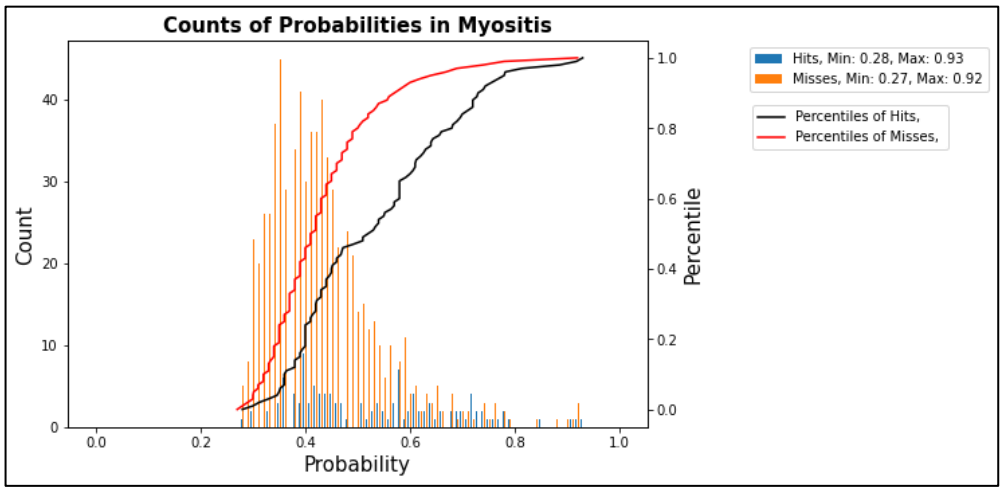

Figure 64. Count of hits and misses of myositis by their probabilities and percentile showing how values fall below specific probabilities.

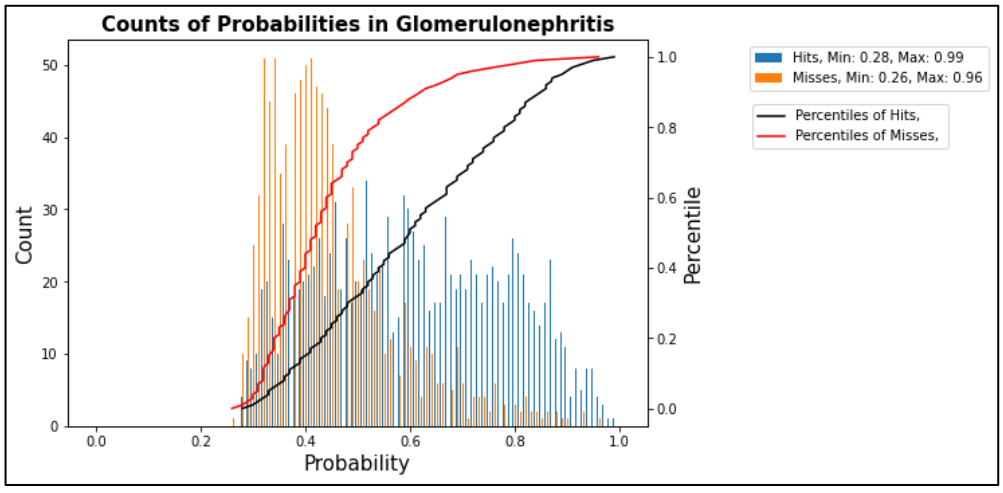

Figure 65. Count of hits and misses of glomerulonephritis by their probabilities and percentile showing how values fall below specific probabilities.

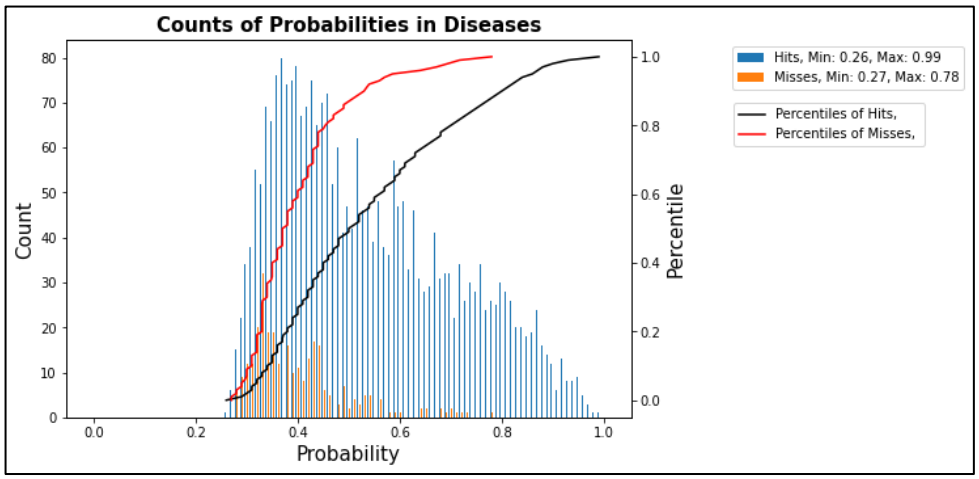

Figure 66. Count of hits and misses of diseases by their probabilities and percentile showing how values fall below specific probabilities.

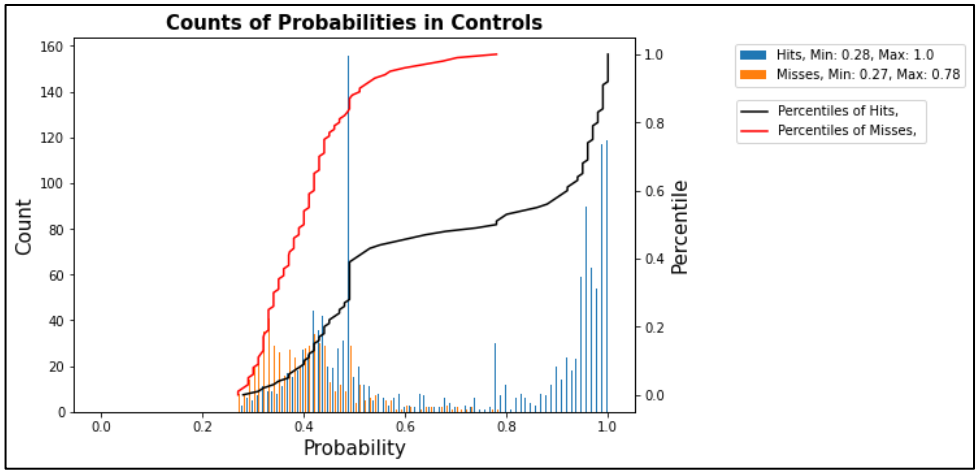

Figure 67. Count of hits and misses of control by their probabilities and percentile showing how values fall below specific probabilities.

Figures from 63 to 65 show misses and hits of individual diseases in specific probabilities and percentiles. Figures 66 and 67 show the same metrics, but for diseases and control.

The vasculitis' results in Figure 63 describe similar results than in the earlier tests. There are much more misses than hits. The maximum probability that is given for hits is 95%, and for misses it is 96%.

Figure 64 describes the results for myositis. It is also showing similar results than earlier, which means that there are more misses than hits. A notable difference is that the maximum probability has decreased significantly, and for hits it is now 93% and for misses it is 92%.

Figure 65 shows the results for glomerulonephritis. The hits are quite equally distributed across the probabilities, and the maximum is in the 99%. The graph

distribution is more similar with the 30 days masking results than with the 2160 days masking, as the hits are not increasing towards the highest maximum probability. The maximum value of misses is 96%, and the number of misses is dropping fast in around 50% probability.

Figure 66 shows the results for the combined diseases. This describes quite the different results, as the maximum values have a big gap between them. For hits the score is 99%, and for misses it is 78%, the difference being 21%. Distribution is closer to the 30 days masking results than the 2160 days masking.

The Control Figure 67 shows very different results than the earlier results. The certainty of the classification increases very fast after 80% probability, which has not been common for this model. The maximum values also have a difference, in hits it is 100% and in misses it is 78%, which means that the difference is 22%.

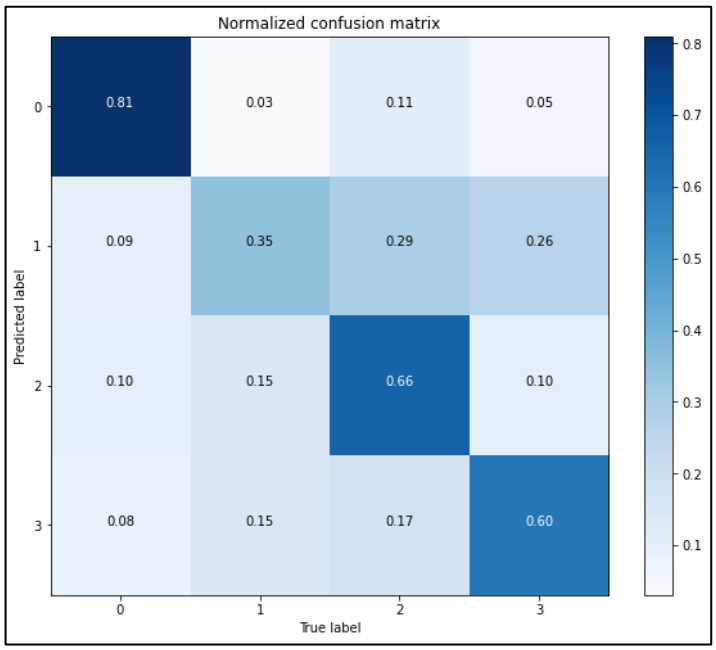

Figure 68. Confusion matrix of the predicted labels and true labels. Class zero is control, one is vasculitis, two is myositis and three is glomerulonephritis.

Figure 68 is a confusion matrix of the predicted labels and true labels, and this helps to see how hits and misses are distributed with classes.

It can be seen that the control predictions are improved from the previous masking test, and it is now 81%. There is a great improvement with myositis with the accuracy of 66%. Glomerulonephritis' performance decreased a bit in comparison to the 2160 days masking results, but still it was better than the 30 days masking results, as the accuracy is now 60%. Vasculitis' performance has decreased constantly every time the masking has increased. Now the accuracy is 35%. Despite the decrease of vasculitis' performance, it is still more likely to be classified with other diseases than control, and most likely it is

classified correctly. The same thing goes with all of the diseases – if they are miss classified, they are more likely classified with other disease. There is also a small difference in which diseases are a bit more often miss classified as control in comparison to the other way around.

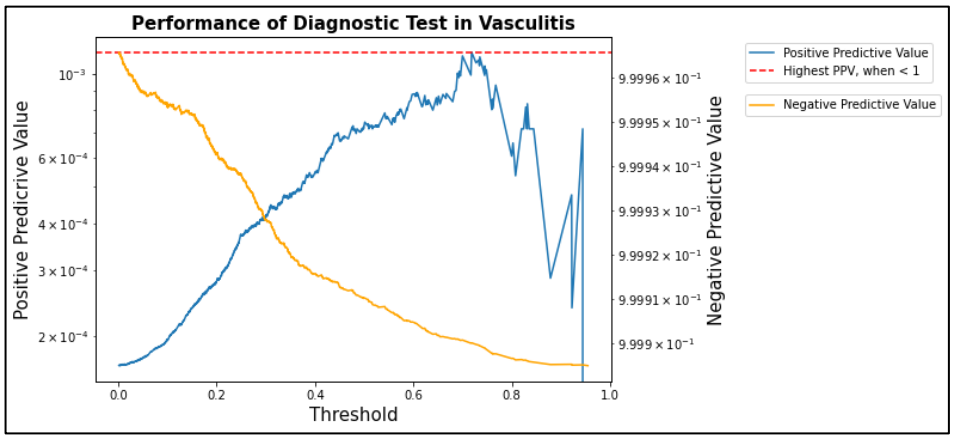

Figure 69. PPV and NPV for the vasculitis.

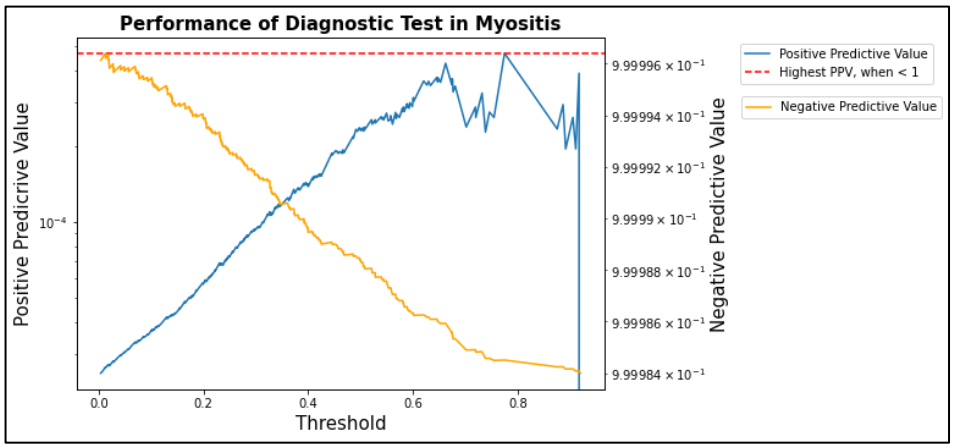

Figure 70. PPV and NPV for the myositis.

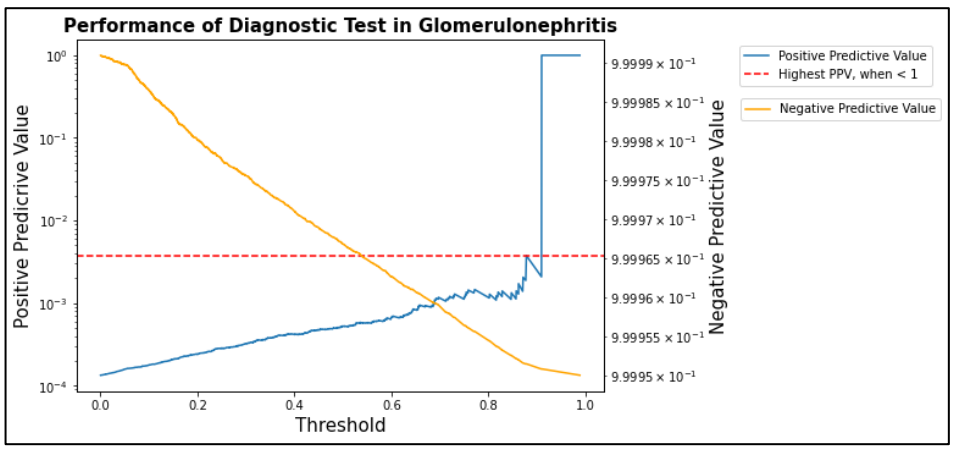

Figure 71. PPV and NPV for the glomerulonephritis.

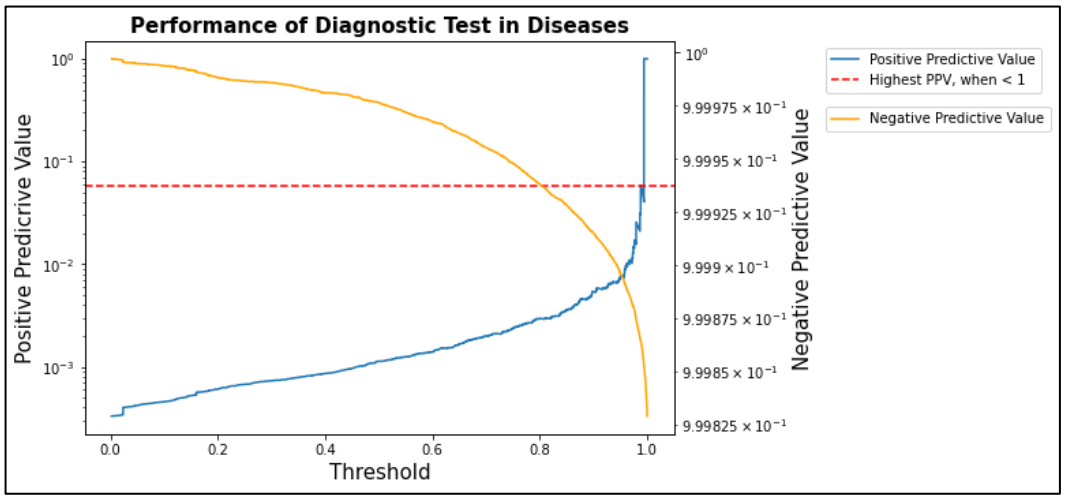

Figure 72. PPV and NPV for the diseases.

	PPV	NPV	THRESHOLD
VASCULITIS	0.1%	99.99%	72%
MYOSITIS	0.05%	99.998%	77%
GLOMERULONEPHRITIS	0.4%	99.995%	88%
DISEASES	6%	99.986%	99%

Table 7. Highest score of PPV which is below one and corresponding NPV and threshold value for that.

Figures 69 till 72 present the PPVs and NPVs in different thresholds, and the red horizontal line indicates the peak values of PPV. The Table 7 shows the specific peak values of the PPV, as well as the thresholds of that position and the corresponding NPV value.

The results of vasculitis are shown in Figure 69, and it shows that the NPV decreases steadily, but it still does not get below 99%. PPV's peak is in the threshold 72% with values of 0.1%, which means this model would have one TP for every 999 FPs. Also, NPV in that same threshold is 99.99%, which indicates that there would be one FN for every 9999 TNs.

Figure 70 shows similar curve with vasculitis, as NPV gradually decreases, but does not get below 99%. Peak of the PPV happens at the threshold of 77%. PPV is there 0.05% and NPV is 99.998%. With this it would be expected to have one TP in every 1.999 FPs, and one FN for every 49999 TNs.

Glomerulonephritis results can be seen in Figure 71. It shows a similar gradually decreasing NPV as it did with the vasculitis and myositis. The peak of PPV is 0.4%, and threshold in that point of graph is 88%. The NPV value in that threshold is 99.995%.

Based on the PPV, it would be expected to have one TP in every 249 FPs, and with NPV it is expected to have one FN in every 19999 TNs.

In the combined diseases in Figure 72 the NPV curve is very different as it decreases slowly at the start till 70%, and then it starts to decrease faster. Still, it stays above 99% and closer to 100%. The PPV peak of 6% and NPV of 99.986% are in the point of threshold 99%. The NPV would expect that one FN would happen in every 7142 TNs, and with PPV it is expected to have 16 FPs for each TP.

4.4.2. XGBoost

The ROC curve with the AUC shows the ratio of the TPR and FPR. AUC is how much space is under the curve. Figure 73 shows the control's and disease's results, and in Figure 74 it describes the individual disease results.

Figure 73 has similar results there was with the 30 days masking and the 2160 days masking. Both control and rare disease has the value of 96.3%, which is 0.1% lower than earlier. The optimal value here would be TPR of about 85% and then FPR of about 7%.

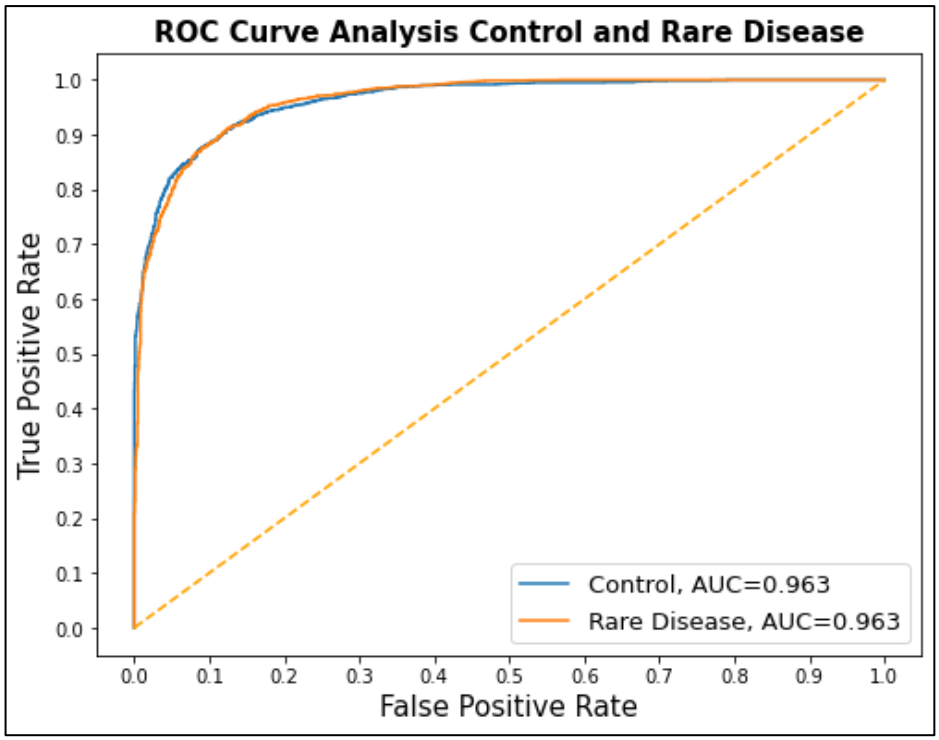

Figure 73. ROC curve of control and rare disease classifications with AUC value.

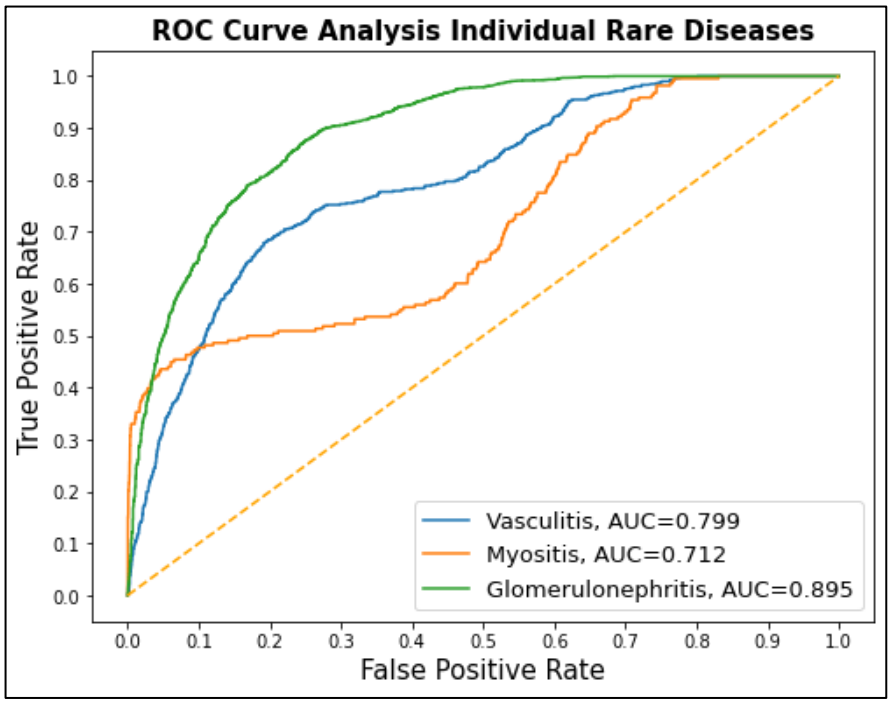

Figure 74. ROC curve of control and rare disease classifications with AUC value.

From Figure 74 it can be seen that myositis and vasculitis have dips on their curve. The dip of the myositis is about the same as it was in the 2160 masking results in Figure 50, and for vasculitis it has grown a bit in comparison to the same results. Myositis value of AUC has increased a bit from Figure 50 results to 71.2%, and the vasculitis result of 79.9% decreased, and glomerulonephritis results stayed on about the same level with 89.5%, which decreased only 0.5%. The theoretically optimal values for glomerulonephritis would have TPR of about 80% and FPR of 15%. For vasculitis the value of TPR would be about 70%, and FPR about 20%, and for Myositis TPR of 45%, and FPR would then be about 10%.

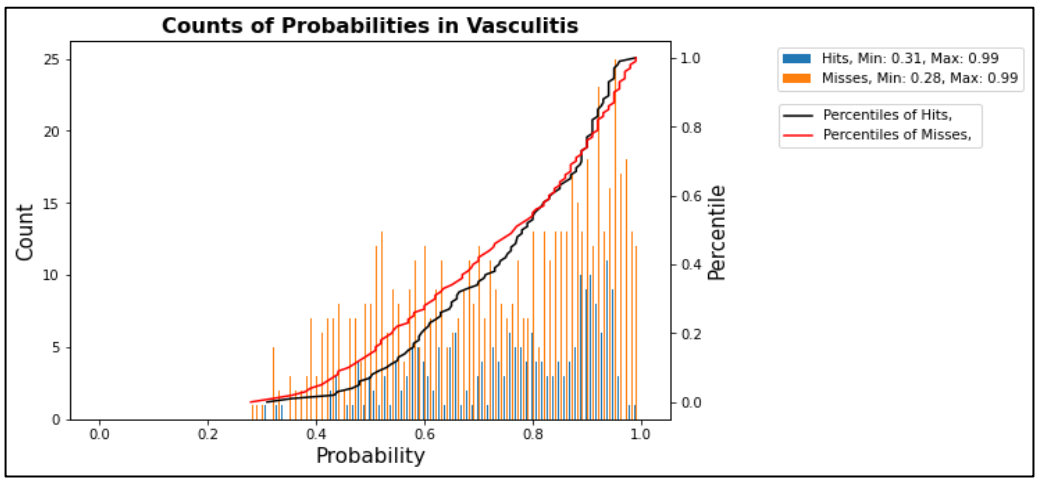

Figure 75. Hits and misses of vasculitis by their probabilities and percentile showing how values are distributed.

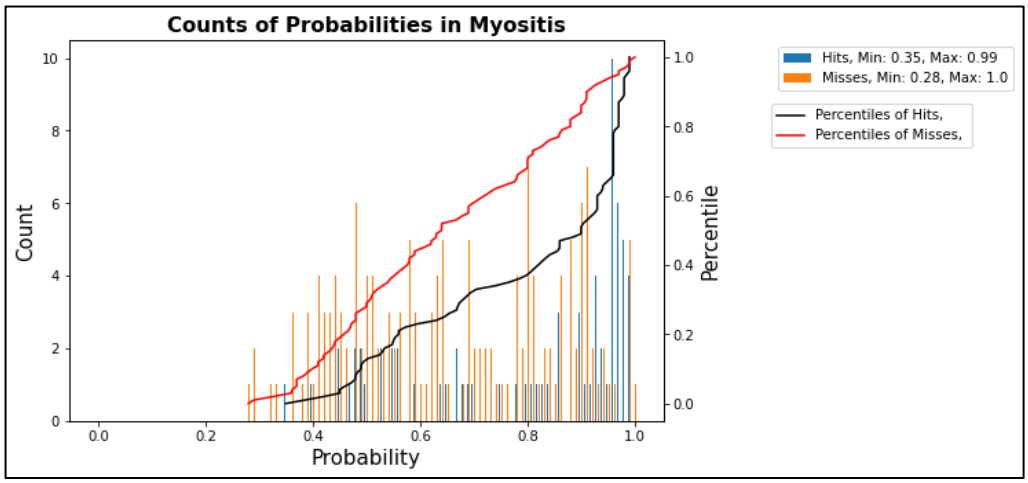

Figure 76. Count of hits and misses of myositis by their probabilities and percentile showing how values fall below specific probabilities.

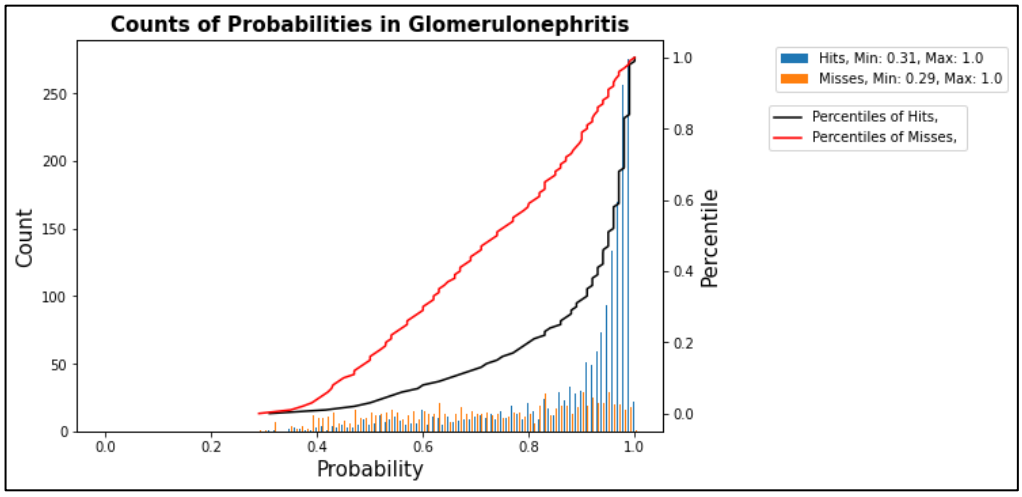

Figure 77. Count of hits and misses of glomerulonephritis by their probabilities and percentile showing how values fall below specific probabilities.

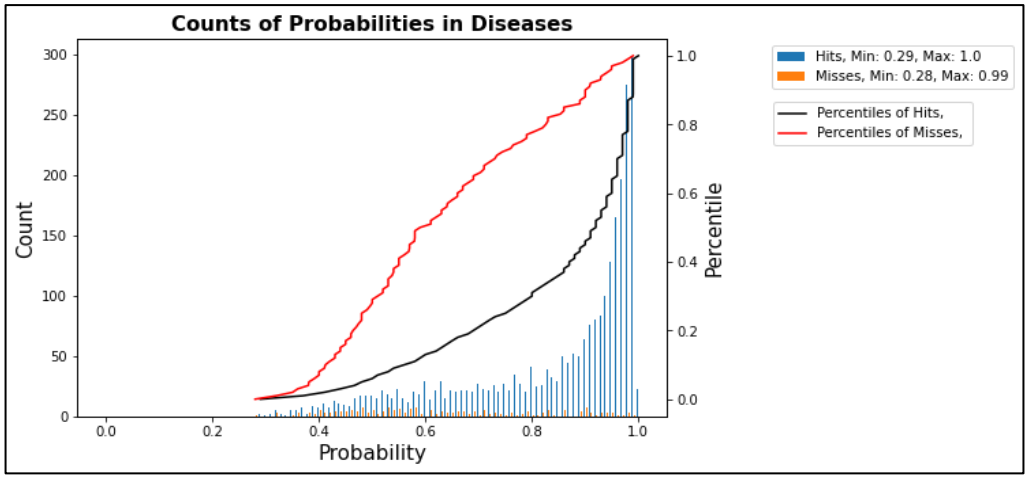

Figure 78. Count of hits and misses of diseases by their probabilities and percentile showing how values fall below specific probabilities.

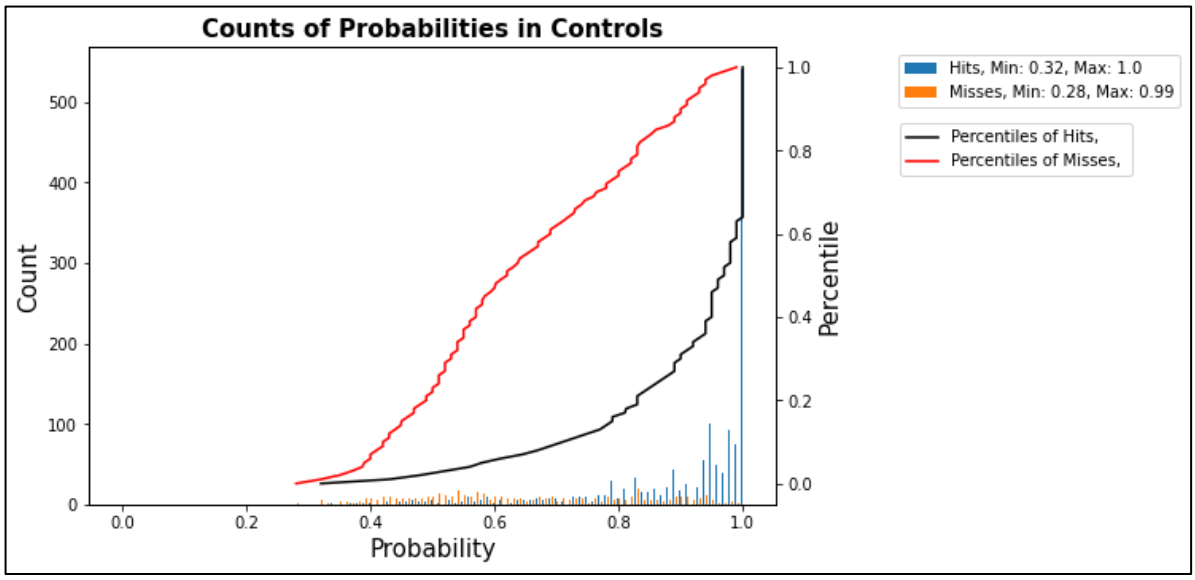

Figure 79. Count of hits and misses of control by their probabilities and percentile showing how values fall below specific probabilities.

Figures from 75 to 79 shows the count of misses and hits and their minimum- and maximum probabilities. In addition, the percentiles are shown in the graph.

Figure 75 shows the results of hits and misses for vasculitis. The results are about the same as in the earlier tests, so no visible improvement or decline can be seen overall in the different masking tests. The maximum probability of hits and misses is for both hits and misses 99%.

Figure 76 shows the results of myositis, and it has also similar results as it did in the 30 days masking test. The results differ a bit from the 2160 days masking results as it has a lower peak of hits at the end. Still, there are more misses than hits. The maximum probabilities of hits is 99%, and misses it is 100%.

The glomerulonephritis' results in Figure 77 show similar results as earlier. There the misses are distributed quite equally across the probabilities, and the amount of hits start to increase from 80% towards the maximum probability of 100%. The misses' maximum value is also 100%.

The combined diseases results are presented in Figure 78. It shows quite equal results with the 30- and 2160-days masking results. No big differences are visible, but a small difference number of hits in the peak bar is about 50 hits lower than earlier. The maximum probabilities for misses is 99%, and for hits it is 100%. Also, as earlier, there are much more hits than misses.

The control's results in Figure 79 show improvement from earlier. There the highest peak of hits has increased about 250 hits. Also, the peak is in the maximum probability of 100%. The maximum probability for misses is 99%. The misses are pretty evenly distributed across the probabilities, as it can be seen in the percentile line.

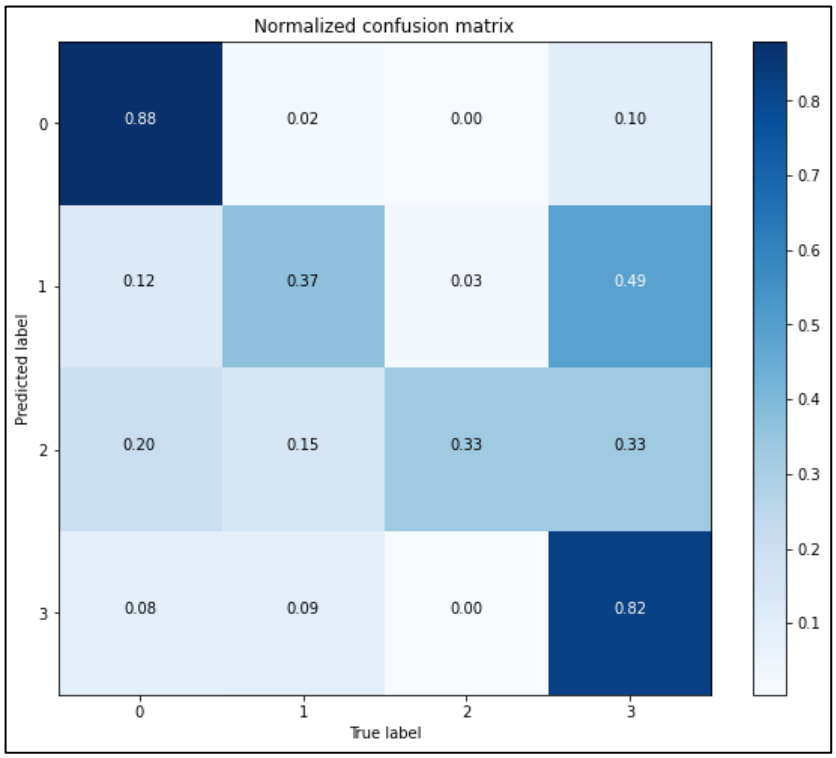

Figure 80. Confusion matrix of the predicted labels and true labels. Class zero is control, one is vasculitis, two is myositis and three is glomerulonephritis.

The normalized confusion matrix of the predicted labels and true labels can be seen in Figure 80. With this it is easier to see classification obstacles and successes with different classes.

The results here indicate that the control and glomerulonephritis classification works very well, as the control has the accuracy of 88% and glomerulonephritis has the accuracy of 82%. Then we have vasculitis and myositis, which have issues with that. Vasculitis has a correct prediction accuracy of 37%, and myositis has the accuracy of 33%. Vasculitis has miss classification to the glomerulonephritis with the accuracy of 49%, which is 12% higher than the correct prediction. Myositis has miss classification accuracy of 33% to glomerulonephritis, which equals the correct prediction value. There is neither significant increase- or decline of accuracy, when Figure 80 results are compared to the earlier results with different size of masking. In addition, the results present the same as earlier tests – that control is rarely miss-classified to the disease, and the other way around it is more common with the exception of glomerulonephritis, which is almost the same. From the matrix it is also easy to see that the other classes are rarely or never miss-classified to myositis, and the other way around myositis has the most distributed classifications. In addition to these, it is more common for diseases to be miss classified to other diseases than control.

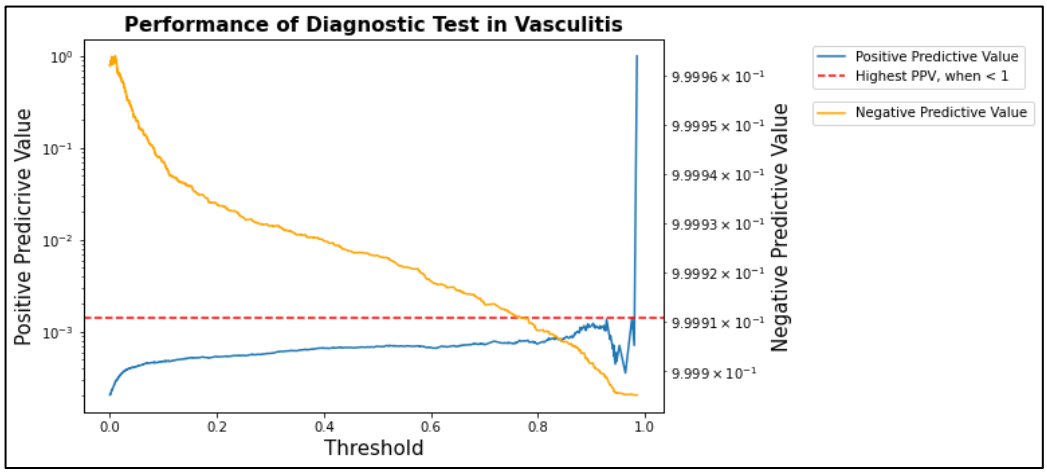

Figure 81. PPV and NPV for the vasculitis.

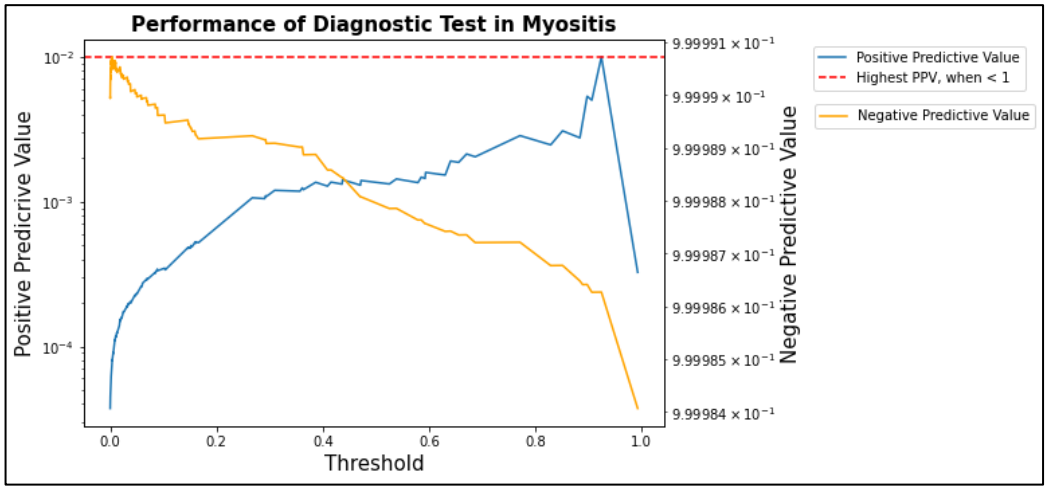

Figure 82. PPV and NPV for the myositis.

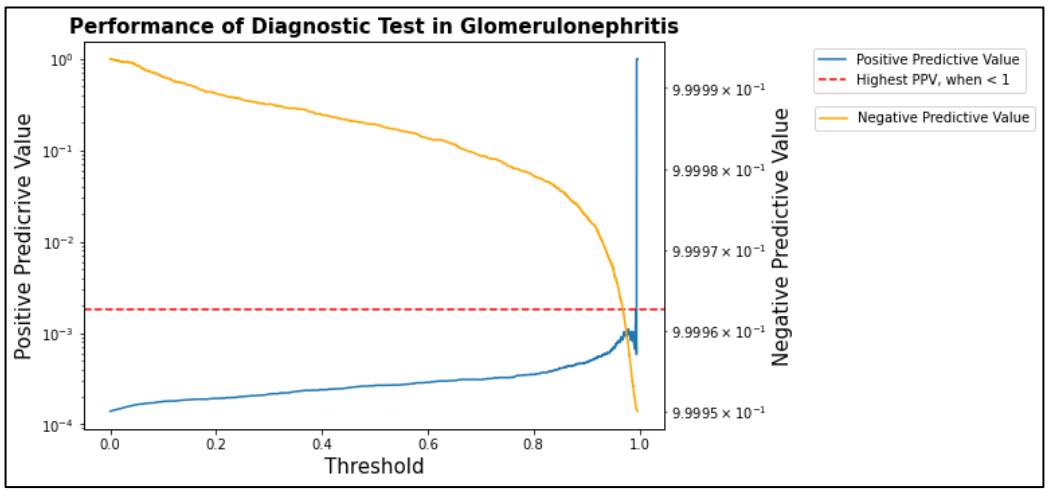

Figure 83. PPV and NPV for the glomerulonephritis.

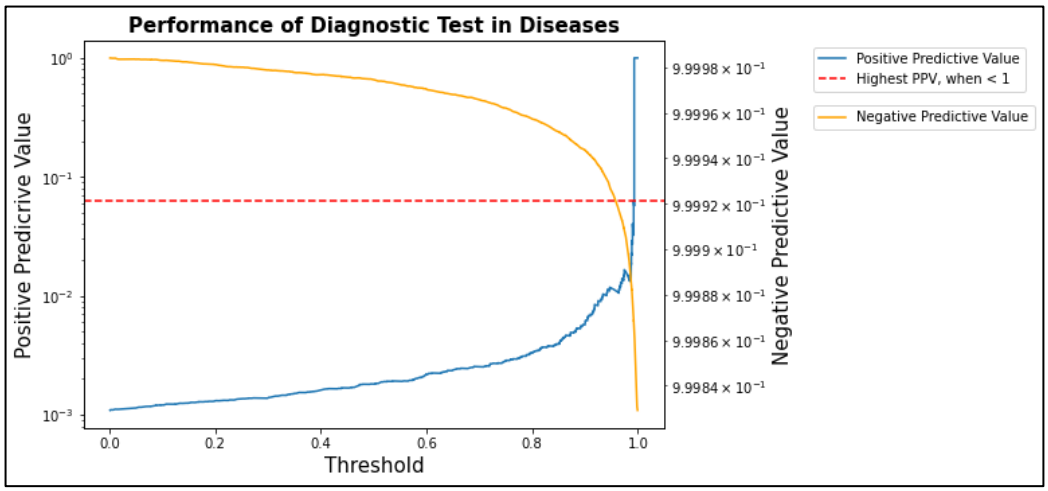

Figure 84. PPV and NPV for the diseases.

	PPV	NPV	THRESHOLD
VASCULITIS	0.1%	99.99%	98%
MYOSITIS	1%	99.999%	92%
GLOMERULONEPHRITIS	0.2%	99.995%	99%
DISEASES	6%	99.987%	99%

Table 8. Highest score of PPV which is below one and corresponding NPV and threshold value for that.

Figures from 81 till 84 present the curves of the PPVs and NPVs, and the horizontal line for the PPV's highest peak. There is also Table 8 where the exact peaks of PPVs and corresponding NPVs are in the same thresholds.

The results of vasculitis are in Figure 81. From Figure 81 we can see that the highest peak of PPV has declined in comparison to the earlier tests with different masking sizes, but no notable differences have occurred in the NPVs. The exact peak of PPV is 0.1% in the threshold of 98%. The NPV is 99.99%. With these values it is expected to have one TP in every 999 FPs, and one FN in every 9999 TNs.

The myositis results can be seen in Figure 82. There are no notable changes in these results to the previous results, as those have been about the same. The peak of PPV happens in the threshold of 92%, and the PPV reaches 1% and at the same time the NPV is 99.999%. Based on PPV, it is expected to have one TP for every 99 FPs and one FN in every 99999 TNs.

The glomerulonephritis results in Figure 83 describe similar results as in the 2160 days masking test, which was a decline from the 30 days masking test. The peak of PPV is reached in the 99% threshold with the value of 0.2%, and NPV in that threshold point

is 99.995%. NPV indicates that expected value of TNs is 19999 for one FN, and for PPV that one TP is expect for every 499 FPs.

From diseases' Figure 84 results show improvement from the 2160 days masking test, but it is still worse than the 30 days masking test. The PPV peak of 6% and NPV in the same threshold of 99% is 99.987%. With these results it is expected to have one TP in every 16 FPs, and one TN in every 7691 FNs.

5. Discussion

The purpose of this work was to show that early identification of rare diseases is possible by using residual neural network and compare received results to the results obtained by random forest classifier.

The results obtained with residual neural network model show that it is more than capable in the early identification of rare diseases. The models performed well above expectations across the different masking tests. The best result was received with the 2160 days masking when PPV was inspected. It won three out of four peak PPVs against the other masking tests of this model. Based on this, an assumption can be made that this model could find indications of rare diseases about six years earlier than initial diagnose was given. If just the indication patient was having some diseases, the 4320 days masking model received equal PPV result, and it could be argued that it is better, as identification is made 12 years earlier than the initial diagnosis. There are still differences in how individual diseases performed. Glomerulonephritis detection was the best with the 30 days masking, which could indicate that it is more aggressive disease compared to the two other diseases. Vasculitis and myositis had their best results with 2160 days masking. In addition, it was shown that if diseases are miss-classified, they are more probably miss-classified to other rare disease than to control, which means that rare diseases have some similarities which makes it harder to specify which rare disease is in question. With these results it could be proposed that the residual neural network's performance does not decrease rapidly when increasing the size of the mask.

Compared to the XGBoost model, the results are very promising, as it was shown that the best results of the XGBoost was received in the 30 days masking test, and with increased size of mask the performance declined. Although when comparing the best results of the XGBoost and residual neural network, the clear winner would have been XGBoost. Yet those results cannot be compared, because the differences of masking size. The residual neural network had 2130 days bigger mask than the XGBoost, which is about five years and 11 months. If those 2160 days masking results were compared, the residual neural network would have won based on performance. With the biggest mask these models tied with categories based on performance, where both won one and tied two. Then from the hits and misses it is clearly visible that the XGBoost's certainty of decision-making increases towards the highest probabilities. The residual neural network on the other hand has hits either distributed quite steadily across the probabilities or it is declining towards the highest probabilities, and number of misses are declining towards the higher probabilities. There is an exception in the residual neural network's control results, as the certainty increases towards the highest probabilities. It is notable that the residual neural network does not have miss-classifications after the threshold probability of 78% when inspecting the binary classification that either patient is healthy or has a

disease. However, XGBoost is still having misses in threshold probability of 99%. Hence it could be argued that residual neural network performs better also with the highest mask, despite the draw in the categorical PPV inspection.

In population level PPVs for both Inception and XGBoost models are excellent for rare diseases. Inception model's PPV probability for vasculitis was 0.2%, and myositis and glomerulonephritis had 0.3%. Probability of patient having any one of these diseases was 6%. XGBoost's PPV probability for vasculitis was 0.6%, for myositis it was 1% and for glomerulonephritis it was 0.5%. For patient having any one of these diseases was the probability 8.8%. Random selection would produce estimated probabilities for vasculitis 0.005%, for myositis it is 0,0105% and for glomerulonephritis it is 0.0016%. When random selection probabilities are compared to the models results, random selection really cannot compete with ML models where the probabilities are at least 10 times better or even more than 100 times better in some cases.

Compared to the Cheng *et al.* [2016] research's state-of-the-art models' results where risk prediction of *congestive heart failure* (CHF) and *chronic obstructive pulmonary disease* (COPD) were made, residual neural network worked better in its problem. For data they used electronic health records, as was also used in this thesis. From the various models used in their research, *SF-CNN* (single-frame convolution neural network) performed the best with both diseases. The top AUC scores they received with 90% ratio of training data, and CHF received about 76.75%, and COPD about 73.88%. [Cheng *et al.* 2016] Residual neural network bottom AUC score was of individual disease 72.6%, and the top score of 84.7%. The bottom score was obtained with the worst performing model, and the second bottom AUC score is 76.3%, which was received with the 2160 days masking and 4320 days masking tests. This shows that the residual neural network performs well beyond the SF-CNN. In addition to that, my tests used multiclass softmax layer in classification where SF-CNN used softmax as well, but classes were binary "in risk" and "no risk" [Cheng *et al.* 2016]. Jia *et al.* [2018] researched rare diseases detection with few deep neural network classifiers with *CPML* and *APML* models. Their results include PPVs and TPRs, but in PPV does not take prevalence into account, so comparison is hard to make. They reached TPR of 30% for their best disease with APML, and with CPML 39%. For top 10 diseases APML TPR is 82% and CPML has 95%. [Jia *et al.* 2018] Compared to residual neural network, the best results were given by glomerulonephritis with 2160 days masking and theoretical optimal value of TPR of 75%, which is well above their results. If an assumption is made that top 10 results are combined results of 10 diseases, some comparison can be made with the combined diseases TPR, which includes three diseases. The optimal value from the ROC curve in Figure 37 would be about 85%, which is more than APML model's result, but residual neural network would lose about 10% to the CPML model's TPR. Also, it could be argued that if their PPV

results would take prevalence into account, the PPVs would be lower. With these results it can be said that the residual neural network results for individual rare diseases detection is better, and for combined diseases it is competitive.

I succeeded in implementation of the inception type model in a new way that is proven competitive against the other state-of-the-art models, as well as solve the classification problem with great results. In addition to that, I managed to surpass few issues including slow training time of the models, class imbalance, and overfitting. Solutions increased performance, and decreased the time it took to train the model.

The weakness of the work includes insufficient medical knowledge of the diseases and patterns found from the them. Furthermore, if more data was available, it could have increased the performance, as the minimum number of patient data in one disease group was below 1000, and the others had at least three times more.

As early identification of rare disease with residual neural networks, or more specifically, with inception type models, are quite unexplored, this work should be continued in the future with deeper analysis of the models and capabilities with other diseases. Furthermore, electronic health records could have more insight to offer, but lots stays hidden without closer exploration. This should be harnessed to assist healthcare professionals in their work.

References

Imran Ahmad, 2020. *40 Algorithms Every Programmer Should Know.* Packt Publishing.

Anthony K. Akobeng, 2007. *Understanding diagnostic tests 1: sensitivity, specificity and predictive values.* Received 8 June 2006; revised 4 December 2006; accepted 8 December 2006. Blackwell Publishing.

Douglas G. Altman and J. M. Bland. 1994. *Statistics Notes: Diagnostic tests 2: predictive values*
. British Medical Journal Publishing Group 309(6947), 102.

Vishal Anjaiah Gujjary and Ashutosh Saxena. 2011. *A neural network approach for data masking.* Neurocomputing (Amsterdam) 74(9), 1497-1501.

Gene V. Ball, Barri J. Fessler and S. L. Bridges. 2014. *Oxford Textbook of Vasculitis.* 3rd edition. Oxford University Press.

Ajay Baranwal, Alizishaan Khatri and Tanish Baranwal. 2019. *What's New in TensorFlow 2.0.* 1st edition. Packt Publishing.

Rohan Bhardwaj, Ankita R. Nambiar and Debojyoti Dutta. 2017. A Study of Machine Learning in Healthcare *In: 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC),* IEEE, 236-241.

Sandra Brasil, Carlota Pascoal, Rita Francisco, Vanessa Dos Reis Ferreira, Paula A. Videira and Gonçalo Valadão. 2019. *Artificial Intelligence (AI) in Rare Diseases: Is the Future Brighter?* Genes 10(12), 978.

Ebubekir Buber and Banu Diri. 2018. Performance Analysis and CPU vs GPU Comparison for Deep Learning. *In: 2018 6th International Conference on Control Engineering & Information Technology (CEIT),* IEEE, 1-6.

Pornchai Chaweewat and Jai G. Singh. 2020. *An electricity price interval forecasting by using residual neural network.* International transactions on electrical energy systems 30(9), 15 pages.

Chien-Hua Chen, Po-Hsiang Lin, Jer-Guang Hsieh, Shu-Ling Cheng and Jyh-Horng Jeng. 2020. Robust Multi-Class Classification Using Linearly Scored Categorical Cross-Entropy. *In: 202020 3rd IEEE International Conference on Knowledge Innovation and Invention (ICKII),* IEEE, 200-203.

Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. *In: Proceedings of the 22nd ACM SIGKDD International Conference on knowledge discovery and data mining,* ACM, 785-794.

Yu Cheng, Fei Wang, Ping Zhang and Jianying Hu. 2016. Risk prediction with electronic health records: a deep learning approach. *In: 16th SIAM International Conference on Data Mining 2016,* Society for Industrial and Applied Mathematics, 432-440.

Hector Chinoy and Robert G. Cooper. 2018. *Myositis.* Oxford University Press.

Davy Cielen, Arno D. B. Meysman and Mohamed Ali. 2016. *Introducing Data Science: Big data, machine learning, and more, using Python tools.* Manning Publications.

Patrick Davey and David Sprigings. 2018. *Diagnosis and Treatment in Internal Medicine.* Oxford University Press.

Jay Dawani, 2020.
Hands-On Mathematics for Deep Learning. Packt Publishing.

Wei Di, Anurag Bhardwaj and Jianing Wei. 2018. *Deep Learning Essentials.* Packt Publishing.

Khaled E. Emam and Luk Arbuckle. 2013. *Anonymizing Health Data.* O'Reilly Media, Inc.

Reynold Erwandi and Suyanto Suyanto. 2020. Improved Residual Neural Network for Breast Cancer Classification. *In: 2020 3rd International Conference on Information and Communications Technology (ICOIACT),* IEEE, 362-366.

European Commission. 2021. *Rare diseases. https://ec.europa.eu/info/research-and-innovation/research-area/health-research-and-innovation/rare-diseases_en*. 3.5.2021.

Aurélien Géron, 2017. *Hands-On Machine Learning with Scikit-Learn and TensorFlow.* O'Reilly Media, Incorporated.

Jiawei Han, Jian Pei and Micheline Kamber. 2011. *Data Mining: Concepts and Techniques.* 3rd edition. Morgan Kaufmann.

David J. Hand, 2009. *Measuring classifier performance: a coherent alternative to the area under the ROC curve.* Machine learning 77(1), 103-123.

Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. 2016. Deep Residual Learning for Image Recognition. *In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* IEEE, 770-778.

Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar and Pierre-Alain Muller. 2019. *Deep learning for time series classification: a review.* Data Min Knowl Disc 33(4), 917-963.

Hassan Ismail Fawaz, Benjamin Lucas, Germain Forestier, Charlotte Pelletier, Daniel F. Schmidt, Jonathan Weber, Geoffrey I. Webb, Lhassane Idoumghar, Pierre-Alain Muller and François Petitjean. 2020. *InceptionTime: Finding AlexNet for time series classification.* Data mining and knowledge discovery 34(6), 1936-1962.

Jinmeng Jia, Ruiyuan Wang, Zhongxin An, Yongli Guo, Xi Ni and Tieliu Shi. 2018. *RDAD: A Machine Learning System to Support Phenotype-Based Rare Disease Diagnosis.* Front. Genet. 9, 587.

Heechul Jung, Min-Kook Choi, Jihun Jung, Jin-Hee Lee, Soon Kwon and Woo Y. Jung. 2017. ResNet-Based Vehicle Classification and Localization in Traffic Surveillance Systems. *In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),* IEEE, 934-940.

Chayakrit Krittanawong, Hongju Zhang, Zhen Wang, Mehmet Aydar and Takeshi Kitai. 2017. *Artificial Intelligence in Precision Cardiovascular Medicine.* Journal of the American College of Cardiology 69(21), 2657-2664.

Rahul Kumar, 2019. *Machine Learning Quick Reference.* Packt Publishing.

Min Lin, Qiang Chen and Shuicheng Yan. 2013. *Network In Network.* arXiv , 10 pages.

James Loy, 2019. *Neural Network Projects with Python.* Packt Publishing.

Navin K. Manaswi, 2018. *Deep Learning with Applications Using Python: Chatbots and Face, Object, and Speech Recognition With TensorFlow and Keras.* 1st edition. Apress.

Ryan Marvin, Mark Nga and Amos Omondi. 2018. *Python Fundamentals.* Packt Publishing.

Umberto Michelucci, 2019. *Advanced Applied Deep Learning: Convolutional Neural Networks and Object Detection.* Apress.

Caron Molster, Debra Urwin, Louisa Di Pietro, Megan Fookes, Dianne Petrie, Sharon Van Der Laan and Hugh Dawkins. 2016. *Survey of healthcare experiences of Australian adults living with rare diseases.* Orphanet J Rare Dis 11(1), 30.

Jojo Moolayil, 2018. *Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python.* Apress.

Stéphanie Nguengang Wakap, Deborah M. Lambert, Annie Olry, Charlotte Rodwell, Charlotte Gueydan, Valérie Lanneau, Daniel Murphy, Yann Le Cam and Ana Rath. 2020. *Estimating cumulative point prevalence of rare diseases: analysis of the Orphanet database.* European journal of human genetics: EJHG 28(2), 165-173.

Orphanet. 2021a. *Dermatomyositis. https://www.orpha.net/consor/cgi-bin/OC_Exp.php?lng=EN&Expert=221*. 21.6.2021.

Orphanet. 2021b. *Inclusion body myositis. https://www.orpha.net/consor/cgi-bin/OC_Exp.php?Lng=GB&Expert=611*. 21.6.2021.

Orphanet. 2021c. *Polymyositis. https://www.orpha.net/consor/cgi-bin/OC_Exp.php?Lng=GB&Expert=732*. 21.6.2021.

Orphanet. 2021d. *Vasculitis. https://www.orpha.net/consor/cgi-bin/OC_Exp.php?Expert=52759&lng=EN*. 17.6.2021.

Arjun Panesar, 2020. *Machine Learning and AI for Healthcare: Big Data for Improved Health Outcomes.* Apress.

Adnan Qayyum, Junaid Qadir, Muhammad Bilal and Ala Al-Fuqaha. 2021. *Secure and Robust Machine Learning for Healthcare: A Survey.* RBME 14, 156-180.

Neil J. Salkind, 2010. *Encyclopedia of Research Design.* SAGE Publications, Inc.

Julia Schaefer, Moritz Lehne, Josef Schepers, Fabian Prasser and Sylvia Thun. 2020. *The use of machine learning in rare diseases: a scoping review.* Orphanet journal of rare diseases 15(1), 145.

TensorFlow. 2021. *tf.keras.metrics.Accuracy | TensorFlow Core v2.5.0. https://www.tensorflow.org/api_docs/python/tf/keras/metrics/Accuracy*. 9.7.2021.

Ivan Vasilev, Daniel Slater, Gianmario Spacagna, Peter Roelants and Valentino Zocca. 2019. *Python Deep Learning.* 2nd edition. Packt Publishing.

Stephen D. Walter, 2005. *Sensitivity, Specificity, False Positive, False Negative and Predictive Value. Encyclopedia of Biostatistics.* 3rd edition. John Wiley & Sons, Ltd.

Fushuai Wang, Jiyuan Qiu, Zhefeng Wang and Wenrui Li. 2021. Intelligent recognition of surface defects of parts by Resnet. *In: Journal of physics. Conference series,* IOP Publishing, 6 pages.

Hadley Wickham, 2014. *Tidy Data.* Journal of statistical software 59(10), 1–23.

Keng-Thye Woo, Choong-Meng Chan, Yoke M. Chin, Hui-Lin Choong, Han-Kim Tan, Marjorie Foo, Vathsala Anantharaman, Grace S. L. Lee, Gilbert S. C. Chiang, Puay H. Tan, Cheng H. Lim, Chorh C. Tan, Evan Lee, Hwee B. Tan, Stephanie Fook-Chong, Yeow-Kok Lau and Kok-Seng Wong. 2010. *Global Evolutionary Trend of the Prevalence of Primary Glomerulonephritis over the Past Three Decades.* Nephron. Clinical practice 116(4), 337-346.