UPPSALA
UNIVERSITET

UPTEC STS 26023
Examensarbete 30 hp
Juni 2026

Interpretable deep-learning models for automatic ECG classification

Olle Wernersson

Civilingenjörsprogrammet i system i teknik och samhälle

UPPSALA UNIVERSITET

Interpretable deep-learning models for automatic ECG classification

Olle Wernersson

Abstract

Deep neural networks (DNNs) have demonstrated expert-level accuracy in automated 12-lead ECG diagnosis, yet their "black-box" nature remains a major barrier to clinical adoption. For medical professionals to safely rely on artificial intelligence, the underlying decision-making processes must be transparent. This thesis addresses this challenge by combining inherently interpretable data representations with post-hoc explanation techniques to develop accurate and transparent diagnostic models.

To achieve structural interpretability, a β-variational autoencoder (β-VAE) was trained on the MIMIC-IV-ECG dataset. The model successfully compressed 12-lead median beats into a low-dimensional space consisting of 32 latent factors. Visual and statistical evaluations confirmed that these factors can capture distinct clinical ECG measurements, including ventricular rate, QT interval, and PR interval.

To test the adaptability of the learned latent space, the MIMIC-trained β-VAE was fine-tuned on the PTB-XL training dataset to extract latent factors. Four different classification architectures (logistic regression, XGBoost, TabPFNv2.5, and TabICLv2) were then trained on these extracted training factors and evaluated on the unseen test data. The results indicate that this dimensionality reduction preserves the vast majority of relevant diagnostic information. When evaluating 23 diagnostic subclasses, the TabICLv2 model achieved a Macro-AUC of 0.927, performing on par with a 101-layer deep residual network trained directly on raw signals (AUC 0.929). Across all 71 distinct diagnostic classes, the model achieved a Macro-AUC of 0.906 compared to the residual network's 0.925. This remaining performance gap was partly driven by rhythm-related diagnoses, as the extraction of median beats naturally discards temporal inter-beat dynamics.

Furthermore, applying SHAP analysis made it possible to track how the more flexible classification methods used these latent factors in a non-linear manner. When detecting Left Bundle Branch Block (LBBB), the evaluated models consistently prioritized the exact same latent variable (Factor 8) across both the MIMIC-IV-ECG dataset and the external PTB-XL dataset. This cross-dataset and cross-architecture stability indicates that the classifiers independently learned to rely on this same underlying morphological information.

Overall, this thesis shows that pairing a VAE latent space with detailed SHAP analysis enables the construction of flexible classification models whose decisions can be traced back to visual ECG changes. The slight drop in predictive performance compared to unconstrained black-box networks is a trade-off. However, if this transparency helps doctors verify the AI's reasoning, this approach could be a step toward integrating deep-learning models into clinical practice.

Teknisk-naturvetenskapliga fakulteten

Uppsala universitet, Utgivningsort Uppsala

Handledare: Fabio Bonassi, Jiawei Li (Co-supervisor) Ämnesgranskare: Antonio Horta Ribeiro

Examinator: Elísabet Andrésdóttir

Populärvetenskaplig sammanfattning

Hej alla läsare, i den här uppsatsen har jag undersökt en specifik metod som inte används inom vården idag, men som kanske kan göra det i framtiden om den visar sig lösa problemet med nuvarande black-box-modeller på ett tillräckligt bra sätt. Men detta kommer att kräva mer utveckling av metoden än vad jag har gjort i denna uppsats. Metoden jag har undersökt är designad för att kunna vara ett substitut för just black-box-modeller, genom att kunna ge förklaringar till vilken information modellen använder för att ställa sin diagnos. Det handlar i detta fall om diagnoser av hjärtat, och informationen som har undersökts i uppsatsen är hjärtsignaler som är tagna från 12 olika vinklar på samma gång. Det är ett standardiserat hjärttest och kallas på engelska för 12-lead ECG. Detta test används överallt i världen, och i Sverige används det bland annat i ambulansen eller på sjukhuset om man misstänker hjärtproblem.

Grundidén bakom metoden jag undersökte är att man kan ta denna signaldata från 12-lead ECG-testet, som annars består av tusentals datapunkter, och transformera den till 32 datapunkter istället. Detta gjordes med en variational autoencoder (VAE) som lärde sig att återskapa 12-lead ECG-datan med hjälp av dessa 32 datapunkter. Eftersom denna VAE kan återskapa signalen med endast dessa 32 punkter, innebär det att de innehåller all nödvändig information om hela 12-lead ECG:t. Då utnyttjar vi att informationen finns inbakad i dessa, för att diagnostisera hjärttillstånd enbart med hjälp av dessa 32. Vi kollar alltså inte på det stora 12-lead ECG:t längre. När vi väl kan diagnostisera med dessa 32 faktorer, så undersöktes hur modellerna använde dem för att ställa sin diagnos.

När detta har gjorts så är kanske den största utmaningen kvar för att verkligen lösa black-box-problemet: att förstå hur dessa 32 faktorer exakt översätts till termer som doktorer kan förstå och använda i praktiken. I denna uppsats kunde några faktorer översättas väl till enskilda kliniska mått som doktorer använder, och detta kunde också visualiseras. Men för att verkligen uppnå en modell som inte alls är någon black-box, så behövs det forskas vidare på hur man kan skapa faktorer som alltid är lätttolkade och helt oberoende av varandra.

Contents

1. Introduction

During the last decade there has been a surge in research for AI in healthcare. The number of papers released in the area of AI in healthcare has steadily increased over time, but only in the last decade the number went from 158 scientific articles in 2014 to 731 scientific articles in 2024 [1]. This surge has brought important progress towards better models and integration for doctors, especially in the field of image analysis. Image analysis is important for many fields in medicine, like X-ray images of bones and teeth, and mammography looking for cancer. The heart can also be looked at through ultrasounds but more commonly the heart is studied through electrical signals taken from the skin with leads. These recordings of electrical signals are called electrocardiograms (ECGs). It has been shown that for 12-lead ECGs, deep neural networks (DNNs) can diagnose heart conditions on par with expert-level performance [2, 3], and in some cases even be able to identify conditions that normally are not identified by experts.

However, despite this high accuracy of DNNs, they remain in limited use in practice for several reasons. For example in clinical settings, models need to be proven to improve outcomes of patients, and for a model to be useful to doctors, it must output more than just the probability of a diagnosis. Doctors need to understand the reasoning behind a prediction so that they can verify the result and retain control over the final clinical decision. It is here that a major limitation of traditional DNN approaches lies. DNNs almost always function as black boxes, meaning that while they give a diagnosis, they do not explain how they reached that conclusion. For doctors to rely on these suggestions in a clinical setting, they need to be able to trust and understand the process behind the result. This is not only to make the process more secure and better for patients, but is also a legal requirement. Recently, the EU introduced the European AI Act, which requires that high-risk AI systems such as ECG diagnosis systems be transparent and explainable among other things, to ensure patient safety.

To address these concerns, the field of explainable AI (XAI) has become increasingly popular. The goal of XAI is to reduce the opacity of these models and provide clear insights into their decision-making processes. Two main strategies exist within XAI, post-hoc methods which attempt to explain the decisions of a complex model after it has been trained (e.g., Grad-CAM), and ante-hoc methods which aim to make the model architecture itself structurally interpretable by design.

This thesis aims to contribute to the problem of interpretability in 12-lead ECG diagnosis by using a combination of ante-hoc and post-hoc methods. Specifically, the thesis investigates two main objectives, these are:

1. Training a variational autoencoder (VAE), which learns to project ECG beats into a disentangled, low dimensional space, allowing for an investigation into how these latent variables link to traditional ECG features
2. Training downstream classifiers on this latent space to demonstrate how interpretable diagnostic models can be constructed

This thesis will not experiment with different architectures in the encoder and decoder of the VAEs, but will instead use the architecture given by van de Leur et al.[4].

2. Background

The ECG is a powerful exam because it is non-invasive and widely available. AI has proven that it can use ECGs to identify risks and diseases at an early stage, which is crucial for successful treatment. An example of this is the detection of Chagas disease. Jidling et al. [5] demonstrated that a DNN could effectively detect this disease when it had resulted in chronic chagas cardiomyopathy, using only the ECG. This represents a significant step forward in the development of early detection tools. Another example is the detection of asymptomatic left ventricular dysfunction (ALVD). Attia et al. presented a DNN that detected ALVD [6], and their approach was tested and confirmed to work in more varied settings [7].

These findings show the potential of how AI can contribute to healthcare. Jidling et al. [5] adopted XAI by implementing the post-hoc Grad-CAM method. This highlights which parts of the ECG signal the model focused on. While useful, this does not say why these parts were useful and what information was extracted, and there is a continuous need for research to improve upon these post-hoc methods [8].

To achieve structural (ante-hoc) interpretability, one approach is to translate the high dimensional ECG signal into a lower dimensional set of features that can be individually understood. Recently, van de Leur et al. presented a VAE pipeline that maps 12-lead ECGs into a disentangled latent space [4]. The VAE that was used is a β-VAE. β-VAE was introduced by Higgins et al. from Google DeepMind in 2017 [9]. The idea was to make a VAE that produced a latent space with factors that were more interpretable for humans, and part of this is making the latent factors more disentangled which means that the factors are less correlated to each other. Higgins et al. [9] showed that adding a hyperparameter β to a certain part of the loss function (see Section 3.3) and setting β to an appropriate value greater than 1, resulted in more disentangled latent factors.

Van de Leur et al. transformed the 12-lead ECGs to 12-lead median beats and used the β-VAE with their own layer structure to compress the ECG signal into a set of 21 factors, a latent space they termed the "FactorECG". To understand the clinical meaning behind each of these disentangled factors they used a visualization technique called factor traversal. This visualizes how manipulating one of the factors at the time contributes to changes in the ECG signal.

3. Method

Throughout this thesis, several VAEs are trained to evaluate the impact of different datasets, median beat and ECG measurement extraction, preprocessing techniques and loss function. A summary of the main models referenced in this study is provided in Table 1.

Table 1: Summary and description of the VAE models trained in this thesis.

Model Name	Training Data	Description
β-VAE$_{\text{MIMIC}}$	MIMIC-IV-ECG	The primary VAE model in this thesis and is the base for all interpretability analysis.
β-VAE$_{\text{PTB-XL}}$	PTB-XL	Trained on PTB-XL median beats and ECG measurements extracted using our custom NeuroKit2 pipeline.
β-VAE$_{\text{PTB-XL-v2}}$	PTB-XL	Trained on PTB-XL, using the exact same settings and preprocessing as β-VAE$_{\text{MIMIC}}$ for fair comparison.
β-VAE$_{\text{12SL}}$	PTB-XL+	Trained using industry-standard Marquette 12SL median beats. Used for median beats and ECG measurements extraction comparison.
β-VAE$_{\text{MIMICv2}}$	MIMIC-IV-ECG	Trained on the full MIMIC-IV-ECG dataset with a median lead baseline fix (Evaluated in Appendix A.6).
β-TCVAE	MIMIC-IV-ECG	A Total Correlation VAE trained on the same data as β-VAE$_{\text{MIMIC}}$ (Evaluated in Appendix A.6).

3.1 Datasets and Preprocessing

PTB-XL is an open dataset with 21,799 12-lead ECGs from 18,869 patients, all leads having 10s of data [10]. The data comes in both 100 Hz and 500 Hz, but for this thesis, only the 500 Hz version was used. This gave a higher signal quality without making the training process slower, because extracting 1.2-second median beats from the 500 Hz data naturally results in exactly 600 data points per lead. This perfectly matches the input dimension of the VAE architecture. PTB-XL was chosen to be the first dataset to work on as this dataset is relatively small and therefore does not need as much computing power to be able to experiment with tuning hyperparameters.

The preprocessing of PTB-XL focused on creating median beats instead of using all the 10s of raw data for each lead. The median beats were extracted by using the NeuroKit2 library [11]. Firstly, we find the R-peaks of each lead II by using NeuroKit2 functions `ecg_clean` and `ecg_peaks`. Any ECG with fewer than three R-peaks was discarded, as multiple beats are needed to calculate a median. Secondly, we use the `epochs_create` function to slice up the 10s of each lead using the raw signals and the R-peaks. The slices are created to be 1.2s of length, 0.4s before the R-peak and 0.8s after the R-peak. Lastly, the median beats are created by taking the median of these slices.

Before training β-VAE$_{\text{12SL}}$ and β-VAE$_{\text{PTB-XL}}$, the entire median beat tensor is normalised. But before training β-VAE$_{\text{PTB-XL-v2}}$, linear interpolation is first applied, and then the mean of each single lead is removed to deal with shifting baselines that are far from 0. After this, the median

beat tensor is scaled in amplitude, but without zero-centering. Finally, the median beats are split into training, val, and test data. This is done in the suggested way by the creators of PTB-XL, using the `strat_fold` that comes with it: 1-8 for training, 9 for val, and 10 for test. This is done so that the ratio of diagnoses stays the same and so that patients stay inside only one dataset.

MIMIC-IV-ECG is also an open dataset with the exception that the diagnosis labels are restricted. MIMIC-IV-ECG is larger than PTB-XL with approximately 800,000 diagnostic electrocardiograms across nearly 160,000 unique patients [12]. These ECGs are 10 seconds in length and sampled at 500 Hz just like what is available in PTB-XL.

Because the diagnostic labels are sensitive data, they were only downloaded to a local computer and not uploaded to Google Drive or Colab. This means that any processing involving the labels, such as filtering out certain diagnoses and training downstream classifiers, was done locally without GPUs. Training the VAEs is an unsupervised task that does not need labels, so this could still be done on cloud GPUs. Median beats for the MIMIC-IV-ECG dataset were extracted in the same way as for PTB-XL. However, after training an initial VAE on this data, its reconstructions revealed that the dataset contained some faulty median beats. This led to an additional preprocessing and outlier discarding step, which is detailed in Section 3.2.

The MIMIC-IV-ECG comes with machine measurements that the machine makes, this includes the variables summarized in Table 2.

Table 2: Raw Machine measurements extracted from the MIMIC-IV-ECG dataset.

Variable	Description	Unit
`rr_interval`	Time between successive R-waves	msec
`p_onset`	Time at the onset of the P-wave	msec
`p_end`	Time at the end of the P-wave	msec
`qrs_onset`	Time at the beginning of the QRS complex	msec
`qrs_end`	Time at the end of the QRS complex	msec
`t_end`	Time at the end of the T-wave	msec
`p_axis`	The electrical axis of the P-wave	degrees
`qrs_axis`	The electrical axis of the QRS complex	degrees
`t_axis`	The electrical axis of the T-wave	degrees

These raw machine measurements can be used to calculate standard clinical ECG features, such as QRS duration and ventricular rate. These clinical features are needed later to evaluate what information the latent factors have captured. The translations from raw variables to clinical ECG measurements are summarized in Table 3.

Table 3: Calculated clinical ECG measurements based on the raw machine variables.

Clinical ECG Measurement	Calculation
Ventricular rate	60000 / `rr_interval`
PR interval	`qrs_onset` – `p_onset`
QRS duration	`qrs_end` – `qrs_onset`
QT interval	`t_end` – `qrs_onset`
QTc interval	QT interval $/\sqrt{\texttt{rr\_interval}/1000}$
P duration	`p_end` – `p_onset`
ST duration	`t_end` – `qrs_end`
R, T, and P axis	`qrs_axis`, `t_axis`, and `p_axis`

3.2 MIMIC-IV-ECG Data Exclusion

To detect faulty data in the ECG leads in MIMIC-IV-ECG, a VAE trained on all 12-lead median beats with diagnosis was used, specifically its decoder to make reconstructions of all data in this dataset. Then the R^2, MSE and MAE are calculated for each 12-lead reconstruction. To see the reconstructions that the VAE is struggling with we looked at the reconstructions and original data of the samples with the lowest R^2. The 100 with the lowest R^2 were visually inspected. From this an exclusion list was started with the 61/100 that looked like faulty data. The types of faulty data that were discovered from these 100 samples were missing data/leads, pacemaker signals, misaligned R-peak and low voltage signal.

To improve the data quality for the VAE training, a plan was formed to identify and discard samples with these issues. This filtering process consisted of the following five steps in order:

1. **Missing leads or low voltage:** Any missing data points (NaN) were filled with 0. The standard deviation of every lead was then calculated. If a lead had a standard deviation of less than 0.005, it was considered a missing lead or having too low voltage and was excluded.
2. **Pacemaker data:** Pacemaker data was excluded by checking text reports for phrases like "Pacemaker rhythm" and "Demand pacing" (excluding reports like "Wandering atrial pacemaker" as this does not necessarily indicate a pacemaker). Additionally, ICD-10 diagnoses were checked, and records containing "Z950", "Z95810", "Z45010", "Z45018", or "Z4502" (which all indicate a pacemaker or defibrillator) were removed.
3. **VES/VPB diagnoses:** Samples with the diagnosis VES/VPB (ICD-10 code "I493") were removed. A cardiologist recommended removing both these and the pacemaker data, as these median beats can be abnormal and skew the training of a VAE.
4. **Excessive missing data:** 12-lead ECGs containing a lead with more than 2% missing data were discarded. Since one median beat lead consists of 600 data points, this threshold equals 12 missing data points.
5. **Misaligned R-peaks:** The median beat extraction method is designed to align all R-peaks close to the 200th data point. Having the R-peaks aligned makes it easier for the VAE to pick up detailed information about the peak itself, rather than using latent factors just to inform where the R-peak is located. To detect misplaced R-peaks, a lead baseline correction

was performed by subtracting the mean of each lead. The squared sum of all 12-leads was calculated for each of the 600 timesteps to find the maximum energy. If the highest sum within a ±45 step region around the 200th point, divided by the highest sum across the entire beat, resulted in a ratio below 0.40, the sample was added to the exclusion list.

All the steps where data were filtered are visualized in Figure 1.

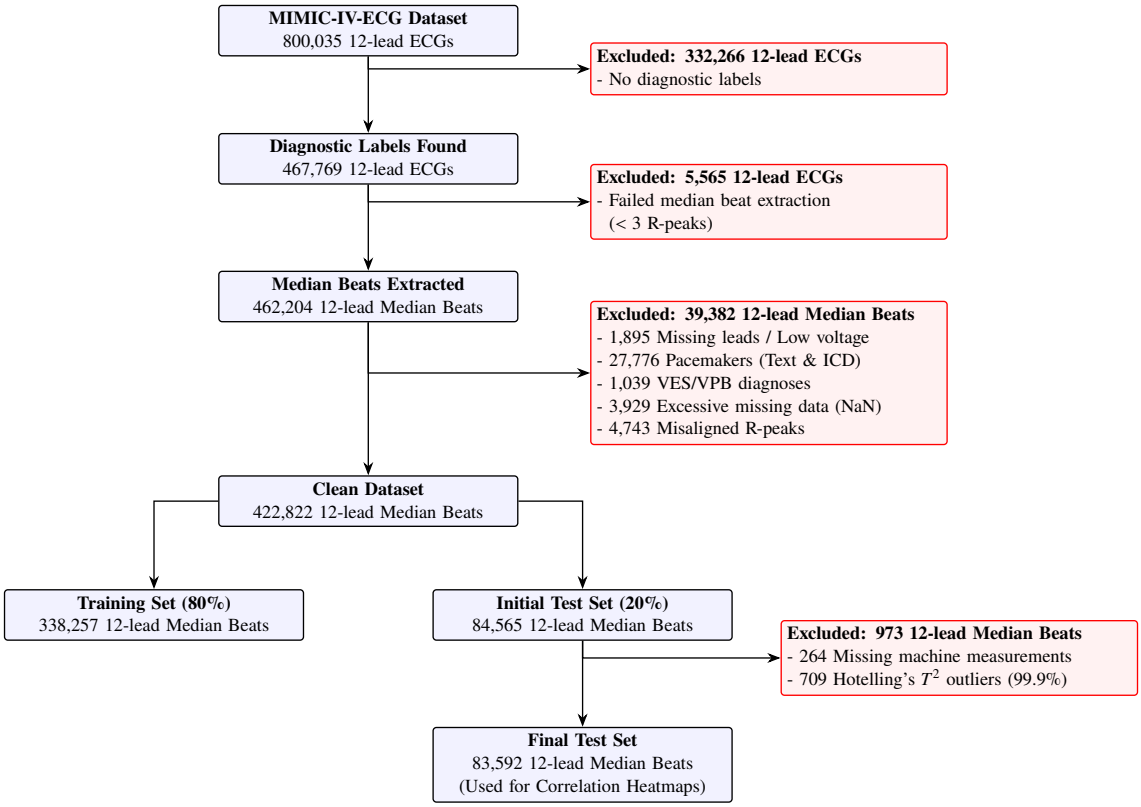

Figure 1: Complete data exclusion flowchart detailing the initial MIMIC-IV-ECG samples, the amount of data removed at each filtering stage, and the resulting sets for training β-VAE$_{MIMIC}$

Even though training a VAE is an unsupervised task that does not need labels, samples without diagnostic labels were initially excluded. This was done to make the MIMIC-IV-ECG dataset more similar to the PTB-XL dataset, where every sample has a diagnosis. By doing this, the experimental pipeline stays consistent across both datasets, meaning that every median beat used to train the VAE could also be used directly to train the downstream classification models.

After the filtering steps, the remaining data was split into an 80:20 training and test set. However, there are some data leakage issues with this approach, even though stratification was used. Samples from the same patient could end up in both the training and test sets. Additionally, without a third validation split, the test set was used for early stopping during the training of β-VAE$_{\text{MIMIC}}$. This might mean that the R^2 reconstruction score on the test set is slightly optimistic.

To address both the exclusion of unlabeled data and the data leakage issues, an extended experiment was conducted. A new model, β-VAE$_{\text{MIMICv2}}$, was trained on the entire dataset without requiring labels and using a strict train:val:test split with the recommended strat folds (see Figure 23). The reconstruction on this proper test set actually showed an improved R^2, indicating that the initial baseline results were not too misleading. The details of this model are discussed further in Appendix A.6.

3.3 VAE Architecture and Training

The layer structure of the encoder and decoder is the same for all VAEs trained here and is taken from van de Leur et. al.[13]. The encoder and decoder are 1D Causal CNNs and are both 7 layers deep, exact configuration can be found in Appendix A.1 (see Table 5). Both the encoder and decoder are Gaussian networks. This means the output from the encoder consists of 32 latent factors, represented as 32 means and 32 standard deviations. Similarly, the output of the decoder is the 12-lead ECG reconstruction formatted as a (600, 12) tensor. Because it is a Gaussian decoder, it outputs two (600, 12) tensors. One for the predicted means and one for the predicted standard deviations of the reconstructed signal.

One way that our training differs from [13] is the addition of a lower bound of 0.1 to the decoder's predicted standard deviation during the reconstruction loss calculation, this is because the data is noisy and so it makes sense to never be 100% certain of data points, and it also made the training more stable, especially for the validation reconstruction loss. The loss functions for the training of β-VAEs consist of two parts. The reconstruction loss ($\mathcal{L}_{\text{recon}}$) and the KL divergence loss ($\mathcal{L}_{\text{KL}}$). The reconstruction loss looks at the decoder's output compared to the original 12-lead ECG and is calculated with:

$$\mathcal{L}_{\text{recon}}(x, \mu, \sigma) = \frac{1}{2}\log(2\pi) + \frac{1}{2}\log(\sigma^2) + \frac{(x-\mu)^2}{2\sigma^2} \tag{1}$$

The KL divergence, which only looks at the latent factors that the encoder outputs is calculated by:

$$\mathcal{L}_{\text{KL}}(\mu, \sigma) = -\frac{1}{2}\left(1 + \log(\sigma^2) - \mu^2 - \sigma^2\right) \tag{2}$$

Training techniques to try to improve disentanglement and performance outcome included using the AdamW optimizer alongside a ReduceLROnPlateau learning rate scheduler. Gradient clipping with a maximum norm of 1.0 was also applied to prevent exploding gradients. Furthermore, the KL divergence weight (β) was increased in the form of a sine curve. The β value started at 4 and increased to a target of 32 over the first 20 training epochs.

3.4 Comparison of ECG Measurements Extraction

To test our custom median beat and ECG measurement extraction pipeline that is using NeuroKit2, a comparative analysis was performed against the median beats and ECG measurements from PTB-XL+. PTB-XL+ is an extension to the PTB-XL dataset that provides ready to use median beats and ECG measurements generated by state of the art methods, including GE Healthcare's Marquette™ 12SL™ [14].

By using the 12SL median beats and ECG measurements to train a β-VAE we can be more certain that the disentanglement results are less sensitive to the specific method used for creating median beats and extracting ECG measurements. We denote this model β-VAE$_{\text{12SL}}$. To compare we also trained a β-VAE that we call β-VAE$_{\text{PTB-XL}}$ using the same network architecture and hyperparameters but on our extracted median beats and ECG measurements from PTB-XL. Complete training specifications for these models are in Appendix A.1 (Table 6).

Comparing the latent space correlations between these two models showed structural similarities. For example, both models got high correlations for the same ECG measurements and struggled on the same ones that could be directly compared. The primary differences were found in ECG measurements that lacked a direct one-to-one mapping between the two extraction methods. Detailed correlation heatmaps and a thorough row by row comparison of this validation experiment are provided in Appendix A.5.

3.5 Implementation Details and SHAP Analysis

For implementing the VAE the PyTorch library was used as well as NumPy and matplotlib. For data handling and traditional machine learning tasks and visualisations, other libraries such as Scikit-learn, Pandas, Seaborn, and TabPFN were also used.

To investigate how the different downstream classification models use the latent factors, SHAP (Shapley Additive exPlanations) values were calculated. Lundberg and Lee [15] introduced SHAP and explained how a model's prediction can be factored into a base value plus a SHAP value for each feature, or latent factor in our case. Conceptually, SHAP values represent the impact of each factor by measuring how a model's output changes when a specific factor is included in the prediction versus when it is excluded. Computing the exact SHAP value requires calculating the outputs of a model while considering all possible combinations of features. Because the evaluated classification architectures in this thesis have different internal structures, specific SHAP algorithms were applied to make this computationally feasible and comparable.

For the tree-based XGBoost ensemble, the `TreeExplainer` algorithm [16] was used. This algorithm uses the internal structure of decision trees to calculate exact SHAP values efficiently in polynomial time, outputting them in log-odds. Exact SHAP values for the logistic regression model were calculated using `shap.LinearExplainer`. Because linear models are computationally inexpensive to explain, the entire training dataset could be used as the background reference point. The resulting SHAP values for logistic regression are also naturally in log-odds.

TabPFN [17] and TabICL [18] are recently developed foundation models designed for tabular data. Due to their transformer-based architectures, the `TreeExplainer` cannot be used. Instead, the model-agnostic `shap.Explainer` algorithm was applied. Unlike the exact explainers, this algorithm only estimates the SHAP values through sampling, making it significantly more computationally expensive as the models have to go through inference many times. It also requires a reference background dataset. To keep computation times feasible, a representative background dataset of 100 training samples was used. This subset was stratified so that the ratio of positive cases matched the overall training data distribution. For TabICLv2, built-in KV caching was used to speed up the inference, allowing SHAP values to be estimated for all 2,195 test samples.

By default, these transformer models output predictions as probabilities. To calculate SHAP values in log-odds, a custom wrapper function was implemented. This function applied the inverse of the standard logistic function (the logit function) to the models' predicted probabilities before passing them to the `shap.Explainer`. This makes it so that these models get SHAP values in log-odds space too, making it possible to directly compare the results across all four classification models.

4. Results

4.1 Evaluation of the VAE's disentanglement

To evaluate what the latent factors had learned, we investigated their correlation with the calculated clinical ECG measurements on the test set. However, to get reliable correlation values, we first needed to handle noisy measurement data. Since the calculated ECG measurements had a more Gaussian-like distribution than the raw variables, Hotelling's T^2 distance was used to detect and discard outliers from the test set before creating the correlation heatmaps. PR interval, P duration and P axis were excluded from this detection method as they had a large amount of data missing. It is suspected this can be due to that the P-wave is the weakest signal out of the measured signals and can also be missing for patients with some diagnoses. The remaining ECG measurements were normalised, and PCA was used to reduce the dimension of each row from 7 to 3. This made it possible to visually inspect this distribution and see which points would be removed. A confidence level of 99.9% was used to calculate the T^2 threshold, which resulted in 709 out of 84,301 rows being discarded from the test set evaluation.

The presence of these outliers can significantly reduce the Pearson correlations between the ECG measurements and latent factors. A comparison was made using the β-VAE$_{\text{MIMIC}}$ model. Complete training specifications for this model are in Appendix A.1 (Table 6). The correlations were calculated both with the outliers removed using the 99.9% T^2 confidence interval, and without any outliers being removed. When evaluating the test set without any statistical outlier removal, the correlations were notably weaker (see Appendix A.2, Figure 14).

We can see that even with the careful confidence of 99.9% we still see differences. For example, the highest correlation values for P duration and P axis strengthened from -0.03 to -0.16 and from -0.08 to -0.19, respectively. Furthermore, disentanglement slightly improved across all evaluated ECG measurements.

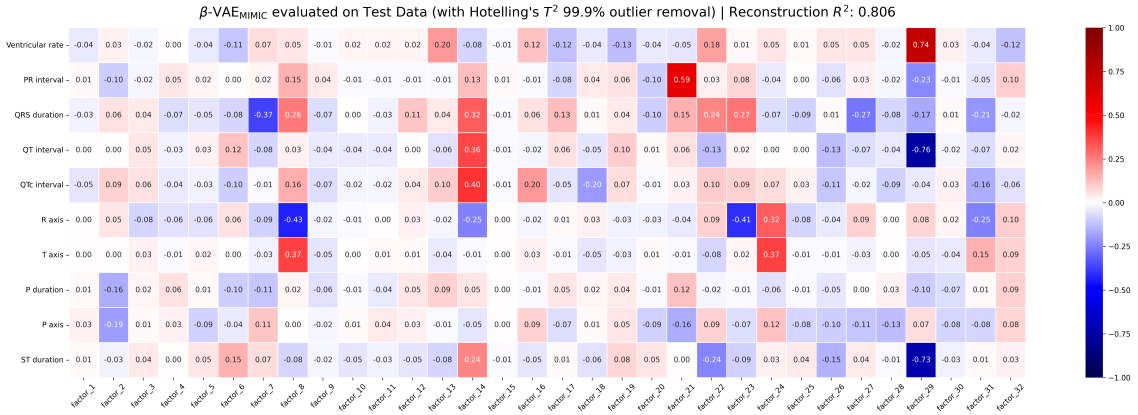

Figure 2: Correlation heatmap evaluating β-VAE$_{MIMIC}$ latent factors on the MIMIC-IV-ECG final test set after discarding outliers using Hotelling's T^2 with 99.9% confidence.

This correlation heatmap (Figure 2) was shown to a cardiologist, and a discussion was held regarding the factors and whether they made clinical sense. The cardiologist thought it made sense that Factor 29, which has a positive correlation with ventricular rate, also has a negative correlation with duration measurements. Furthermore, it was noted as logical that the QTc interval is uncorrelated with this factor compared to the QT interval. The cardiologist also thought it made sense that both the R axis and T axis were positively correlated with the same factor (Factor 24), and

that Factor 14, which has a positive correlation with the QTc interval, also has positive correlations with other duration measurements. For further studies, a blind test with written feedback could provide more objective information on the factors, instead of an open discussion where we might induce confirmation bias.

4.2 External Validation of VAE Representations

To test how VAEs can work for new unseen data, a test using external data that the model was not trained on is done. This test will in this thesis be constructed by using the β-VAE$_{\text{MIMIC}}$ that has only been trained on MIMIC-IV-ECG and using it to reconstruct median beats for PTB-XL. This is expected to be a very challenging task for β-VAE$_{\text{MIMIC}}$ because of how different the datasets are. (1) MIMIC-IV-ECG comes from patients in the USA, while PTB-XL comes from people in Germany. (2) MIMIC-IV-ECG comes from patients in a hospital setting and intensive care setting, while PTB-XL has a large portion of samples from healthy individuals. (3) The data was recorded from different machines and is collected in different times, MIMIC-IV-ECG being collected from 2008-2019 while PTB-XL is from 1989-1996.

β-VAE$_{\text{MIMIC}}$ was tested on how well it would reconstruct median beats made from PTB-XL. To make a fair comparison a model β-VAE$_{\text{PTB-XL-v2}}$ was trained on PTB-XL training data with the exact same hyperparameters as the β-VAE$_{\text{MIMIC}}$ model. Unlike the previously discussed β-VAE$_{\text{PTB-XL}}$ which was specifically preprocessed to match the 12SL dataset, this v2 model uses the exact same baseline correction and linear interpolation as β-VAE$_{\text{MIMIC}}$. Calculating R^2 on the PTB-XL test set yielded 0.729 for β-VAE$_{\text{PTB-XL-v2}}$ and 0.687 for β-VAE$_{\text{MIMIC}}$. This shows that even a model trained on a bigger dataset can perform worse when changing data distributions.

An experiment was made where β-VAE$_{\text{MIMIC}}$ was fine-tuned on the PTB-XL training data by training it with 1% of the original learning rate. This was done 5 different times with different amounts of epochs trained (5, 10, 20, 30 and 40 epochs) and their R^2 performance was evaluated on the PTB-XL test data and the MIMIC-IV-ECG test data to see if the model would become better on the PTB-XL distribution and if there would be any catastrophic forgetting on the MIMIC-IV-ECG data it was originally trained on. The results of this experiment can be seen in Figure 3.

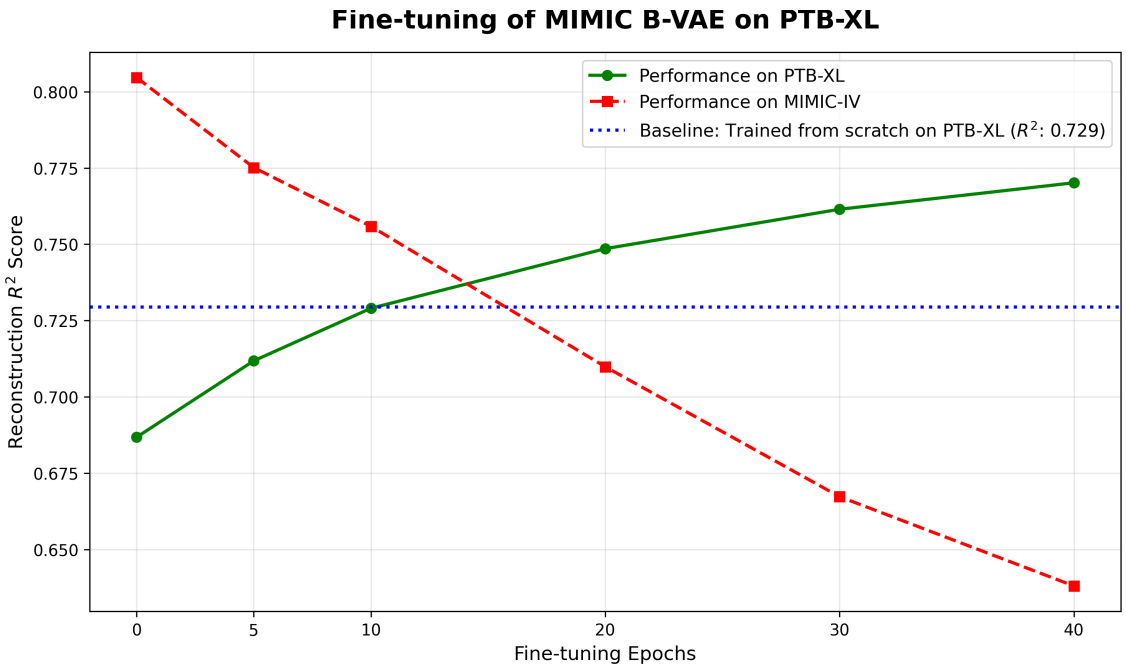

Figure 3: Reconstruction R^2 scores. β-VAE$_{MIMIC}$ is fine-tuned on PTB-XL training data. The green solid line shows improving performance on PTB-XL test data, while the red dashed line shows the performance degradation on MIMIC-IV-ECG test data. The blue dotted line represents the performance of β-VAE$_{PTB\text{-}XL\text{-}v2}$ trained only on PTB-XL training data.

We can see that with this way of fine-tuning a new model with 10 epochs gets the same R^2 performance as a model only trained on PTB-XL training data gets. Models with higher R^2 performance can be achieved but at the cost of catastrophic forgetting on the MIMIC-IV-ECG data.

4.3 Impact of Transfer Learning on Disentanglement

In the transfer learning experiment (Figure 3), we saw that β-VAE$_{\text{MIMIC}}$ could be fine-tuned to achieve greater R^2 on PTB-XL but at the same time suffer degraded R^2 on MIMIC-IV-ECG. Correlation heatmaps were generated for the baseline model(β-VAE$_{\text{PTB-XL-v2}}$), β-VAE$_{\text{MIMIC}}$, and the fine-tuned models (10 and 40 epochs), (see Appendix Figure 13)

A primary observation from these heatmaps is that the heatmaps for the β-VAE$_{\text{MIMIC}}$ look very similar to its heatmaps on MIMIC-IV-ECG (Figure 2) For example, Factor 29 has high correlation with the same ECG measurements, Factor 21 with PR interval, Factor 2 for P duration, the correlation is actually slightly higher for PR interval and P duration than on MIMIC-IV-ECG, even before any fine-tuning on PTB-XL. This suggests that correlation heatmaps for this VAE can keep its structure with data distribution shifts and ECG measurement extraction method. Another observation is that fine-tuning β-VAE$_{\text{MIMIC}}$ does not disrupt its disentanglement, but rather seems to slightly improve it for this experiment on PTB-XL.

4.4 Initial Interpretability Analysis

Using the β-VAE$_{\text{MIMIC}}$ model, we can create factor values for each 12-lead ECG in MIMIC-IV-ECG. These factor values can then be used to predict the diagnostic label for a given 12-lead ECG.

To train the logistic regression model, the exact same training samples were used as for the β-VAE$_{\text{MIMIC}}$ training. However, instead of using the 12-lead median beats, the input data now consisted of the corresponding 338,257 rows of latent factors generated by the VAE encoder. For this classification task, samples with the Left Bundle Branch Block (LBBB) diagnosis were labeled as positive, and all remaining samples were treated as negative. To test the logistic regression model, the exact same test samples that were used for the correlation heatmaps were also used. This test set consists of 83,592 samples represented as latent factors, out of which 1,113 have the label of LBBB. Training a logistic regression model without any regularization yields a model with the coefficients visualized in Figure 4.

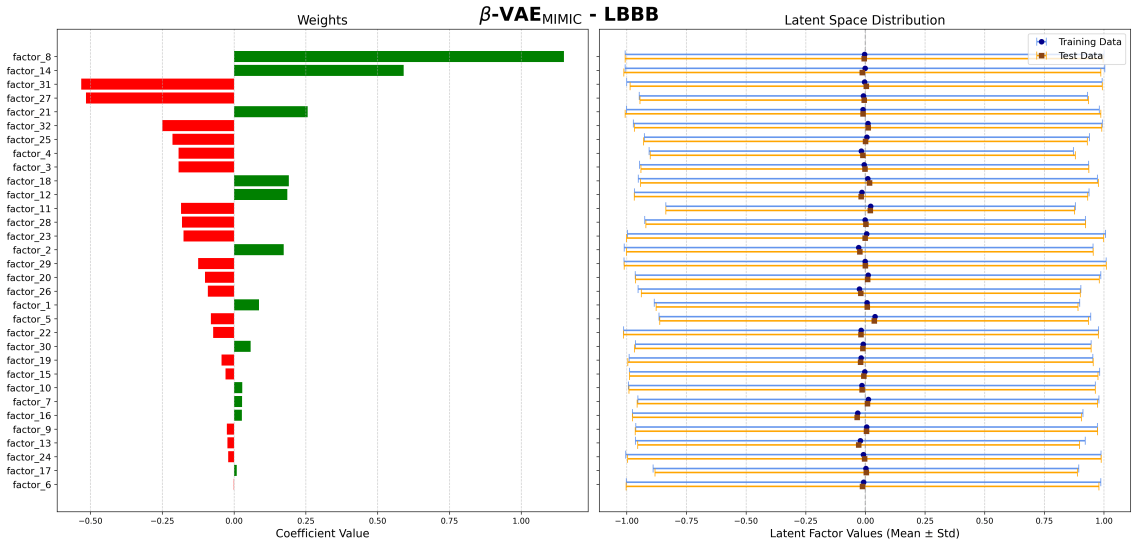

Figure 4: Left: Logistic regression coefficients for predicting LBBB. Green bars indicate factors that increase the probability of an LBBB diagnosis, while red bars indicate factors that decrease it. Right: Latent factors distribution for the same factors for the training data the Logistic regression model and β-VAE$_{MIMIC}$ was trained on, as well as the test data.

We can observe that Factor 8 has the highest absolute coefficient value, but this could simply be a result of its distribution. If Factor 8 had a very narrow distribution with a small standard deviation, then the coefficient value for that factor would have to be amplified to have the same impact on the classification as the other factors. But as we can see from the distribution plot on the right, when looking at all factor values for both the training and test set we see that they form a $\mathcal{N}(0, 1)$ distribution (which is what we want and are trying to force with the KL divergence part of the loss function). Now we can confidently say that Factor 8's value for a single 12-lead ECG has the most impact on if this logistic regression model will classify it as LBBB or not.

We can also look at how the latent space is distributed by dividing it up by wether the 12-lead ECG has the label LBBB or not. Doing this for the training data results in Appendix Figure 11.

Factor 8 is most differently distributed of the factors depending on if the 12-lead ECG has the label LBBB or not. For the Factor 8 values with the label LBBB, its distribution is shifted to the positive values but still stays in a normal shape, we can see that all factors have a normal shaped distribution. Looking at the shift in the distribution for the LBBB group, it corresponds very well to what the coefficient for that factor is in the logistic regression model, both in the magnitude of the shift and the direction.

Using the decoder of β-VAE$_{\text{MIMIC}}$ a factor traversal is made by keeping all factors at 0 while manipulating Factor 8 between -5 and +5, this while letting the decoder reconstruct the median beat with these factor values and plotting all the reconstructions in the same plot. This results in

Figure 5.

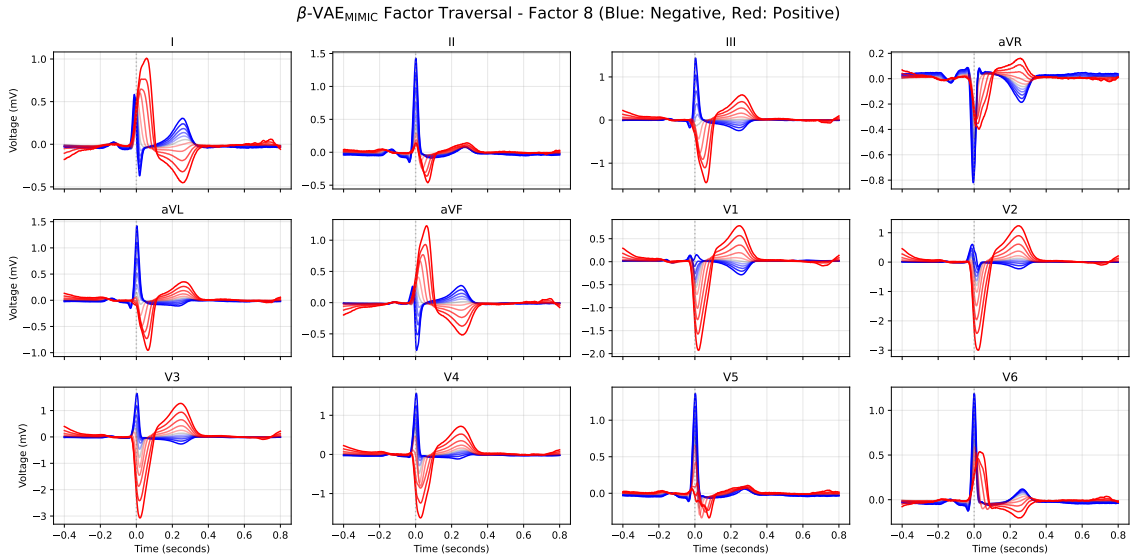

Figure 5: Factor traversal of Factor 8 between -5 and +5, with all other latent factors set to zero.

Changing Factor 8 in this way between -5 and +5 appears to have a substantial effect on the QRS wave and T wave for most leads. After examining a few median beats of 12-lead ECGs in the LBBB group, they do seem to have this characteristic of a deeply negative and wide S wave in the V1-V4 leads for example. This is what we see for more positive values of Factor 8 as well.

Further interpretability on Factor 8 can be found in the correlation heatmap seen in Figure 2. For the ECG measurements calculated from MIMIC-IV-ECG, Factor 8 has the highest correlation with R axis(-0.43), T axis(0.37) and QRS duration(0.26)

4.5 External Validation of Classification Models

Following the external validation of the VAE reconstructions, a similar cross-dataset evaluation was performed for the classification tasks. Instead of reconstructing median beats, the focus here is to evaluate the downstream diagnostic performance on unseen data. The goal of this experiment is to demonstrate that the interpretability and predictive power derived from the latent factors can generalize to new environments. A successful outcome on the PTB-XL dataset indicates that the pipeline can handle the kind of data distribution shifts expected in real clinical settings, such as data collected from different hospitals or varying ECG machines.

Using β-VAE$_{MIMIC}$, latent factors were created for the median beats of all 12-lead ECGs from PTB-XL. A logistic regression model was trained the same way as for the MIMIC-IV-ECG, just that PTB-XL has two subcategories of LBBB: Incomplete LBBB and Complete LBBB. CLBBB is the label looked at for these logistic regression models. The coefficient values for these logistic regression models are seen in Figure 6.

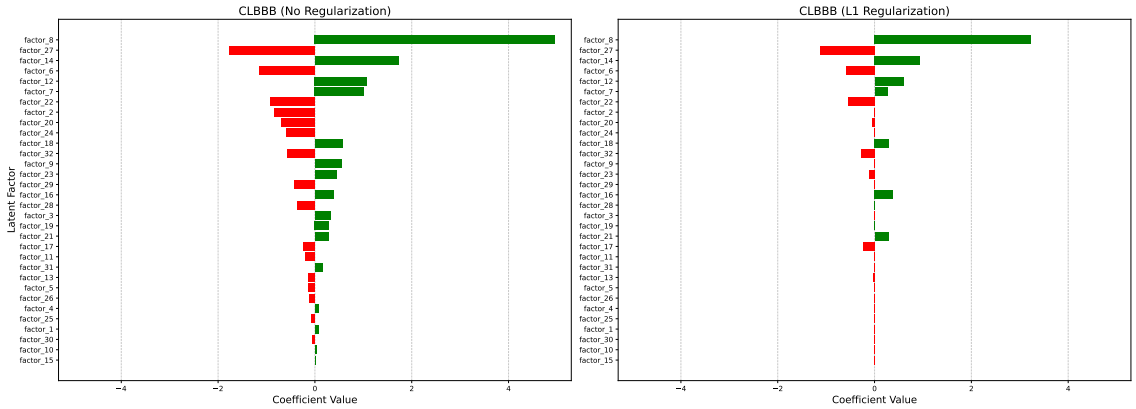

Figure 6: Logistic regression coefficients for predicting CLBBB using the β-VAE$_{MIMIC}$ latent factors. The left graph shows model without regularization, while the right graph shows model with L1 regularization with an inverse regularization strength of $C = 0.1$.

At this stage, the same VAE model has been used to produce latent factors for two different datasets, with separate logistic regression models trained on each. Comparing the coefficient values for the CLBBB group from PTB-XL (Figure 6) and the LBBB group from MIMIC-IV-ECG (Figure 4), Factor 8 consistently exhibits the highest magnitude in both models. Additionally, Factor 14 retains a high positive coefficient value across both datasets. However, applying the β-VAE$_{\text{MIMIC}}$ to the unseen PTB-XL dataset introduces a domain shift. Because the VAE was not trained on PTB-XL, the distribution of the generated latent factors deviates from a standard normal distribution. This deviation is confirmed in Appendix Figure 10.

Using the exact same latent factors and data splits, an XGBoost model was trained to predict CLBBB on the PTB-XL dataset. For interpretability, we do not have the same transparent situation as with logistic regression, where the 32 weights can be studied directly. Instead, the XGBoost model consists of an ensemble of 1000 decision trees, each with binary splitting criteria 5 levels deep. This complex structure makes it practically impossible to derive interpretability directly from the model architecture. Therefore, SHAP analysis was used to explain the predictions of XGBoost and the tabular foundation models.

An example for the concept of SHAP values: if a certain sample has a high value for Factor 8, and excluding this factor causes the model's predicted risk for CLBBB to decrease, then Factor 8 receives a positive SHAP value for that sample. This indicates that the specific value of Factor 8 for this patient effectively increases the model's certainty of a CLBBB diagnosis. Looking at SHAP values for multiple samples can therefore give us an idea on how the model uses a certain factor in general for its prediction.

To compare the decision-making process across all four classification models, their SHAP values were standardized into log-odds (as detailed in Section 3.5). The resulting global feature importances and SHAP value distributions are presented in Figure 7.

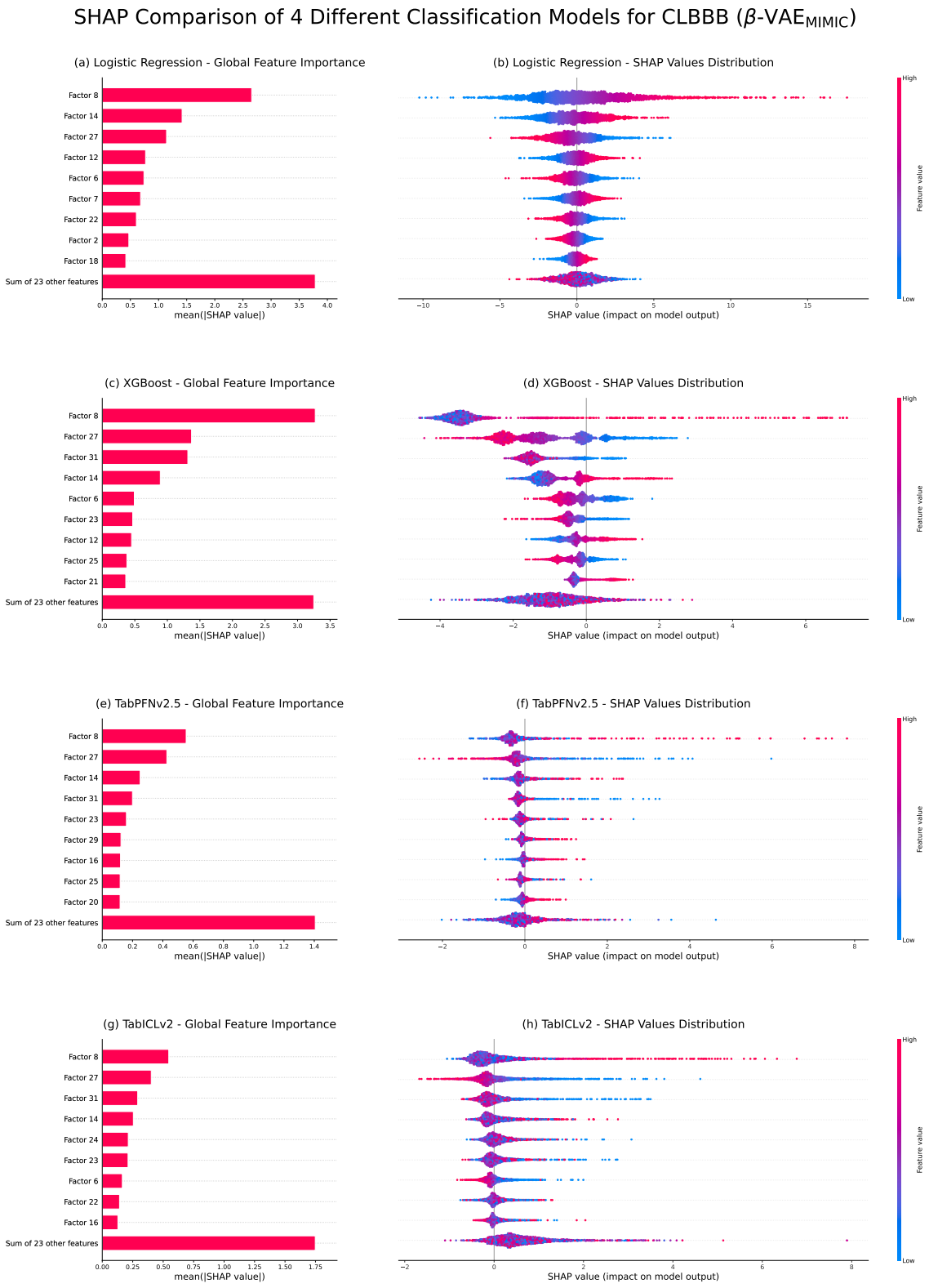

Figure 7: SHAP comparison of four different classification models (Logistic Regression, XGBoost, TabPFNv2.5, and TabICLv2) trained to predict CLBBB using β-VAE$_{MIMIC}$ latent factors. The left column (a, c, e, g) shows the global feature importance when considering SHAP values in the PTB-XL test data. All 4 models are most affected by the same Factor 8 and have similar factors in the top spots. The right column (b, d, f, h) shows the SHAP value distributions. Note that for TabPFNv2.5 it only shows SHAP values for 950 out of the 2195 samples in the test data which is why it looks a little sparser.

Figure 7 shows that Factor 8 is still the factor with the highest impact on if the XGBoost model predicts CLBBB or not, just as Factor 8 had the highest impact when training a logistic regression model on the same latent factors. The SHAP value distribution for Factor 8 gives us more information. There is a large gathering of SHAP values around -3.5 that includes values of Factor 8 from the low to the middle of the spectrum, and it is not until Factor 8 has a high value that the SHAP value increases rapidly. This indicates a non-linear relationship between Factor 8 and the XGBoost model prediction.

While TabICLv2 got performance metrics for CLBBB (AUC: 0.997, AUPRC: 0.888), testing another diagnosis such as Anterior Myocardial Infarction (AMI) got a significantly lower performance (AUC: 0.734, AUPRC: 0.053). This even though both diagnoses having a similar prevalence of approximately 2% in the PTB-XL dataset. With metrics so low for AMI, interpretability also gets harder as the models themselves cannot reliably diagnose. Computing SHAP values for AMI (see Appendix Figure 12) shows how uncertain the models are and no single factor can provide much confidence.

From the SHAP value distributions we can see that there is a non-linear relationship between Factor 8 and the SHAP values for all models except for logistic regression, but to make this easier to see we can put Factor 8 and the SHAP values in a scatter plot instead. Furthermore, for the XGBoost model it is possible to calculate so called SHAP interaction values exactly. These are calculated from the `shap.TreeExplainer` and are what happens if we ask how much more the SHAP values change when we change two factors together compared to if we change them one at a time. The scatter plots for Factor 8 for CLBBB and the interaction with Factor 27 can be found in Appendix A.2. The main takeaway from this analysis is that, apart from the logistic regression model, the value for Factor 8 where they start to increase their log odds for CLBBB rapidly is around 1, and there is not really any factor that interacts too much with Factor 8 in a way that changes the shape or log odds for the scatter plot. This might be good news for Factor 8 as it seems pretty independent in this regard.

Now we have analysed the SHAP values for CLBBB and AMI in detail, but how about the other diagnoses in PTB-XL? The SHAP values were calculated for all diagnoses in PTB-XL for XGBoost models with the same method and settings as when done for CLBBB and AMI. To visually analyse SHAP values for multiple diagnoses at the same time, a visualisation was made in a more compressed form using information from both global feature importance and SHAP value distribution into a single square. The visualization for 44 of the PTB-XL diagnoses can be seen in Figure 8. These are the ones in the diag level that we compare classification performance on later in Section 4.6.

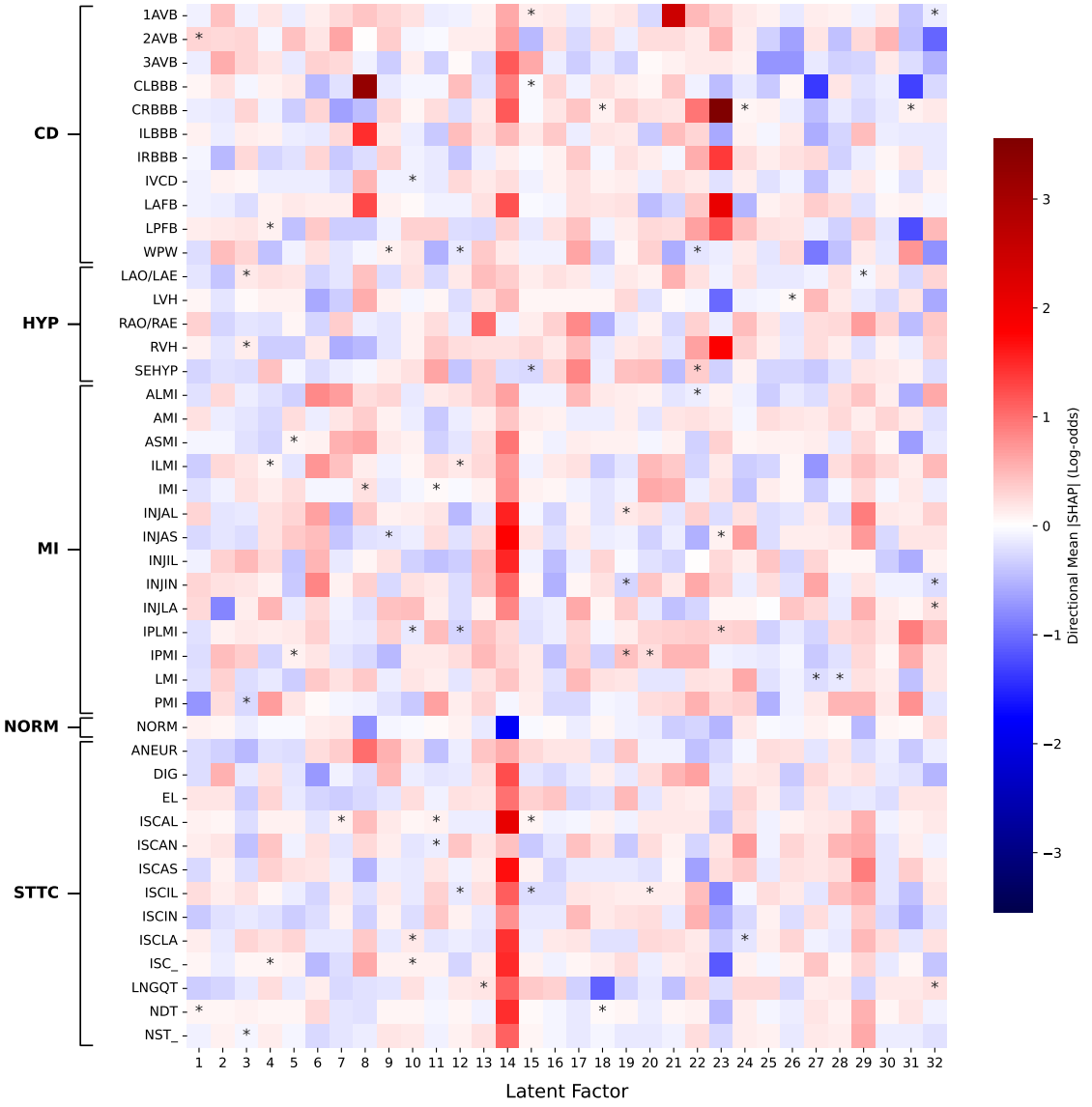

Figure 8: Directional SHAP importance across 44 diagnostic classes using XGBoost. The magnitude of each square corresponds to the mean absolute SHAP value, while the color indicates the direction of the effect based on Pearson correlation between the SHAP values and the factor. Red if the correlation is positive and blue otherwise. Asterisks (*) mark factors with weak correlation (between -0.10 and 0.10). The color on the squares with the (*) should therefore be interpreted with caution

We can learn about the factors by seeing them in this overview perspective, because now we can see how the factors are being used for different diagnoses at the same time. For example, something that pops out from this is the almost complete red line that forms from Factor 14, with the clear blue square for the NORM diagnosis which is the label for normal ECG. The takeaway from this red line is that the XGBoost models consistently separate normal ECGs from other ECGs with the specific Factor 14. We can also look at the CLBBB row that we have already analysed in more detail and see the same factors that we talked about before stand out, like Factor 8, and it is red, meaning that a more positive value increases the model's certainty that the patient has CLBBB. For AMI, the row looks bland with no clear factor standing out, just like we saw before.

It is possible to make this visualisation for the other classification models also, and the reason why XGBoost is chosen here is because the better performing models TabPFNv2.5 and TabICLv2

take a lot of compute to create SHAP values and the SHAP values are not exact with the methods we have. XGBoost was chosen over logistic regression as it performs better without losing too much interpretability. There was a comparison made between logistic regression, XGBoost and TabICLv2 on the super-diag level (see Figure 18 in Appendix A.3) which showed similarities like we have seen before for CLBBB that different models find the same factors to be important. The same kind of visualisations for rhythm and form levels can be found in Figures 20 and 19 (Appendix A.3). Some clear cases were seen in these, like the rhythm diagnoses of STACH (Sinus Tachycardia) and SBRAD (Sinus Bradycardia) which have deep red and deep blue for Factor 29 respectively. This makes a lot of sense as this is the factor with the high positive correlation with ventricular rate. There is also the LPR (prolonged PR interval) diagnosis in the form level that gets a deep red for Factor 21, the factor with a high positive correlation with PR interval.

4.6 Performance Comparison

Strodthoff et al. [19] used a variety of DNN methods to train and evaluate discriminative performance in PTB-XL. They separated the PTB-XL diagnostics into 6 different levels (diag, sub-diag, super-diag, rhythm, form and all) and added which diagnosis falls into which level into the PTB-XL dataset so that others can also use these. The DNNs are trained directly on the raw 12-lead ECGs and the output is on all diagnostics inside the level at once. The results can be seen in Figure 2 of their publication [19].

In this paper, the input data for classification are the 32 latent factors and the output is a binary classification between the looked at diagnostic and all others. This means that a separate model is made for each diagnostic. Using the same levels and the same split between training and test data (with the exception of excluded data that median beats could not be made for), a summary of the classification models discriminative performance can be seen in Figure 9. The only data excluded from the test set are 3 samples with the rhythm-level PACE (pacemaker) diagnosis. This means we still use 2195 out of the 2198 that Strodthoff et al. used to test performance. This will not change the macro-AUC in any meaningful way.

Table 4: Discriminative performance calculated in Macro-AUC of downstream classifiers trained on latent factors across all PTB-XL evaluation levels. The highest score among our models in each category is highlighted in bold.

Method	all	diag.	sub-diag.	super-diag.	form	rhythm
TabICLv2 (β-VAE$_{\text{PTB-XL-v2}}$)	.895	.915	.920	.902	.845	.900
TabICLv2 (β-VAE$_{\text{MIMIC}}$)	.901	.919	.922	.906	.863	.899
TabICLv2 (β-VAE$_{\text{MIMIC}}$ FT 40)	**.906**	**.922**	**.927**	**.908**	**.869**	**.906**
TabPFNv2.5 (β-VAE$_{\text{MIMIC}}$ FT 40)	.899	.920	.925	.905	.866	.876
XGBoost (β-VAE$_{\text{MIMIC}}$ FT 40)	.877	.889	.909	.894	.843	.889
LogReg (β-VAE$_{\text{MIMIC}}$ FT 40)	.851	.864	.894	.846	.809	.868
`xresnet1d101` (Strodthoff et al. [19])	.925	.937	.929	.928	.896	.957

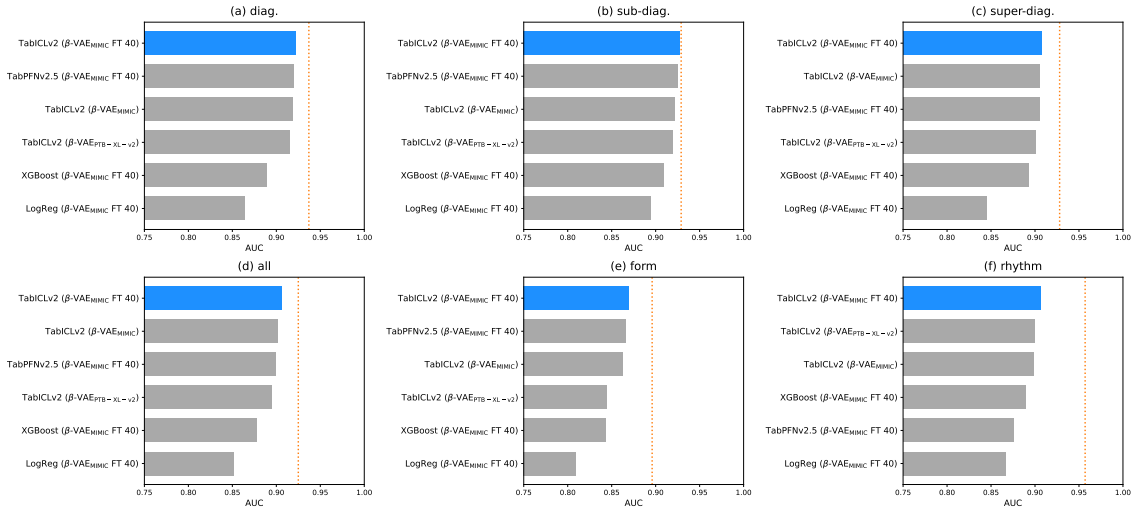

Figure 9: Visual comparison of macro-averaged AUC scores across the six PTB-XL evaluation levels. The models within each subfigure are sorted by Macro-AUC. The orange dotted line represents the performance of the 101-layer deep residual network (`xresnet1d101`) by Strodthoff et al. [19].

The TabICLv2 evaluated on the fully fine-tuned β-VAE$_{\text{MIMIC}}$ FT40 latent factors performed better than all other classification models tested here and latent factors from β-VAE$_{\text{PTB-XL-v2}}$ and all β-VAE$_{\text{MIMIC}}$ versions. The overall trend with some exceptions was that with all classification models, the performance was worse with the β-VAE$_{\text{PTB-XL-v2}}$ and got slightly better with the untuned β-VAE$_{\text{MIMIC}}$, and then performance got better with more fine-tuning up to the FT40 that was tested. That the performance on the PTB-XL dataset improves with fine-tuning on that dataset makes sense. The part that is a little surprising is that the β-VAE$_{\text{MIMIC}}$ often performed better than the β-VAE$_{\text{PTB-XL-v2}}$ latent factors even though we saw that the reconstruction R^2 was worse for β-VAE$_{\text{MIMIC}}$.

To train the XGBoost models for Figure 9, the `XGBClassifier` function was used from the XGBoost Python library. Fine-tuning of hyperparameters was done with respect to the latent factors from β-VAE$_{\text{PTB-XL-v2}}$ and β-VAE$_{\text{MIMIC}}$ and all its fine-tuned models, and the performance on the validation set on the highest level of diagnoses. That being STTC, NORM, MI, HYP and CD. The final hyperparameters that were used can be found in Appendix Table 7.

TabPFNv2.5 is the most recent version (at the project development stage) of the tabular foundation model TabPFN [17]. To fit the TabPFNv2.5 models for Figure 9, the `TabPFNClassifier` function was used from the `tabpfn` Python library. The weights for the default TabPFNv2.5 classifier were downloaded to fit the models without an API, but instead with Google Colab GPUs. This is okay to do with PTB-XL as the database is open.

TabICLv2 is a recent addition to tabular foundation models like TabPFN [18] and is said to compete with the performance of TabPFNv2.5 and offer more speed and scalability [18]. This model also comes with fine-tuning capabilities. But even without fine-tuning we see that it barely outperformed TabPFNv2.5 on PTB-XL discrimination. TabPFNv2.5 does not come with public fine-tuning capabilities. To train the TabICLv2 models for Figure 9 fine-tuning was not used. An experiment was carried out to train on the β-VAE$_{\text{MIMIC}}$ FT 40 latent factors, and a similar performance as non fine-tuned TabICLv2 was achieved. Full experimental details with hyperparameters and performance comparison are provided in Appendix A.4.

5. Discussion

β-VAE$_{\text{PTB-XL-v2}}$ and β-VAE$_{\text{PTB-XL}}$ exhibit different disentanglement structures even though they were trained with the exact same hyperparameters. This was generally observed during the hyperparameter tuning process, particularly when training on the smaller PTB-XL dataset. Even for models with identical hyperparameters and preprocessing, the correlation heatmap would show that sometimes certain ECG measurements were highly correlated to only one factor (making them disentangled), while other times different measurements became more disentangled. Exactly which ECG measurements achieved clear disentanglement seemed somewhat random each time a new VAE was trained. While this phenomenon was most prominent when training on the smaller PTB-XL dataset compared to the larger MIMIC-IV-ECG dataset, it was still present to some degree even with the VAEs trained on MIMIC-IV-ECG. The consequence of this phenomenon, at least for unsupervised VAEs like the ones trained in this study, is that we cannot guarantee that different VAEs will produce corresponding factors. This means that the interpretation of downstream models must be individually tuned and validated based on the specific VAE used.

Locatello et al. [20] released a paper where they used 6 different VAE models with the same architecture and hyperparameters, including β-VAE and β-TCVAE, and trained them on special datasets (where data can be generated from factors of variations like rotation and size) with different regularisation strengths and random seeds. The disentanglement performance of these models was calculated with metrics like the FactorVAE Score. Interestingly, the results showed that it was not only the VAE model type and regularisation strength that affected disentanglement, but also the random seed to a high degree. Random seed in this case does not mean how the data is split up into training and validation data, so it does not change the training data distribution. Looking at the given code by Locatello et al. [20], the random seed affects the initialization of weights in the VAE, for example, and what data is included in the training batches. The only other sources of randomness from our training are the sampling of the latent factors during the forward pass, which most certainly affected the VAE models in Locatello et al., and the dropout in the encoder.

With this in mind, that the exact disentanglement structure of the correlation heatmap can be different just from these random processes, we still try to compare them with each other and also with the heatmap presented by van de Leur et al. [4] (Figure 3B). Comparing Van de Leur's correlation heatmap to our β-VAE$_{\text{MIMIC}}$ correlation heatmap (Figure 2) shows that factors with similar disentanglement through the ECG measurements are created even though we trained on different datasets. The clearest examples are that a factor with both high correlation with ventricular rate and QT interval at the same time seems to always be created, like our Factor 29, as well as factors with correlations in the same direction for duration measurements, like our Factor 14.

Van de Leur et al. had an impressive 0.96 correlation with ventricular rate, which is a strong relationship. Having high correlations close to 1 or -1 is exactly what we want for a factor, as the connection between the factor and this aspect of the ECG then becomes even more robust. Van de Leur et al. trained on 1,144,331 samples. In this thesis, a correlation strength of 0.90 was achieved for the QT interval in only one training run with 659,741 samples, and around 0.82 was achieved when training on 338,257 samples, down to around 0.7 when using PTB-XL's training data of around 17,000 samples. These observations indicate that more training data can make the correlations between latent factors and ECG measurements stronger.

The factors' correlations with ECG measurements are, however, only one aspect of what we can learn from the latent space. Using the factor traversals, other morphological structures can be documented, revealing more information that a factor carries. For example, β-VAE$_{\text{MIMIC}}$ Factor 8, when increased, creates median beats with deep S-waves in the V1-V4 leads and flips the T-wave

for many of the leads. In theory, it might be possible to explicitly include many more ECG measurements, like T-wave height and S-wave height for different leads, and pick up on this information through correlation metrics instead. However, the visual information provided by the factor traversals remains in a much more detailed and holistic form.

For the factors to be useful, they also have to retain predictive information for classification purposes. When comparing our downstream classification performance to the benchmarks established by Strodthoff et al. [19] on the PTB-XL dataset, several notable insights emerge.

First, despite compressing the high-dimensional ECG signal into 32 latent variables, our best performing TabICLv2 model (β-VAE$_{\text{MIMIC}}$ FT 40) achieves highly comparable results for some evaluation levels. For instance, on the *sub-diag* level, our model achieves a Macro-AUC of 0.927, which is nearly identical to the 0.929 achieved by Strodthoff's best performing `xresnet1d101` model. Considering that the `xresnet1d101` is a highly flexible, 101-layer deep 1D residual network trained directly on the raw signal, achieving comparable performance using only 32 numerical values is a strong testament to the quality of the latent space. For the broader *diag* category, the difference remains relatively small (0.922 vs. 0.937). This demonstrates that the VAE successfully preserves the vast majority of the morphological information required for accurate pathology detection. The slight decrease in overall predictive performance is a trade-off that might be acceptable if the gain in interpretability makes the model more useful in practice. This aligns with recent findings by Patlatzoglou et al. [21], who systematically quantified the "cost of explainability" when comparing variational autoencoders to standard convolutional neural networks. By evaluating predictive tasks on estimating age, biological sex, and 5-year mortality risk, they demonstrated that forcing the model to compress the ECG into a small set of interpretable factors acts as an information bottleneck. This bottleneck inherently filters out high-frequency noise and irregular signal artifacts (such as pacemaker spikes) that black-box models can exploit to artificially boost performance. While smoothing out these details results in a slight drop in predictive power compared to residual networks, it increases the likelihood that the model bases its predictions on physiological changes coming from the heart. For example, their interpretable VAE pipeline successfully demonstrated that predictions for higher mortality risk were driven by known clinical markers like broadened QRS complexes and T-wave flattening, providing transparency for medical applications.

The most significant performance gap between the two approaches is observed in the *rhythm* level (0.906 for our model compared to 0.957 for `xresnet1d101`). This discrepancy is highly logical and highlights a fundamental limitation in our current data processing pipeline rather than the classification models themselves. Strodthoff et al. trained their models on the full 10-second continuous ECG signals, whereas our latent factors are derived exclusively from a 1.2-second median beat. Rhythm abnormalities, such as atrial fibrillation or premature ventricular contractions, are inherently temporal and are characterized by inter-beat variations (e.g., irregular R-R intervals). Constructing a median beat inevitably averages out or completely discards these crucial temporal dynamics. For diagnoses concerning irregular heartbeats, other strategies than using median beats will likely have to be used.

Having the ability to classify based on the 32 latent factors is good, but it would not be much more helpful or interpretable if the classification models did not tell us anything about how they used the factors. To begin with, a logistic regression model was tested on LBBB using the MIMIC-IV-ECG dataset (the same dataset that the VAE was trained on). By studying the weights of the logistic regression model, it could be determined that the most important factor was Factor 8, and that an increase in Factor 8 would make the model more inclined to classify the sample as positive for LBBB. Then, an external validation setting was used to test this VAE (β-VAE$_{\text{MIMIC}}$) and its ability to create useful latent factors for the PTB-XL dataset. Here, it was found that when training a logistic regression model for practically the same diagnosis, CLBBB, the same important Factor 8 showed up just like it did on MIMIC-IV-ECG trained logistic regression model. Not only did this

happen for logistic regression, but the other classification models (XGBoost, TabPFNv2.5, and TabICLv2) also relied heavily on the same factor. This result really shows that β-VAE$_{\text{MIMIC}}$'s Factor 8 is generally useful for the detection of LBBB.

To understand how all the four different classification models use the latent factors, SHAP values and SHAP interaction values were analyzed. The reason that SHAP works so well even for complicated transformer-based models such as TabICLv2 is that SHAP tells us information about how the model uses the input space. In our case, this is "only" 32 latent factors, whereas if SHAP had been used on a raw-signal model like `xresnet1d101`, it would have given a SHAP value for each data point in the ECG, which would be much harder to interpret.

With our situation of 32 factors, exactly how logistic regression uses these is easy to understand, as the model uses the factors independently of each other and they have a strictly linear impact on the prediction. But for the more complicated models, they can use combinations of multiple factors to differentiate heart conditions, and the factors have a non-linear impact on the model output. While this makes the models better at classifying diagnoses, it also makes the work to understand exactly how they used the factors more complicated.

In this thesis, the non-linear use of SHAP has been analyzed in the specific case of CLBBB in the external validation setting. Scatter plots of SHAP values depending on a specific factor show what the non-linear relationship looks like. Checking the factor with the most interaction impact with Factor 8 showed that, in this case, the interaction did not have enough impact to majorly change the interpretability of what Factor 8 means alone for the XGBoost model. While exact interaction values are easiest to compute for tree-based models like XGBoost, it is also possible to estimate SHAP values and SHAP interaction values for the tabular foundation models. However, this estimation is significantly more computationally expensive. In theory, all 32 factors for any classification model can be analyzed using scatter plots and estimated SHAP interaction values to reveal even more detailed information about how the models uses these factors.

5.1 Future Studies

While the VAE pipeline itself builds upon previous research, the detailed SHAP analyses performed in this thesis show how we can achieve interpretability for classification models like XGBoost and tabular foundation models. However, to make this approach useful in a clinical setting, there are areas that need to be further worked on.

First, there is a need for the development of a clinical inference interface that uses an interpretable pipeline like the VAE pipeline. Even if we have a highly accurate and interpretable model, a doctor will not have the time to manually go through SHAP scatter plots, feature importances, and interaction values to understand why the model made a certain diagnosis. A future study could focus on creating a system where the doctor directly gets the relevant information they need. One idea for this is to use a fine-tuned Large Language Model (LLM). This LLM could be used to process the large amounts of SHAP analyses and the interactions between the 32 factors in the background. It could then summarize the findings and present only the most important interpretable information to the doctor in plain text, perhaps alongside the visual factor traversals.

Secondly, a potential way to improve and expand the pipeline is to address the limitation of rhythm-level diagnoses. As we saw in the performance comparison, extracting median beats throws away much of the temporal dynamics that are needed to accurately detect irregular heartbeats. Future studies could try expanding this pipeline to handle continuous 10-second rhythm ECGs instead of just median beats. If a VAE framework could compress a full 10-second strip into interpretable latent factors, it would likely close the performance gap for rhythm diagnoses while keeping the interpretability that we get from the SHAP analysis.

Lastly, if we can force the latent factors to always change only one aspect or ECG measurement at the time, then the interpretability of the factors could be even more clear. With the current approach the VAEs have no supervision on exactly how standard ECG measurements are calculated and therefore often produce factors that change multiple measurements at the same time.

6. Conclusion

The overall aim of this thesis was to tackle the black-box nature of deep learning in automated ECG diagnosis by combining ante-hoc and post-hoc explainability methods. To achieve this, the work was guided by two main objectives.

The first objective was to train a VAE to project ECG beats into a disentangled, low-dimensional space and investigate how these latent variables link to traditional ECG features. By training a β-VAE on the large MIMIC-IV-ECG dataset, we successfully compressed 12-lead median beats into just 32 latent factors. Through correlation heatmaps and visual factor traversals, it was shown that these factors can capture distinct morphological traits and clinical ECG measurements, such as ventricular rate, duration intervals (PR interval, QT interval, etc.), and electrical axis. While the exact distribution of the disentanglement varies between training runs due to random training processes, the VAEs consistently captured similar ECG measurement relationships, such as the link between ventricular rate and the QT interval. The ability to isolate distinct morphological features became even more robust when the model was trained on large amounts of data.

The second objective was to train downstream classifiers on this latent space to demonstrate how interpretable diagnostic models can be constructed. The results showed that a low-dimensional latent space can retain the vast majority of the diagnostic information. For instance, on the *sub-diag* level, our best performing classification model (TabICLv2) achieved a Macro-AUC of 0.927, performing almost identically to a 101-layer deep residual network trained directly on raw signals (AUC 0.929). Furthermore, when evaluating across all 71 classes without exception (the *all* level), our model maintained a Macro-AUC of 0.906 compared to the residual network's 0.925. This performance gap on the *all* level was partly driven by rhythm-related diagnoses, since the median beat extraction process naturally discards temporal inter-beat dynamics.

We demonstrated methods to understand how classification models use these factors, and how the factors themselves relate to clinical ECG measurements. By applying SHAP analysis to flexible, non-linear classifiers like XGBoost and tabular foundation models, we could track their decision-making processes. For example, in the detection of CLBBB in an external dataset, all classification models consistently prioritized the exact same latent variable (Factor 8). This shows that despite their different architectures, the models independently learned to base their predictions on the same underlying morphological information.

Ultimately, this thesis demonstrates that the slight drop in predictive performance compared to unconstrained black-box networks is a trade-off. However, by pairing a VAE latent space with detailed SHAP analysis, it becomes possible to build flexible classification models where decisions can be traced back to visual changes in the ECG and ECG measurements. If this transparency helps doctors verify the AI's reasoning, this trade-off could be a step toward integrating deep-learning models into clinical practice.

References

[1] Md. Faiyazuddin, Syed Jalal Q. Rahman, Gaurav Anand, Reyaz Kausar Siddiqui, Rachana Mehta, Mahalaqua Nazli Khatib, Shilpa Gaidhane, Quazi Syed Zahiruddin, Arif Hussain, and Ranjit Sah. "The Impact of Artificial Intelligence on Healthcare: A Comprehensive Review of Advancements in Diagnostics, Treatment, and Operational Efficiency". In: *Health Science Reports* (2025).

[2] Antônio H. Ribeiro, Manoel Horta Ribeiro, Gabriela M. M. Paixão, Derick M. Oliveira, Paulo R. Gomes, Jéssica A. Canazart, Milton P. S. Ferreira, Carl R. Andersson, Peter W. Macfarlane, Wagner Meira Jr., Thomas B. Schön, and Antonio Luiz P. Ribeiro. "Automatic diagnosis of the 12-lead ECG using a deep neural network". In: *Nature Communications* (2020).

[3] Veer Sangha, Bobak J. Mortazavi, Adrian D. Haimovich, Antônio H. Ribeiro, Cynthia A. Brandt, Daniel L. Jacoby, Wade L. Schulz, Harlan M. Krumholz, Antonio Luiz P. Ribeiro, and Rohan Khera. "Automated multilabel diagnosis on electrocardiographic images and signals". In: *Nature Communications* (2022).

[4] Rutger R. van de Leur, Max N. Bos, Karim Taha, Arjan Sammani, Ming Wai Yeung, Stefan van Duijvenboden, Pier D. Lambiase, Rutger J. Hassink, Pim van der Harst, Pieter A. Doevendans, Deepak K. Gupta, and René van Es. "Improving explainability of deep neural network-based electrocardiogram interpretation using variational auto-encoders". In: *European Heart Journal – Digital Health* (2022).

[5] Carl Jidling, Daniel Gedon, Thomas B. Schön, Claudia Di Lorenzo Oliveira, Clareci Silva Cardoso, Ariela Mota Ferreira, Luana Giatti, Sandhi Maria Barreto, Ester C. Sabino, Antonio L. P. Ribeiro, and Antônio H. Ribeiro. "Screening for Chagas disease from the electrocardiogram using a deep neural network". In: *PLOS Neglected Tropical Diseases* (2023).

[6] Zachi I. Attia, Suraj Kapa, Francisco Lopez-Jimenez, Paul M. McKie, Dorothy J. Ladewig, Gaurav Satam, Patricia A. Pellikka, Maurice Enriquez-Sarano, Peter A. Noseworthy, Thomas M. Munger, Samuel J. Asirvatham, Christopher G. Scott, Rickey E. Carter, and Paul A. Friedman. "Screening for cardiac contractile dysfunction using an artificial intelligence–enabled electrocardiogram". In: *Nature Medicine* (2019).

[7] Rickey E. Carter, Patrick W. Johnson, Jordan B. Strom, Jonathan W. Waks, Andrew Krumerman, Kevin J. Ferrick, Roger DeRaad, Benjamin A. Steinberg, Mikolaj A. Wieczorek, Jessica Cruz, Zachi I. Attia, Francisco Lopez-Jimenez, Paul A. Friedman, Samir Awasthi, Mohan Krishna Ranganathan, Rakesh Barve, Heather M. Alger, Konstantinos C. Siontis, and Peter A. Noseworthy. "Multisite, External Validation of an AI-Enabled ECG Algorithm for Detection of Low Ejection Fraction". In: *JACC: Advances* (2026).

[8] Ahcène Boubekki, Konstantinos Patlatzoglou, Joseph Barker, Fu Siong Ng, and Antônio H. Ribeiro. "Explaining deep learning for ECG using time-localized clusters". In: *arXiv preprint arXiv:2509.15198* (2025).

[9] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. "beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework". In: *International Conference on Learning Representations*. 2017.

[10] Patrick Wagner, Nils Strodthoff, Ralf-Dieter Bousseljot, Dieter Kreiseler, Fatima I. Lunze, Wojciech Samek, and Tobias Schaeffter. "PTB-XL: A large publicly available electrocardiography dataset". In: *Scientific Data* (2020).

[11] Dominique Makowski, Tam Pham, Zen J. Lau, Jan C. Brammer, François Lespinasse, Hung Pham, Christopher Schölzel, and S. H. Annabel Chen. "NeuroKit2: A Python toolbox for neurophysiological signal processing". In: *Behavior Research Methods* (2021).

[12] Brian Gow, Tom Pollard, Larry A. Nathanson, Alistair Johnson, Benjamin Moody, Chrystinne Fernandes, Nathaniel Greenbaum, Jonathan W. Waks, Parastou Eslami, Tanner Carbonati, Ashish Chaudhari, Elizabeth Herbst, Dana Moukheiber, Seth Berkowitz, Roger Mark, and Steven Horng. "MIMIC-IV-ECG: Diagnostic Electrocardiogram Matched Subset (version 1.0)". In: *PhysioNet* (2023).

[13] Rutger R. van de Leur, Max N. Bos, Karim Taha, Arjan Sammani, Ming Wai Yeung, Stefan van Duijvenboden, Pier D. Lambiase, Rutger J. Hassink, Pim van der Harst, Pieter A. Doevendans, Deepak K. Gupta, and René van Es. *ecgxai: Source code for Variational Auto-Encoders in ECG interpretation*. `https://github.com/rutgervandeleur/ecgxai`. Accessed: 2026-04-01. 2022.

[14] Nils Strodthoff, Temesgen Mehari, Claudia Nagel, Philip J. Aston, Ashish Sundar, Claus Graff, Jørgen K. Kanters, Wilhelm Haverkamp, Olaf Dössel, Axel Loewe, Markus Bär, and Tobias Schaeffter. "PTB-XL+, a comprehensive electrocardiographic feature dataset". In: *Scientific Data* (2023).

[15] Scott M. Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems*. 2017.

[16] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. "From local explanations to global understanding with explainable AI for trees". In: *Nature Machine Intelligence* (2020).

[17] Léo Grinsztajn, Klemens Flöge, Oscar Key, Felix Birkel, Philipp Jund, Brendan Roof, Benjamin Jäger, Dominik Safaric, Simone Alessi, Adrian Hayler, Mihir Manium, Rosen Yu, Felix Jablonski, Shi Bin Hoo, Anurag Garg, Jake Robertson, Magnus Bühler, Vladyslav Moroshan, Lennart Purucker, Clara Cornu, Lilly Charlotte Wehrhahn, Alessandro Bonetto, Bernhard Schölkopf, Sauraj Gambhir, Noah Hollmann, and Frank Hutter. "TabPFN-2.5: Advancing the State of the Art in Tabular Foundation Models". In: *arXiv preprint arXiv:2511.08667* (2026).

[18] Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. "TabICLv2: A better, faster, scalable, and open tabular foundation model". In: *arXiv preprint arXiv:2602.11139* (2026).

[19] Nils Strodthoff, Patrick Wagner, Tobias Schaeffter, and Wojciech Samek. "Deep Learning for ECG Analysis: Benchmarks and Insights from PTB-XL". In: *arXiv preprint arXiv:2004.13701* (2020).

[20] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. "Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations". In: *Proceedings of the 36th International Conference on Machine Learning*. 2019.

[21] Konstantinos Patlatzoglou, Libor Pastika, Joseph Barker, Ewa Sieliwonczyk, Gul Rukh Khattak, Boroumand Zeidaabadi, Antônio H. Ribeiro, James S. Ware, Nicholas S. Peters, Antonio Luiz P. Ribeiro, Daniel B. Kramer, Jonathan W. Waks, Arunashis Sau, and Fu Siong Ng. "The cost of explainability in artificial intelligence-enhanced electrocardiogram models". In: *npj Digital Medicine* (2025).

[22] Maximilian Kapsecker, Matthias C. Möller, and Stephan M. Jonas. "Disentangled representational learning for anomaly detection in single-lead electrocardiogram signals using variational autoencoder". In: *Computers in Biology and Medicine* (2025).

[23] Anand Karthik Subramanian. *PyTorch-VAE: A collection of Variational Auto-Encoders implemented in PyTorch*. `https://github.com/AntixK/PyTorch-VAE`. Accessed: 2026-04-01. 2020.

A. Appendix

A.1 Additional Figures and Tables

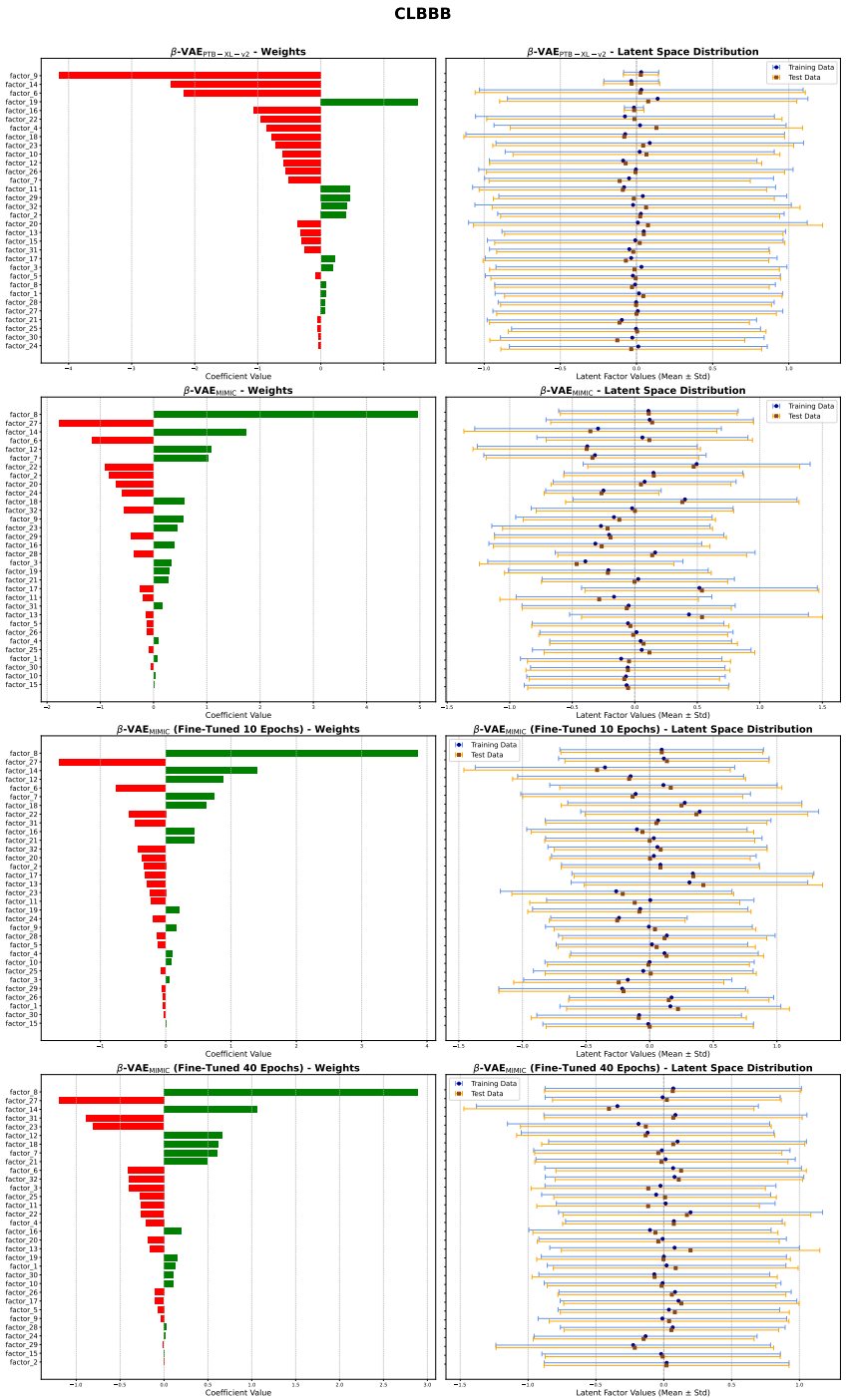

Figure 10: Logistic regression weights and latent space distributions for predicting CLBBB in the PTB-XL dataset across four different models. The right column illustrates the domain shift occurring when the β-VAE$_{MIMIC}$ model is applied to PTB-XL data, as the factor distributions deviate from $\mathcal{N}(0,1)$. But also how it becomes closer to $\mathcal{N}(0,1)$ as β-VAE$_{MIMIC}$ is fine-tuned.

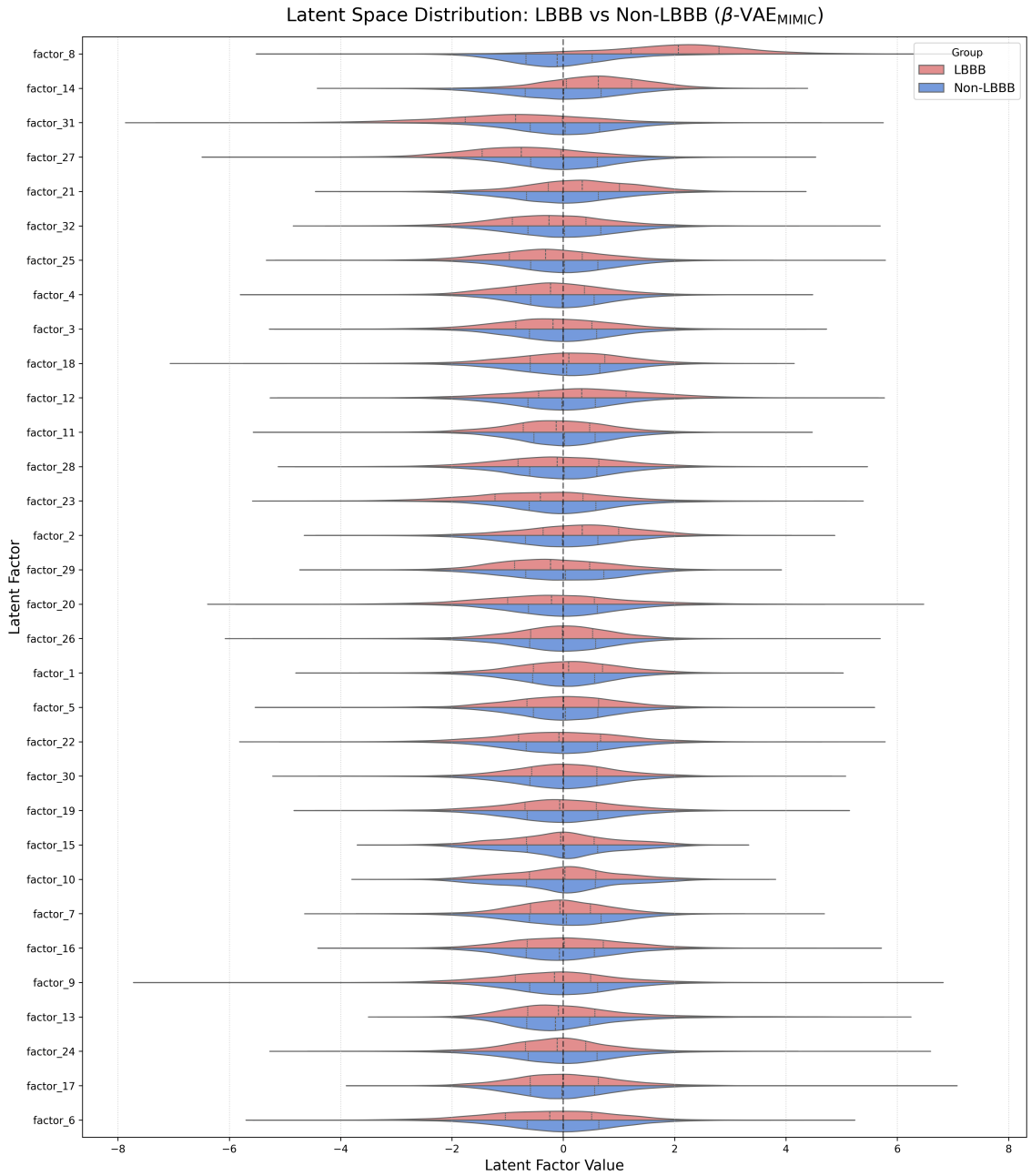

Figure 11: Violin plot showing the latent space distribution for LBBB versus non-LBBB patients. The factors are sorted by their absolute coefficient weight from the logistic regression model. Lines in the distribution corresponds to 25th quartile, 50th and 75th.

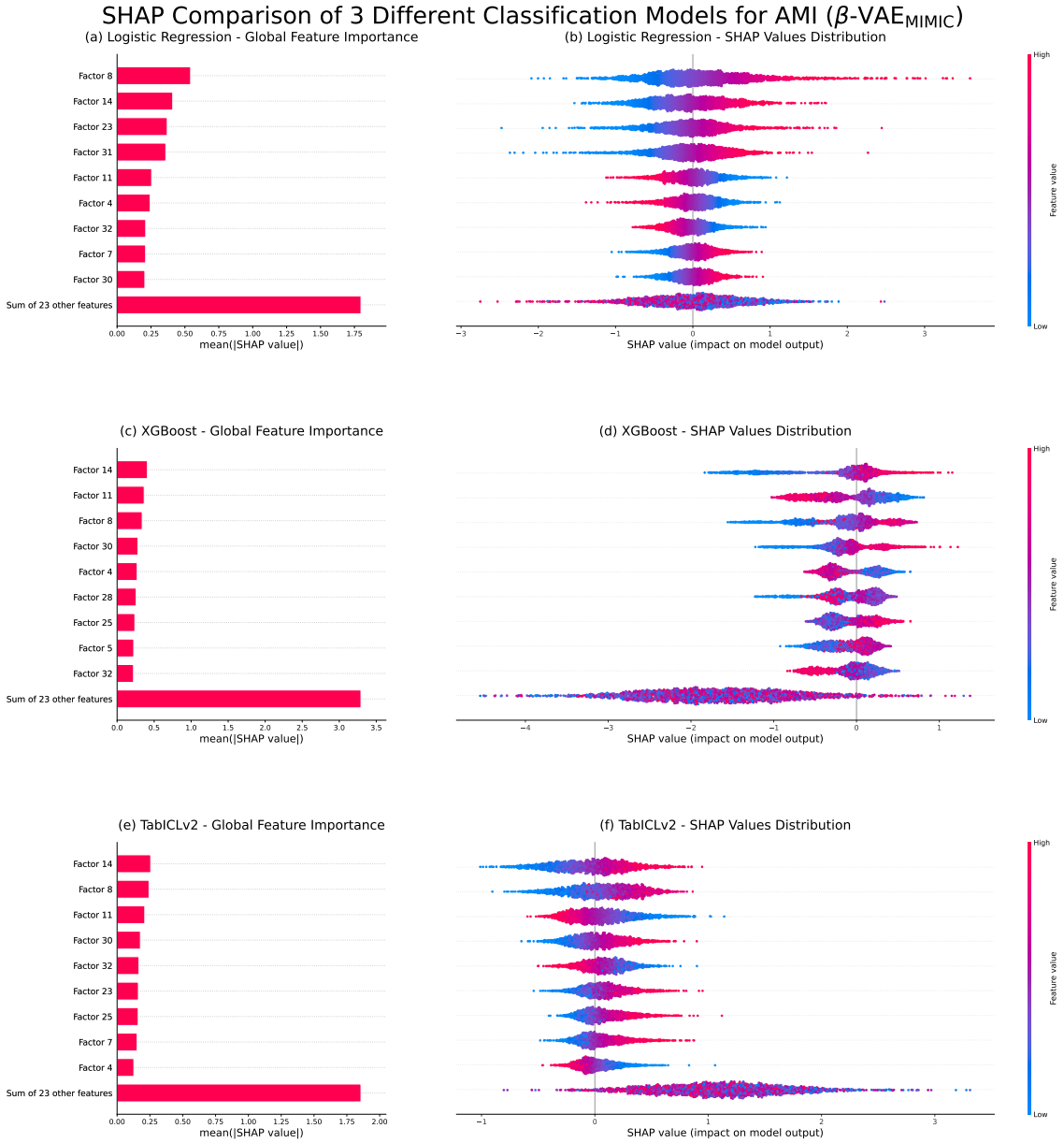

Figure 12: SHAP comparison of Logistic Regression, XGBoost, and TabICLv2 trained to predict AMI using the β-VAE$_{\text{MIMIC}}$ latent factors. Unlike the clear feature importance seen for CLBBB, the models here are highly uncertain. No single factor dominates the decision.

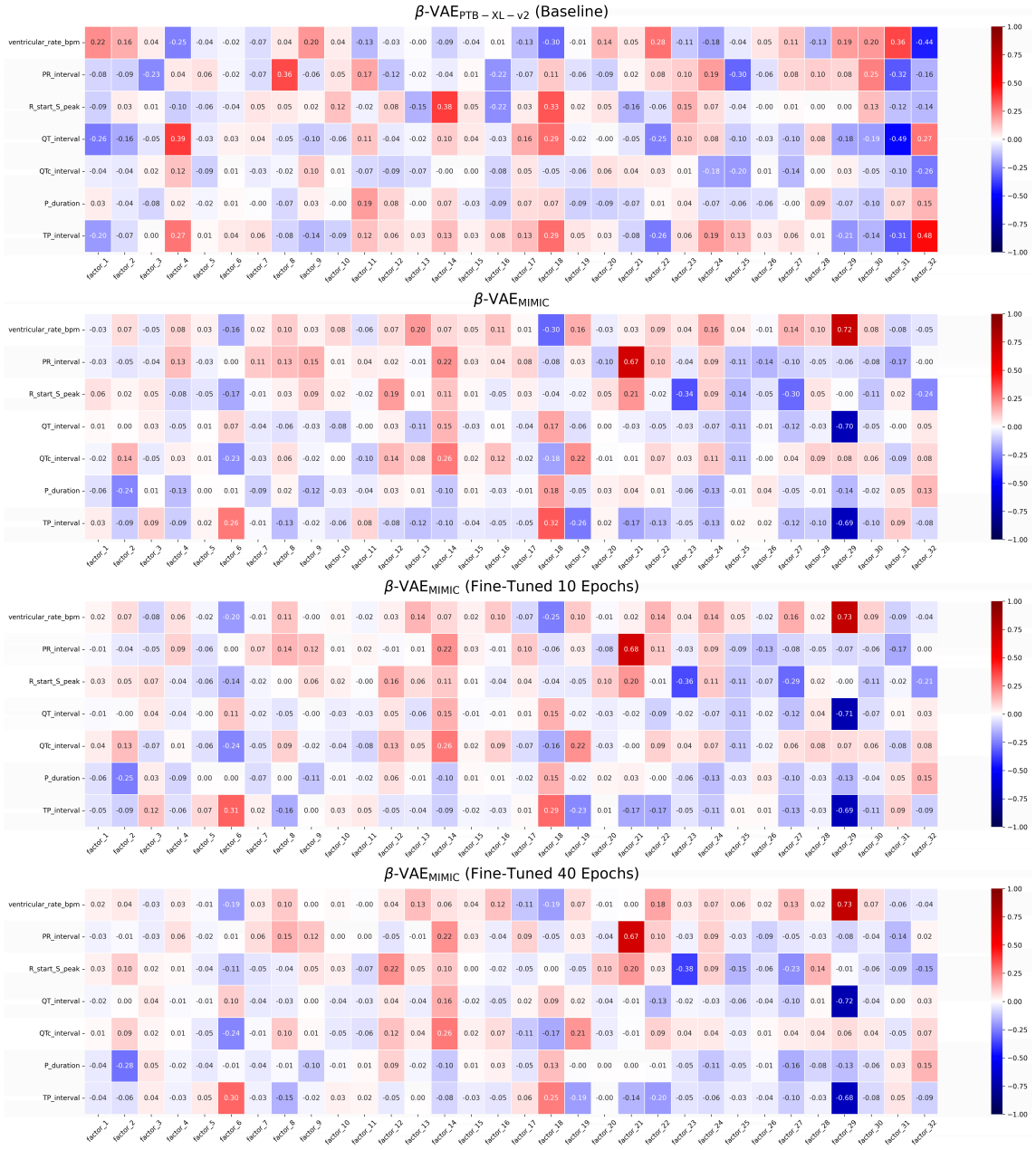

Figure 13: Comparison of latent space disentanglement across four models evaluated on the PTB-XL test set. Top: Baseline model. Second: β-VAE$_{MIMIC}$. Third and Bottom: β-VAE$_{MIMIC}$ fine-tuned for 10 and 40 epochs, respectively.

Model Architecture Configurations

Table 5 shows the Encoder and Decoder configurations used for the VAE models in this study. The fundamental layer structure is based on the implementation by van de Leur et al. [13].

Table 5: Architectural details and layer configurations for the VAE models.

Parameter	Value
Encoder (1D Causal CNN)	
Input Channels (Leads)	12
Depth (Convolutional Layers)	7
Internal Channels	128
Kernel Size	5
Dropout	0.3
Latent Space Dimension (K)	32
Decoder (Transposed 1D Causal CNN)	
Depth (Convolutional Layers)	7
Internal Channels	128
Kernel Size	5
Dropout	0.0
Output Width (Time Steps)	600

Table 6: Training specifications for β-VAE$_{MIMIC}$, β-VAE$_{12SL}$, β-VAE$_{PTB\text{-}XL}$ and β-VAE$_{PTB\text{-}XL\text{-}v2}$

Training Specifications	
Parameter	**Value**
Max Epochs	200
Batch Size	128
Optimizer	AdamW (lr = 0.001, weight decay = 0.01, $\epsilon = 10^{-4}$)
Gradient Clipping (max_norm)	1.0
LR Scheduler	ReduceLROnPlateau (factor = 0.5, patience = 5)
Early Stopping Patience	10 epochs
Reconstruction Loss `min_std`	0.1
KL Divergence Weight (β)	Sine schedule ($4 \rightarrow 32$ over the first 20 epochs)

Table 7: Final hyperparameter configuration for the XGBoost models.

Parameter	Value
Number of Trees (`n_estimators`)	1000
Learning Rate (`learning_rate`)	0.025
Max Depth (`max_depth`)	5
Subsample Ratio (`subsample`)	0.8
Objective	Binary Cross Entropy

A.2 Additional Interpretability Analysis

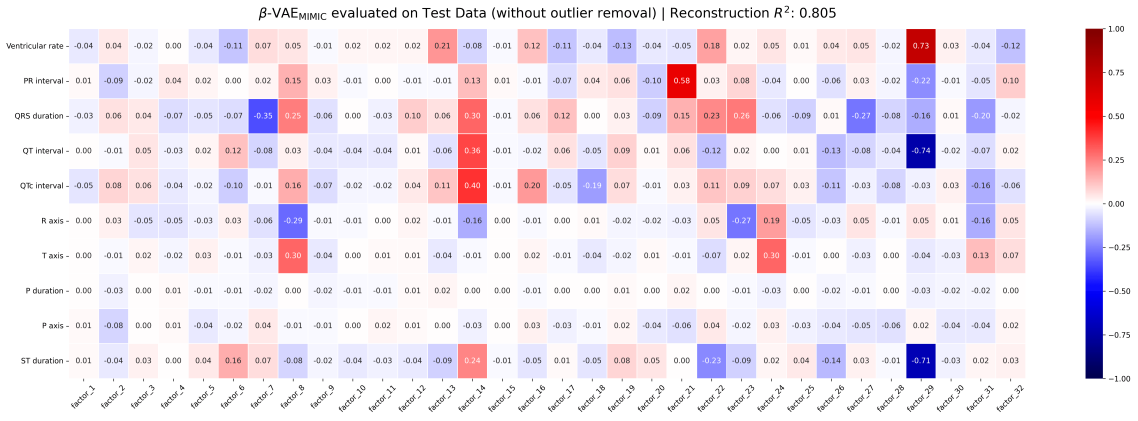

Figure 14: Correlation heatmap evaluating the test set without any statistical outlier removal applied to the ECG measurements. Compared to Figure 2, the correlations are generally weaker, demonstrating the difference made by the Hotelling's T^2 preprocessing step.

To further investigate a factor like Factor 8 and how the different classification models use these latent factors from the β-VAE$_{\text{MIMIC}}$ model, SHAP scatter plots were generated. These were generated with Factor 27 as comparison as this is the factor that has the most interaction with Factor 8 in the XGBoost model. Figure 15 illustrates the dependence of the model output on Factor 8.

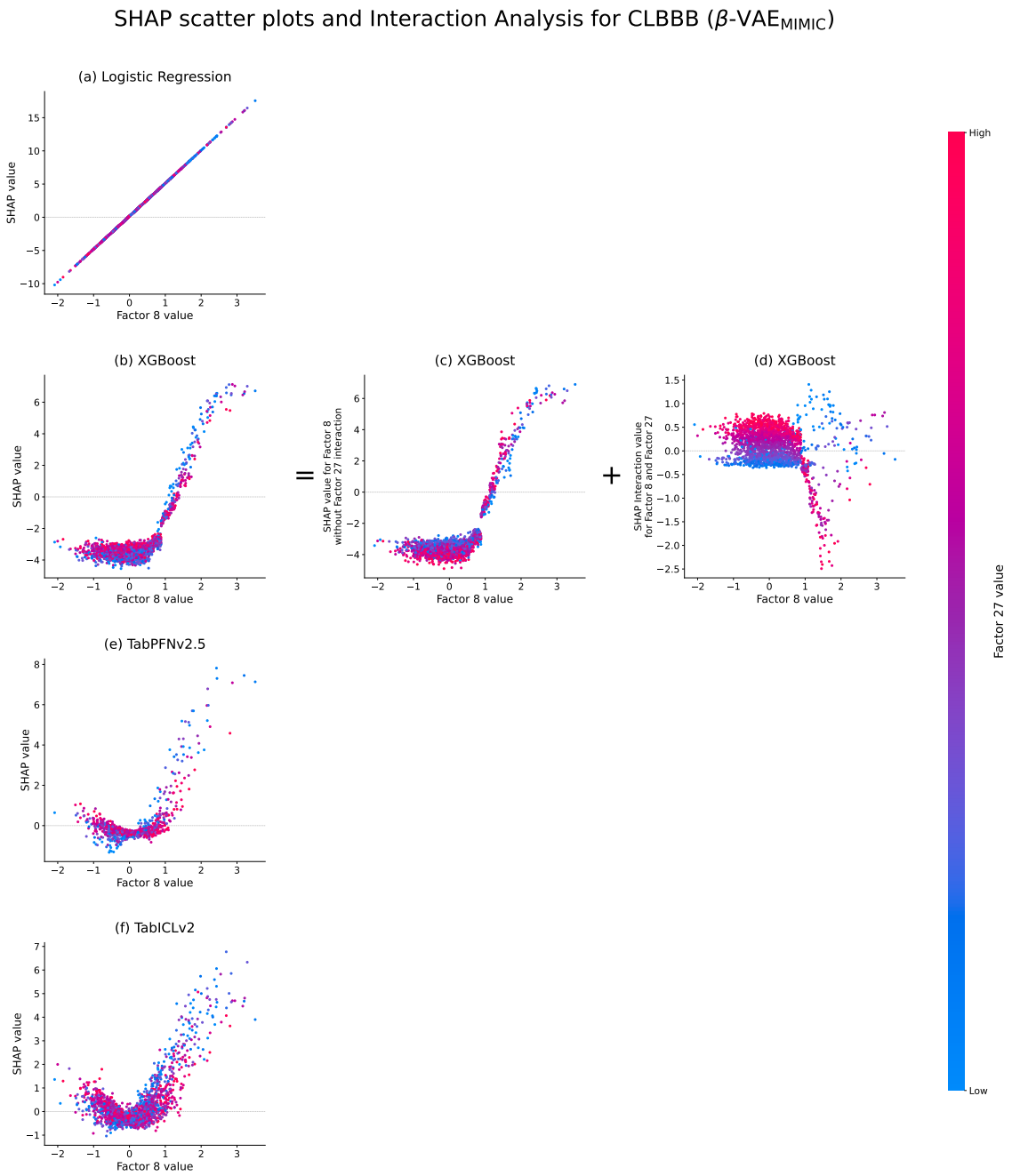

Figure 15: SHAP scatter plots and interaction analysis of four different classification models (Logistic Regression, XGBoost, TabPFNv2.5, and TabICLv2) trained to predict CLBBB using β-VAE$_{MIMIC}$ latent factors. Plot (a) shows that Logistic Regression has a linear dependence on Factor 8 confirming no interaction effects as expected. Plots (b, c, d) show how the SHAP value for Factor 8 in XGBoost (b) can be formalised into its interaction effect with Factor 27 (d) and the SHAP values without Factor 27 interaction (c). Plots (e, f) show the SHAP scatter plots for TabPFNv2.5 and TabICLv2 which show a non-linear relationship close to the XGBoost scatter plot. Note that for TabPFNv2.5 it only shows SHAP values for 950 out of the 2195 samples in the test data which is why it looks a little sparser.

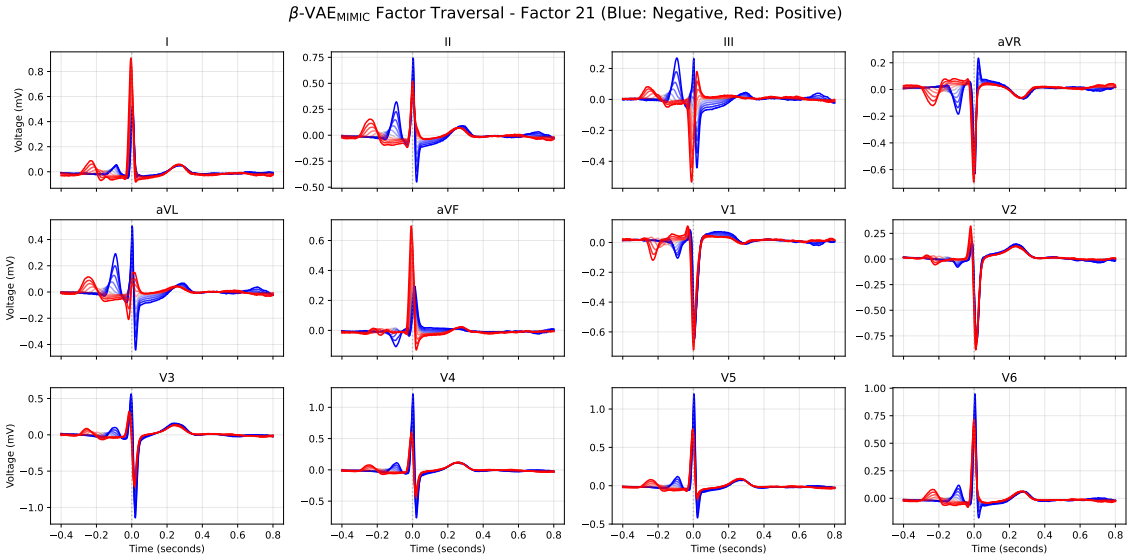

Figure 16: Factor traversal for β-VAE$_{MIMIC}$ Factor 21. By holding all other factors at zero and varying Factor 21 from -5 (blue) to +5 (red), a distinct temporal shift in the P-wave can be observed.

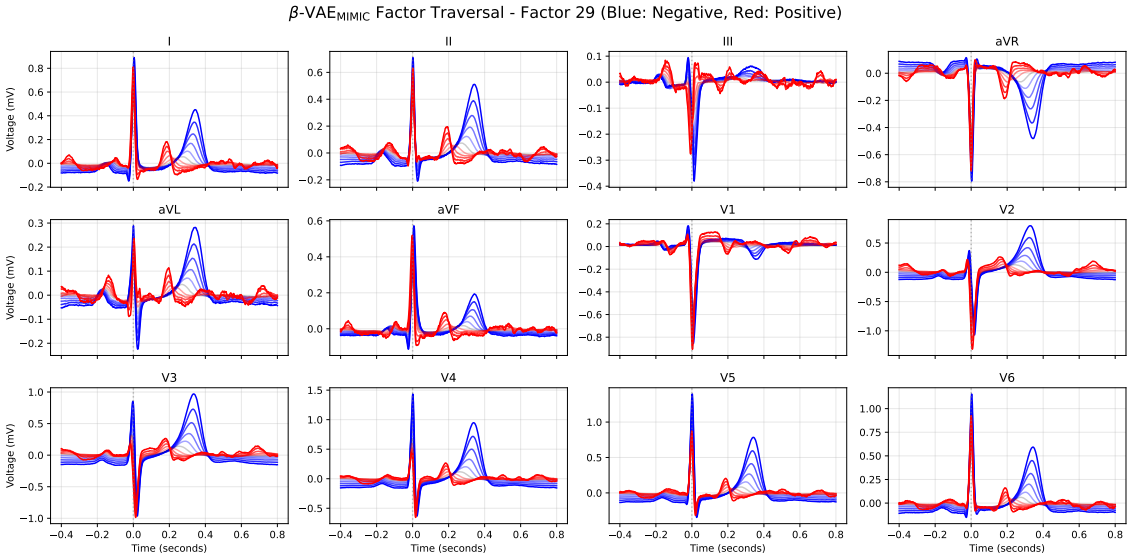

Figure 17: Factor traversal for β-VAE$_{MIMIC}$ Factor 29. As the factor increases (red), the T-wave occurs earlier and with lower amplitude (shorter QT interval).

A.3 Extended SHAP Overview Plots

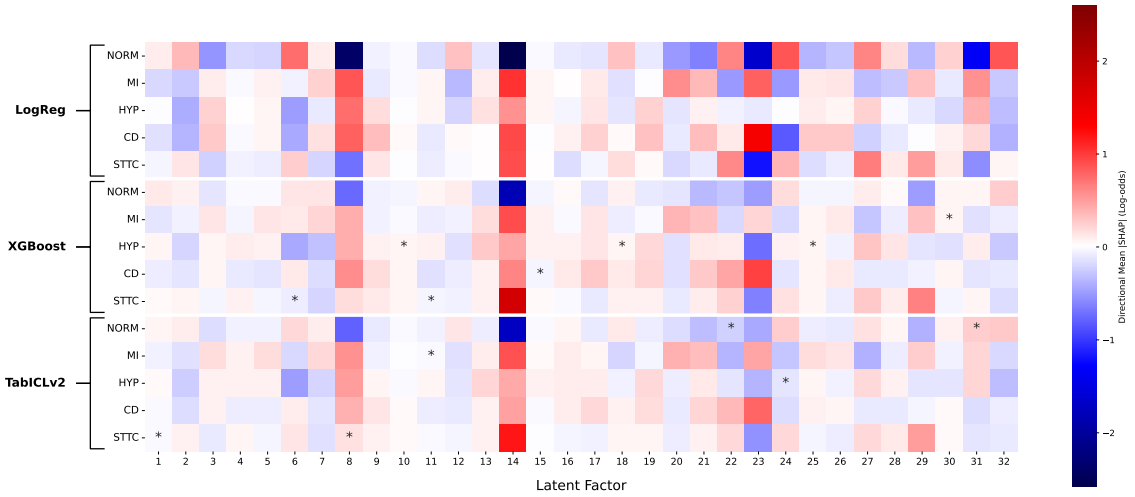

Figure 18: Directional SHAP importance across five diagnostic classes in the super-diag level, comparing Logistic Regression, XGBoost, and TabICLv2.

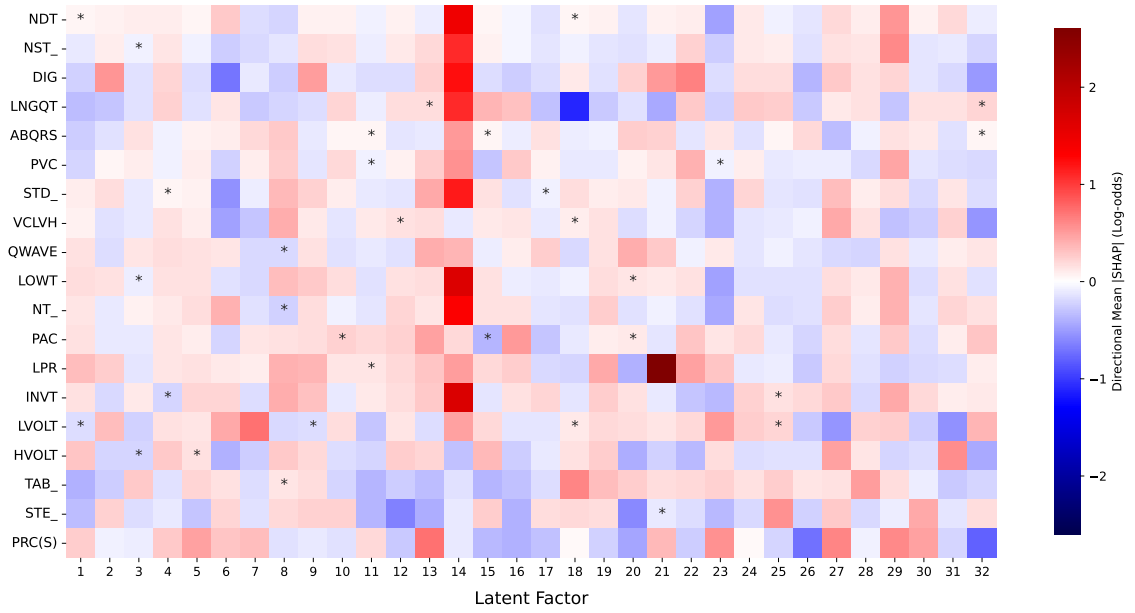

Figure 19: Directional SHAP importance across 19 diagnostic classes in the Form level using XGBoost.

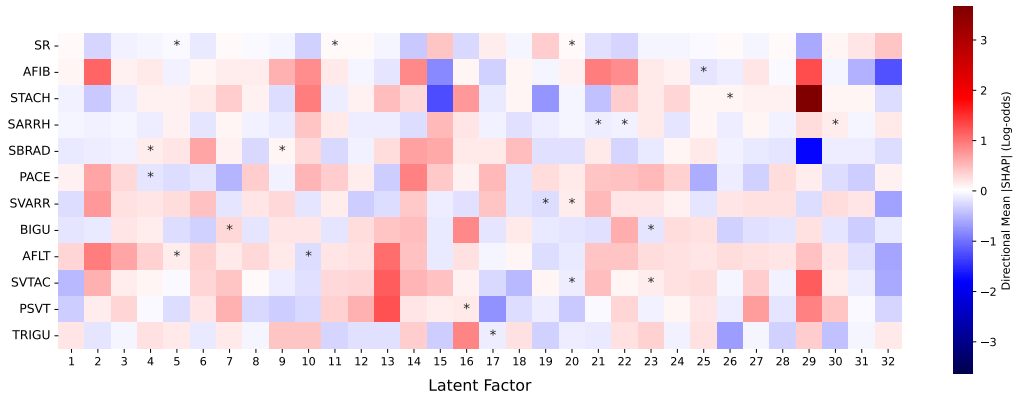

Figure 20: Directional SHAP importance across 12 diagnostic classes in the Rhythm level using XGBoost.

A.4 TabICLv2 Fine-Tuning Experiment

To evaluate if the performance of TabICLv2 could be improved with fine-tuning on PTB-XL discriminative performance, an experiment was conducted using the finetuned TabICL classifier.[18]. The training was conducted on the latent factors generated by the β-VAE$_{\text{MIMIC}}$ FT 40 model.

A small problem is that this fine-tuning cannot be done if there are not enough positive samples in a given class. This was the case for 2/44 classes in diag level and 1/19 classes in form layer. To deal with this we just use the AUC values from the non fine-tuned model for those. This should have a very minimal impact on the experiment. The hyperparameters used for the fine-tuning process are detailed in Table 8.

Table 8: Hyperparameter configuration for the TabICLv2 fine-tuning experiment.

Parameter	**Value**
Epochs	50
Learning Rate	1×10^{-5}
Estimators (Fine-tune)	2
Estimators (Validation)	2
Estimators (Inference)	8
Early Stopping	True
Patience	10 epochs
Evaluation Metric	ROC AUC

The discriminative performance of the fine-tuned model was compared against the non fine-tuned TabICLv2 model. As presented in Table 9, the fine-tuning process yielded only marginal differences, mostly in the third decimal place. This might be showing that there is not much more information that can be extracted from these latent factors.

Table 9: Comparison of non fine-tuned TabICLv2 and fine-tuned TabICLv2 using the β-VAE$_{\text{MIMIC}}$ FT 40 latent factors. Performance is reported in Macro-AUC and Macro-AUPRC across all evaluation levels.

	Macro-AUC		Macro-AUPRC	
Diagnostic Level	**Standard**	**Fine-tuned**	**Standard**	**Fine-tuned**
Diagnostic	0.9215	0.9227	0.3110	0.3103
Diagnostic Subclass	0.9271	0.9273	0.4550	0.4529
Diagnostic Superclass	0.9084	0.9095	0.7720	0.7743
Form	0.8690	0.8724	0.1748	0.1775
Rhythm	0.9062	0.9037	0.4474	0.4536
All	**0.9059**	**0.9069**	**0.3032**	**0.2996**

A.5 Comparison of ECG Measurement Extraction 12SL vs. NeuroKit2

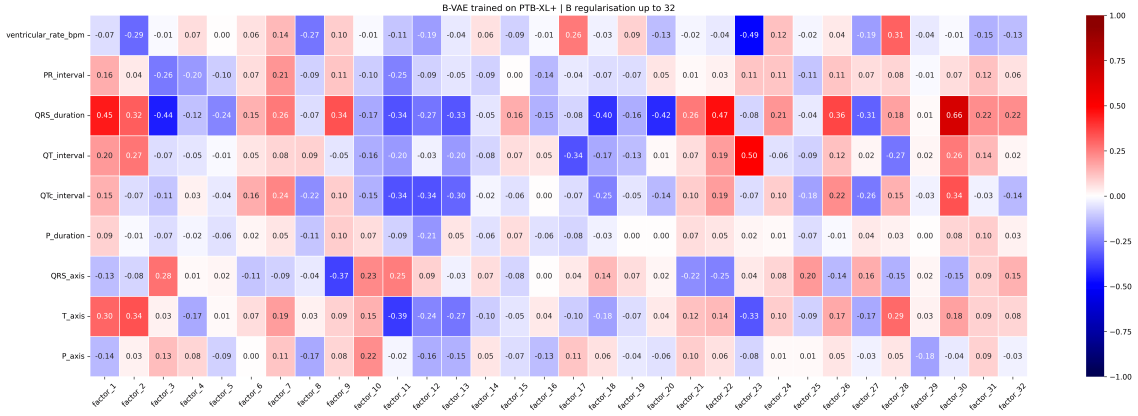

Figure 21: Correlation heatmap between latent factors and 12SL ECG measurements from PTB-XL+ for β-VAE$_{12SL}$

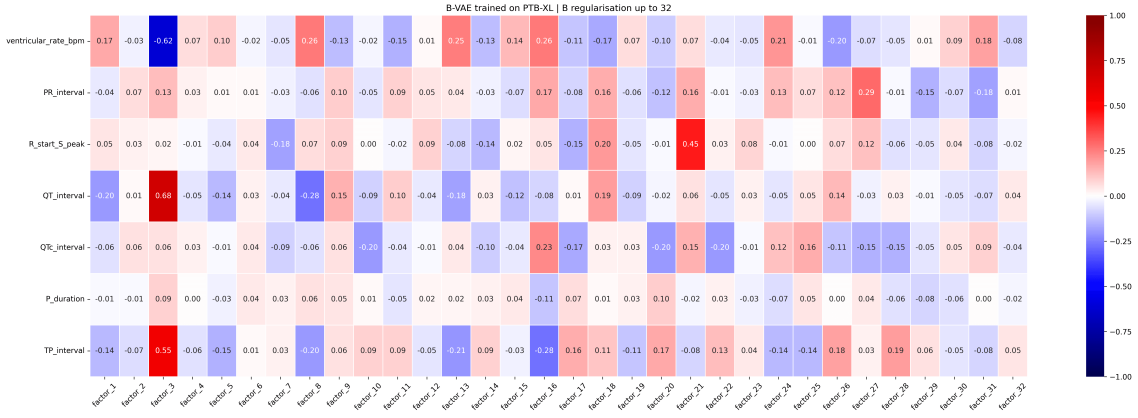

Figure 22: Correlation heatmap between latent factors and our created ECG measurements from PTB-XL for β-VAE$_{PTB\text{-}XL}$

To compare Figure 21 and Figure 22 we have to know that the rows to compare are the top 6 rows as these are the ones matching each other, except for row 3 that has QRS_duration for PTB-XL+ and R_start_S_peak for PTB-XL, this is because no reliable method to extract the QRS was found with NeuroKit2. The idea was that R_start_S_peak was the closest to QRS_duration that could be achieved.

When comparing these 6 rows we see that the most notable difference is just here on row 3 where we do not compare the exact same ECG measurement. Namely that PTB-XL+ QRS_duration seems less disentangled but also has a factor with higher correlation than R_start_S_peak has. Otherwise, structural similarities can be observed. For instance, ventricular rate and QT interval has the same factor that has high correlation just different signs, which makes sense because these measurements are related with each other. PR interval and P duration are both hard to get high correlation with for both models.

A.6 Additional Modeling Experiments

After the training of β-VAE$_{\text{MIMIC}}$, more hypertuning experiments were performed without much difference in disentanglement results. However, one change was promising, and that was using the median of the leads instead of the mean to fix baselines. In addition to this, other things that might improve performance were increasing the training data and using a β-TCVAE instead of just a β-VAE. And so, two new VAEs were trained. One β-VAE that we can call β-VAE$_{\text{MIMICv2}}$ was trained on all MIMIC-IV-ECG training data, not just the ones with a diagnosis. A new exclusion list was made with the same scripts as before, which resulted in the final training set of 659,741 samples with a median lead baseline fix, compared to the 338,257 that β-VAE$_{\text{MIMIC}}$ was trained on. The other new model that was tested was a β-TCVAE trained on the 338,257 original samples, but with a median lead baseline fix instead of the mean. Otherwise, the architecture and training specifications for these two models are the same as seen in Tables 5 and 6, just that for the β-TCVAE it was the TC term of the KL divergence that was sine scheduled from 4 to 32 over the first 20 epochs, and the other 2 regularization terms were set to 4 in this case. The amount of different hyperparameter tuning for the β-TCVAE was limited in this study, and certainly more can be tested, like having the other factors in the KL divergence increase during the training as well, as they tried in [22]. The implementation for calculating the decomposed KL divergence loss in the β-TCVAE model was adapted from the open-source repository by Subramanian [23].

In general, when training β-TCVAE models on MIMIC-IV-ECG, the observation was that although high correlations between factors and ECG measurements could be achieved, it was hard to get a model that had only one factor to have this high correlation; the correlation heatmaps seemed less interpretable. The β-VAE$_{\text{MIMICv2}}$, however, got a more disentangled correlation heatmap and beat the record for all the VAEs that were trained in this study for the highest correlation value between a latent factor and an ECG measurement. It got -0.90 on the QT interval! These two new models were tested on the PTB-XL dataset as well without any fine-tuning, to compare the classification performance to what β-VAE$_{\text{MIMIC}}$ got. The overall results were that the new models performed noticeably better on rhythm types of diagnoses, but unexpectedly worse on other levels of diagnoses, which resulted in overall no major performance difference between these two new models and β-VAE$_{\text{MIMIC}}$. The details can be seen in Table 10.

Table 10: Discriminative performance of non fine-tuned models on the PTB-XL test set using TabICLv2. The table compares the original model (β-VAE$_{MIMIC}$), the model trained on the full dataset with median centering (β-VAE$_{MIMICv2}$), and the Total Correlation VAE (β-TCVAE$_{MIMIC}$).

	Macro-AUC			Macro-AUPRC		
Diagnostic Level	**Original**	**MIMICv2**	**TCVAE**	**Original**	**MIMICv2**	**TCVAE**
Diagnostic (diag.)	**0.9191**	0.9130	0.9063	0.3074	**0.3090**	0.2879
Subclass (sub diag.)	**0.9216**	0.9145	0.9103	0.4514	**0.4580**	0.4283
Superclass (super-diag.)	**0.9057**	0.9027	0.9024	**0.7706**	0.7590	0.7569
Form	0.8630	**0.8638**	0.8617	0.1737	**0.1855**	0.1850
Rhythm	0.8993	0.9206	**0.9233**	0.4482	0.4395	**0.4663**
All	0.9014	**0.9015**	0.8975	0.3000	**0.3019**	0.2929

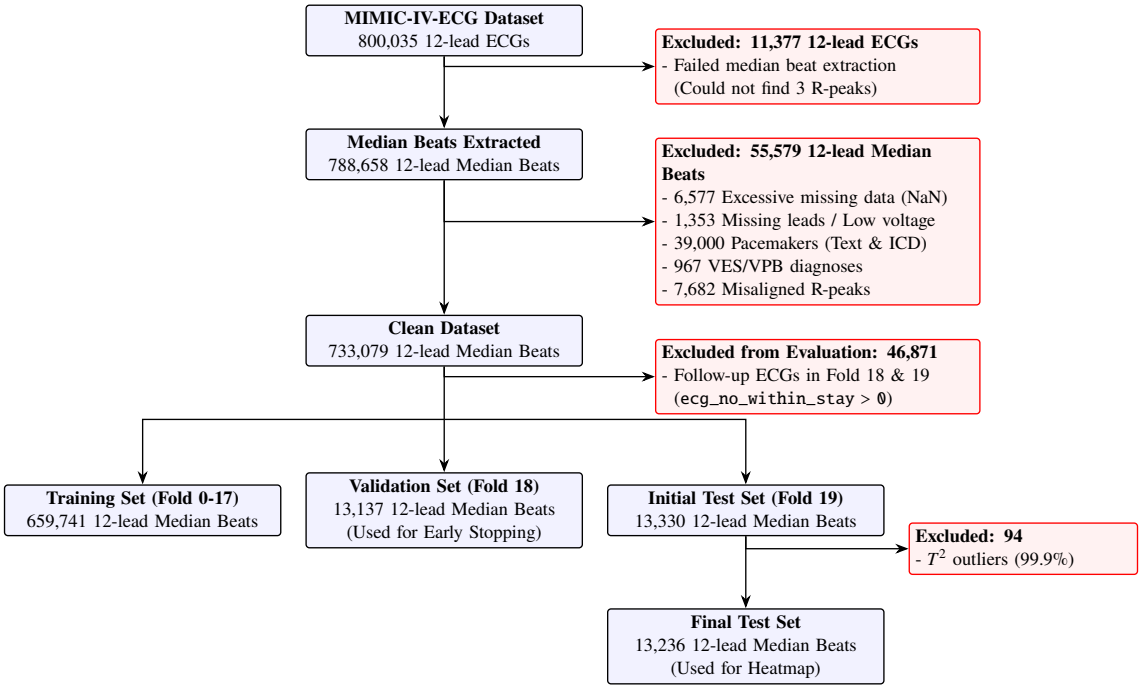

Figure 23: Data exclusion and dataset splitting flowchart for the β-VAE$_{MIMICv2}$ model. Unlike the β-VAE$_{MIMIC}$ model, training was performed on all available data without requiring diagnostic labels and strict fold separation was enforced to prevent patient leakage.

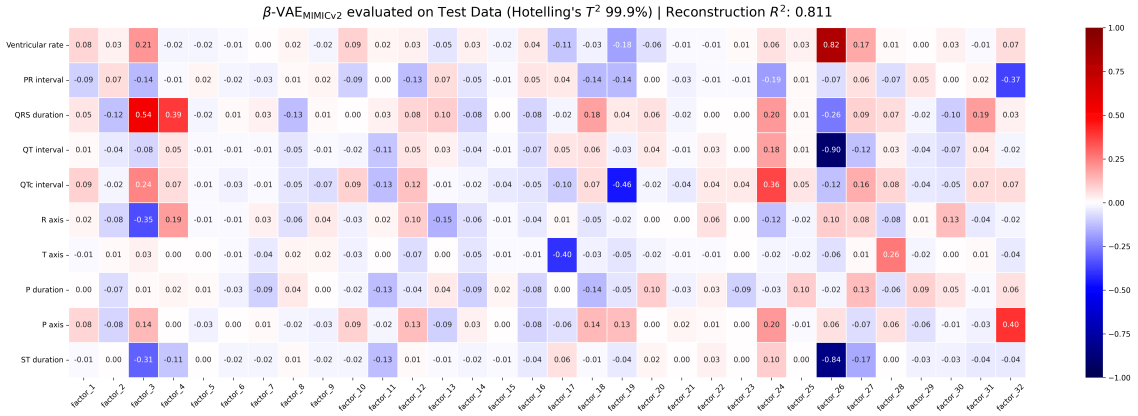

Figure 24: Correlation heatmap evaluating β-VAE$_{MIMICv2}$ latent factors on the final test set seen in the flow chart above.