# Deep Learning Methods Cannot Outperform Other Machine Learning Methods on Analyzing Genome-wide Association Studies

by

Shaoze Zhou
B.Sc., University of Victoria, 2015

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Mathematics and Statistics

Deep Learning Methods Cannot Outperform Other Machine Learning Methods on Analyzing Genome-wide Association Studies

by

Shaoze Zhou
B.Sc., University of Victoria, 2015

Supervisory Committee

Dr. Xuekui Zhang, Co-supervisor
(Department of Mathematics and Statistics)

Dr. Min Tsao, Co-supervisor
(Department of Mathematics and Statistics)

**Supervisory Committee**

Dr. Xuekui Zhang, Co-supervisor
(Department of Mathematics and Statistics)

Dr. Min Tsao, Co-supervisor
(Department of Mathematics and Statistics)

## ABSTRACT

Deep Learning (DL) has been broadly applied to solve big data problems in biomedical fields, which is most successful in image processing. Recently, many DL methods have been applied to analyze Genome-wide Association studies (GWAS). However, genomic data usually has too small a sample size to fit a complex network. They do not have common structural patterns like images to utilize pre-trained networks or take advantage of convolution layers. The concern of overusing DL methods motivates us to evaluate DL methods' performance versus popular non-deep Machine Learning (ML) methods for analyzing genomic data with a wide range of sample sizes.

In this paper, we conduct a benchmark study using the UK Biobank data and its many random subsets with different sample sizes. The original UK Biobank data has about 500k participants. Each patient has comprehensive patient characteristics, disease histories, and genomic information, i.e., the genotypes of millions of Single-Nucleotide Polymorphism (SNPs). We are interested in predicting the risk of three lung diseases: asthma, COPD, and lung cancer. There are 205,238 participants have recorded disease outcomes for these three diseases. Five prediction models are investigated in this benchmark study, including three non-deep machine learning methods (Elastic Net, XGBoost, and SVM) and two deep learning methods (DNN and LSTM). Besides the most popular performance metrics, such as the F1-score, we promote the hit curve, a visual tool to describe the performance of predicting rare events.

We discovered that DL methods frequently fail to outperform non-deep ML in analyzing genomic data, even in large datasets with over 200k samples. The experiment results suggest not overusing DL methods in GWAS studies, even with biobank-level sample sizes. The performance differences between DL and non-deep ML decrease as the sample size of data increases. This suggests when the sample size of data is significant, further increasing sample

sizes leads to more performance gain in DL methods. Hence, DL methods could be better if we analyze genomic data bigger than this study.

# Table of Contents

# List of Tables

# List of Figures

# Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisors, Professor Xuekui Zhang and Professor Min Tsao, for their support of my research and study at University of Victoria in the past years, for their patience, inspirations and technical advice. I could not having imagined having better advisors and mentors for my master study.

Besides my supervisor, I would like thank Professor Yao Dong, Professor Li Xing, Professor Yongfeng Dong, Yumeng Chen, and Ziyu Ren for their help on my research study and for their valuable comments, insights, and guidance.

Last and certainly not least, I would like to thank my parents for their endless love and support throughout my life.

*Shaoze Zhou*

# Chapter 1

# Introduction

Machine Learning (ML) has been widely applied in genomic analysis and disease prediction. ML is considered an objective and reproducible method that integrates multiple quantitative variables to improve diagnostic accuracy [1]. There are many successful applications. In disease prediction, Deberneh and Kim [2] presented an ML model for predicting the T2D (type 2 diabetes ) occurrence in the following year (Y+1) using variables in the current year (Y). The model's performance proved to be reasonably good at forecasting the occurrence of T2D in the Korean population. Park and Lee [3] constructed a disease recurrence prediction model using ML techniques. Their study compared the performance of five ML models( decision tree, random forest, eXtreme Gradient Boosting [XGBoost], LightGBM, and Stacking models) related to recurrence prediction based on accuracy, and the Decision Tree model showed the best accuracy at 95%. In another study, Hussain et al. [4] proposed a voting ensemble classifier with 24 features to identify the severity of chronic obstructive pulmonary disease (COPD) patients. Five ML classifiers were applied, namely random forests (RF), support vector machine (SVM), gradient boosting machine (GBM), XGBoost, and K-nearest neighbor (KNN) in their study. These classifiers were trained with a set of 24 features. After that, they combined the results with a soft voting ensemble (SVE) method. The results showed that the SVE classifier outperforms conventional ML-based methods for patients

with COPD. In addition, ML-based methods for genetic analysis have also been reported in multiple studies [5, 6], such as ML approaches for the prioritization of genomic variants impacting Pre-mRNA splicing; ML suggests the polygenic risk for cognitive dysfunction in amyotrophic lateral sclerosis and so on.

Deep Learning (DL) is a subset of ML, and it goes beyond non-deep ML by creating more complex multi-layer models to mimic how humans function. DL is known to work well in big data applications. Still, DL has been used in disease prediction primarily based on publicly available medical image data, which have common structural patterns to utilize pre-trained networks or take advantage of convolution layers. For example, Chao et al. [7] presented a DL CVD risk prediction model, which was trained with 30,286 LDCTs from the National Lung Cancer Screening Trial. As a result, the model obtained an area under the curve (AUC) of 0.871 on a separate test set of 2085 subjects and was able to identify patients at high risk of CVD mortality (AUC of 0.768). Zhou et al. [8] proposed a DL model to classify the HCM genotypes based on a non-enhanced four-chamber view of cine images. Lin et al. [9] developed and validated a DL algorithm for detecting coronary artery disease (CAD) based on facial photos. Jin et al. [10] presented a multi-task deep learning approach that allows simultaneous tumor segmentation and response prediction. Their approach to capturing dynamic information in longitudinal images may be broadly used for screening, treatment response evaluation, disease monitoring, and surveillance.

However, compared with image data, genomic data has less structure information to train a DL model. Moreover, building an accurate DL model usually requires immense amounts of data, which is often difficult to find in biological studies with a limited number of participants. Therefore, we are motivated to investigate the effectiveness of DL in genomic analysis and the amount of genomic sample size fitting for the DL model.

Our study explores and compares three non-deep ML and two DL methods in genomic analysis, including elastic net, XGBoost, SVM, long short-term memory (LSTM), and deep

neural network (DNN). These methods are applied to the UK Biobank study, which includes a wide array of genotypic and phenotypic information from 502,524 participants. Coupled with the current impact of COVID-19, lung diseases have attracted widespread attention. We choose three specific lung diseases from UK Biobank, combined with SNPs and other relevant covariates to build prediction models with these five typical non-deep ML and DL algorithms. Large-scale computation works are conducted using high-performance computing servers provided by Compute Canada. To investigate how DL and non-deep ML methods perform in genomic analysis on various sample sizes, we generate random subsets of original data with 10 different levels of sample size and evaluate the prediction performance of each method using multiple metrics, including F1 score, precision, recall, and the hit curve. Besides comparing DL and non-deep ML methods, we also investigated the relation between performance change and other important factors, such as sample sizes increase and the imbalanced ratio (defined as the proportion of samples in the number of a control group to the number of case group [11].

The rest of the paper is organized as follows. Chapter 2 and 3 provides detailed processing and summary statistics of the dataset from UK Biobank, and five DL and non-deep ML methods are discussed in detail. In Chapter 4, experiment results are presented and compared. Concluding remarks are given in Chapter 5.

# Chapter 2

# Data and Methods

The workflow diagram is shown in Figure 2.1.

## 2.1 Data

With the rapid spread of COVID-19, lung diseases have attracted widespread social attention. It was suggested that the presence of lung diseases, in general, may contribute to severe COVID-19 symptoms. About 600 million people have asthma, and lung cancer and COPD are the first and the third leading cause of death worldwide. Genetic variants such as single nucleotide polymorphisms (SNPs) have been focused on in lung disease research.

The dataset we use in our study is the release of the 2018 UK Biobank. The original dataset has collected a wide array of phenotypic and phenotypic information from 502,524 participants. We only select three specific lung diseases (i.e. asthma, COPD, and lung cancer), combined with participants' SNPs, sex, body mass index (BMI), age, smoking status, and Z-score of the forced expiratory volume in one second (FEV1Z).

### 2.1.1 Genotype and Quality Control Procedure

Quality control and imputation were performed centrally by UK Biobank. We exclude the following participants from our analyses: (1) participants not of white British ancestry either by self-report or principal component analysis conducted by UK Biobank, (2) participants with more than 10% missing genotype data, (3) participants with putative sex-chromosome aneuploidy, (4) participants where the self-reported sex does not match the genetically-inferred sex, (5) participants that UK Biobank has flagged for having high heterozygosity/missingness and (6) participants with at least ten putative 3rd-degree relatives. Further, we remove SNPs with imputation information score $< 0.1$, minor allele frequency $< 0.001$, more than 5% missing genotype data, p-value $< 10^{-6}$ in the Hardy-Weinberg Equilibrium test, and SNPs that fail UK Biobank quality control in at least one batch. After sample filtering and SNP screening, we are left with a sample size of 205,238 participants and 2,000 SNPs.

### 2.1.2 Data Statistics

The average age of subjects is 56.5 years, with an age range of 40–69 and a sex ratio (females/males) of 1.35. The selected features are BMI, sex, age, Smoking status, FEV1Z, and 2,000 SNPs information. The summary of data is shown in Table 2.1. To explore the model performance and the prediction effect of DL and non-deep ML in the case of large and small data, we randomly generate ten subsets from 10% to 100% and repeat it ten times. The detailed subset information is shown in the Appendix A (Table A.1, A.2, A.3, and A.4).

Table 2.1: Descriptive statistics of the dataset. This table gives the relationships between smoking status and other covariates, i.e., age, sex, BMI, FEV1Z score, asthma status, COPD status, and lung cancer status.

| **Covariates** | **Never smoked** | **Previously smoked** | **Currently smokes** |
|---|---|---|---|
| **Age** | | | |
| $<$ 55 years | 47137(42.1%) | 22112(29.3%) | 8269(46.6%) |
| $\geq$ 55 years | 64826(57.9%) | 53414(70.7%) | 9480(53.4%) |
| **Sex** | | | |
| Male | 69300(61.9%) | 39670(52.5%) | 8912(50.2%) |
| Female | 42663(38.1%) | 35856(47.5%) | 8837(49.8%) |
| **BMI_mean** | 27.00($\pm$4.67) | 27.83($\pm$4.68) | 26.93($\pm$4.65) |
| **FEV1Z_mean** | 0.31($\pm$1.05) | 0.44($\pm$1.10) | 0.85($\pm$1.17) |
| **Asthmastatus** | 15110(13.5%) | 10343(13.7%) | 2239(12.6%) |
| **COPDstatus** | 1350(1.2%) | 3338(4.4%) | 1761(9.9%) |
| **Cancerstatus** | 185(0.17%) | 627(0.83%) | 390(2.2%) |

## 2.2 Methods

### 2.2.1 Elastic Net

In general, the elastic net is the regularized linear regression method [12]. It is a middle ground between ridge regression and lasso regression. The penalty term is a simple mix of ridge and lasso's penalties, and the mix ratio can be controlled. The estimates from the elastic net method are defined by

$$\hat{\beta} = \arg\min_{\beta}(\|y - X\beta\|^2 + \lambda_2\|\beta\|_2^2 + \lambda_1\|\beta\|_1), \tag{2.1}$$

where $\lambda_1\|\beta\|_1$ and $\lambda_2\|\beta\|_2^2$ are the $L_1$ norm and $L_2$ norm, respectively, $y$ is the response variable vector, and $X$ is covariates vector. The relationship between $\lambda_1$ and $\lambda_2$ can be written as

$$\lambda_1 = \alpha\lambda, \tag{2.2}$$

and

$$\lambda_2 = \frac{(1-\alpha)}{2}\lambda. \tag{2.3}$$

When the mix ratio $\alpha$ approaches 0, the elastic net is equivalent to ridge regression, and as the ratio $\alpha$ goes to 1, it is equal to lasso regression. As a result of balancing the L1 norm and L2 norm, the computational cost of the elastic net is expensive. However, it reduces the impact of different features while not eliminating all of the features to improve the model performance.

In this study, Elastic net models are implemented by R. The parameters $\alpha$ and $\lambda$ are tuned and chosen by function cv. glmnet().

### 2.2.2 XGBoost

XGBoost is an optimized distributed gradient boosting library designed to be efficient, flexible, and portable. It implements the algorithms in the Gradient Boosting framework, which integrates many weak classifiers to form a strong classifier [13]. The weak classifiers compensate each other to improve the performance of the strong classifier.

Unlike the traditional integrated decision tree algorithm, XGBoost adds a regular term in the loss function to control the complexity of the model while preventing the model from overfitting. The objective function is defined by

$$F(x) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k), \tag{2.4}$$

where $l(y_i, \hat{y}_i)$ is the model's loss function, $\Omega(f_k)$ is the regular term, $n$ is the number of samples, and $K$ is the number of the CART tree. After that, a second-order Taylor expansion approximation is applied to the loss function, and the objective function is optimized to approach the actual value and improve the prediction accuracy.

GridSearchCV function is used to find the optimal parameters. The parameter max_depth

of the XGBoost model is set to 5. The larger the max_depth, the more specific and local samples the model learns. The min_child_weight determines the minimum sum of instance weight needed in a child, and its value is 4. The parameter subsample is 0.8, which controls the proportion of random samples for each tree. The parameter colsample_bytree is used to manage the percentage of columns sampled per randomly sampled tree (each column is a feature), and its value is 0.8. The objective parameter defines the loss function that needs to be minimized. Reg_alpha and reg_lambada are the L1 regularization terms of the weights and the L2 regularization terms of the weights, respectively. These two parameters help reduce overfitting, and their values are 60 and 2, respectively.

### 2.2.3 SVM

SVM is a supervised learning algorithm. The learning strategy uses supporting vectors and margins to find the optimal segmentation hyperplane to classify the data[14]. SVM can be used for classification and regression analysis. As a training algorithm, SVM has a highly accurate and strong generalization ability.

This study uses the LinearSVC module in SVM. LinearSVC implements a linear classification support vector machine and can choose a variety of penalty parameters and loss functions. Normalization also works well when the number of training set instances is large.

We add the regularization term L1 norm to reduce the impact of overfitting. The parameter C of the LinearSVC model is 1.0.

### 2.2.4 LSTM

LSTM is a recurrent neural network (RNN). It can solve the problem of gradient disappearance and gradient explosion in traditional RNN. LSTM consists of a forget gate, an input gate, and an output gate[15]. The input vector and output vector of the hidden layer of

LSTM are $x_t$ and $h_t$, and the forward propagation process can be used in equations (5)-(9).

The input gate is mainly used to control how many values of the current input will flow directly to a memory unit, defined as follows:

$$i_t = \sigma(W_{x_i} x_t + W_{h_i} h_{t-1} + b_i). \tag{2.5}$$

The forget gate is an essential component of the LSTM memory cell, which controls the retention and forgetting of information to avoid gradient disappearance and gradient explosion caused by the backward propagation of gradients over time. The value of the forget gate $f_t$ and the value of the memory cell $c_t$ is expressed as:

$$f_t = \sigma(W_{x_f} x_t + W_{h_f} h_{t-1} + b_f) \tag{2.6}$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tanh(W_{x_c} x_t + W_{h_c} h_{t-1} + b_c). \tag{2.7}$$

The role of the output gate is to effectively control the effect of a memory processing unit on the input and output values in these messages. The value of the output gate $o_t$ and the output $h_t$ of LSTM at moment $t$ are expressed as:

$$o_t = \sigma(W_{x_o} x_t + W_{h_o} h_{t-1} + b_o) \tag{2.8}$$

$$h_t = o_t \otimes \tanh(c_t). \tag{2.9}$$

We construct a network structure with three LSTM layers and one dense layer and use sigmoid as the activation function and binary cross-entropy as the loss function. We set batch size and epoch as 128 and 100, respectively, and the learning rate is 0.0005.

### 2.2.5 DNN

A deep neural network (DNN) is a framework of deep learning. It is a neural network with at least one hidden layer [16], which can also be called a multi-layer perceptron.

For the DNN model, We divide it into the input layer, hidden layer, and output layer. Since we are exploring the classification and prediction of these three diseases, we choose $binary_crossentropy$ as our loss function. Secondly, we put three total connection layers into the hidden layer. The number of neurons in the hidden layer is set to 64. Each neuron in the top connection layer is fully connected with all neurons in the previous layer, which can integrate the local information with category differentiation in each layer. To improve the network performance of DNN, we applied the ReLU function to the activation function of each neuron.

Meanwhile, we found through experiments that when the batch size was set to 128, the model's accuracy could be effectively improved, and the model could converge more accurately towards the direction where the extreme value was. Moreover, when the epoch was 200 iterations, the training results tended to be stable basically. Although the model performance is improved, it is more prone to overfitting due to many parameters. Therefore, we added a regularization term L1 norm to constrain training parameters by adding a penalty norm for training parameters to the loss function to prevent model overfitting.

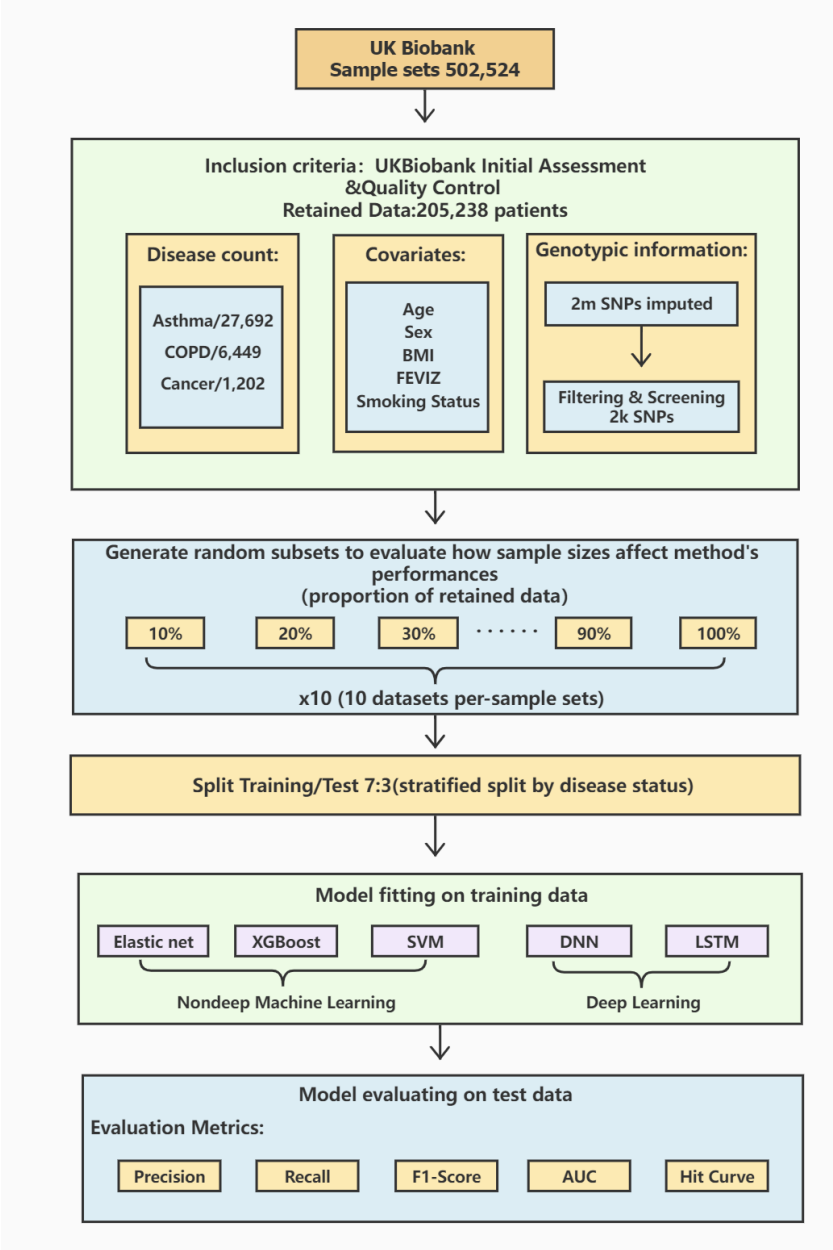


Figure 2.1: A workflow diagram of the study process.We perform data preprocessing on 502,524 sample sets from UK Biobank. After the initial assessment and quality control, the data is retained for 205,238 cases with detailed procedures in Section 2.1. There are 27,692 asthma cases, 6,449 COPD cases, and 1,202 lung cancer cases. Age, sex, BMI, FEVIZ, and smoking status are covariates. 2,000 SNPs are retained after filtering and screening the original 2 million SNPs. The retained dataset was divided into ten subsets per-sample sets from 10% to 100%. We split the data by disease status into 70% as training and 30% as testing sets. This study uses three non-deep ML models (Elastic net, XGBoost, and SVM) and two DL models (DNN and LSTM) to construct the prediction models. Finally, the model performance is evaluated by the metrics, such as precision, recall, F1-score, AUC, and hit curve.

# Chapter 3

# Results

In this study, five disease prediction models based on non-deep ML and DL models, i.e. elastic net, XGBoost, SVM, LSTM, and DNN, are constructed. The original dataset is randomly selected into ten sets of 10%-100% datasets (shown in Appendix A, Table A.1, A.2, A.3, and A.4). In the modelling process, we perform ten cross-validations on each set of 10%-100% datasets to find the optimal threshold for prediction and apply it to the test set. Finally, the mean and the standard deviation of the accumulated AUC, precision, recall, F1-score values, and hit curve plot are used as evaluation metrics. The detailed statistics are described in Appendix A (Table A.5, A.6, and A.7).

As shown in Figure B.22, the proposed models are evaluated by the standard metrics of precision, recall, and F1-score, and an increasing trend is generally discovered. Precision is the proportion of positive predictions that are actually correct. Recall is the proportion of actual patients identified correctly. The F1-score is the harmonic mean of precision and recall, and is often used to interpret imbalanced data. However, AUC is not sensitive to imbalanced data (AUC results are shown in the Appendix A,Table A.5, A.6, and A.7)). Hence, we are more interested in precision, recall, and F1-score due to the imbalanced data structure.

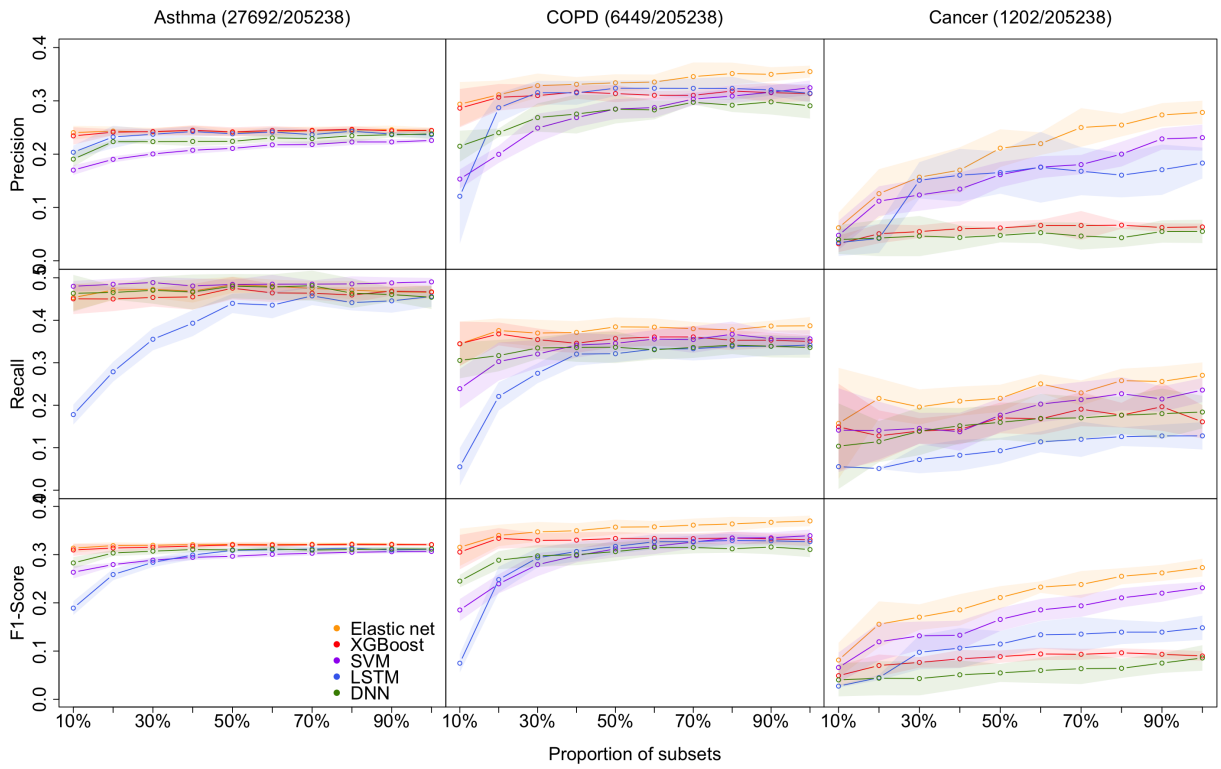


Figure 3.1: Models performance of the 5 methods with the 10 different sample size for predicting asthma, COPD, and lung cancer, respectively. Performances are shown by precision, recall, and F1-score. the shaded parts are the 1 standard error confidence bounds.

## 3.1 Performance on small-sized datasets

**Asthma status prediction.** For the 10% dataset, the highest precision and F1-score are 0.2404($\pm$0.0127) and 0.3135($\pm$0.0098), respectively, obtained by the elastic net model. SVM beats other models' recall value, which is 0.4800($\pm$0.0126). LSTM has the lowest performance on recall and F1-score, which are 0.1780($\pm$0.0234) and 0.1891($\pm$0.0142), respectively. The lowest precision value generated from SVM is 0.1701($\pm$0.0092). For the 20% dataset, LSTM significantly improved recall and F1-score, but still lower than the other four models. Despite that, the performances of all models remained the same.

**COPD status prediction.** For the 10% dataset, elastic net models' results are bet-

ter than that of other models. Its precision, recall, and F1-score are 0.2938(±0.0415), 0.3446(±0.0524), and 0.3153(±0.0386), respectively. The results of XGBoost are very close to those of the elastic net model. However, LSTM has poor performance in this case, and its precision, recall, and F1-score are 0.1210(±0.0886), 0.0550(±0.0443), and 0.0749(±0.0191), respectively. For the 20% dataset, the optimum values of each indicator are also derived from the elastic net. LSTM has a decent improvement in precision performance. And its precision is 0.2871(±0.0274), while the elastic net has precision value of 0.3115(±0.0264).

**Cancer status prediction.** All metrics of two DL models underperform that of three non-deep ML models for 10% and 20% datasets. The top F1-score of the two DL models is 0.0402 for the 10% dataset, which is evaluated from the DNN model, whereas the lowest F1-score from non-deep ML methods (XGBoost) is 0.0088 higher.

In summary, it is clear that on a small dataset, the performance of non-deep ML models is superior to that of DL models.

## 3.2 Overall model performance on DL and non-deep ML

As the size of the dataset increases, the overall model performances increase, and the gap between non-deep ML and DL decreases.

**Asthma status prediction.** The F1-score of elastic net, XGBoost, SVM, LSTM, and DNN for 50% dataset are 0.3214 (±0.0047), 0.3201 (±0.0047), 0.2966 (±0.0043), 0.3088 (±0.0060), and 0.3098 (±0.0032), respectively. As the data volume rises to 100%, the performances of the five models do not change a lot.

**COPD status prediction.** When the dataset size increases to 50%, LSTM improves its performance rapidly. The F1-score of LSTM has grown three times from 0.0749 (±0.0191) to 0.3171 (±0.0154). When the dataset size expands from 50% to 100%, the optimal F1-

score is 0.3699 (±0.0110) from the elastic net. The F1-scores of XGBoost, SVM, LSTM and DNN become 0.3307(±0.0130), 0.3394(±0.0125), 0.3269 (±0.0145), and 0.3106 (±0.0157), respectively.

**Cancer status prediction.** On 50% of the dataset, the performance of all five models has improved. As the dataset grows to 100%, all models' performances are still climbing up.

In summary, DL models do not outperform non-deep ML models, even in extensive data with over 200k samples. The performance of all models improves when the sample size increases. The performance differences between DL and non-deep ML decrease as the sample size of data increases.

## 3.3 Impact of imbalanced data structure

In this study, the datasets are imbalanced, and the imbalanced rates (Control/Case) for asthma, COPD, and lung cancer are 6.5:1, 30.8:1, and 169.6:1, respectively. Model performances on cancer prediction are the lowest since the cancer dataset structure is highly imbalanced. For example, the F1-score of DNN for the 50% dataset is 0.3098 (±0.0032) for predicting asthma status, whereas it is 0.0547 (±0.0187) for predicting cancer status. Moreover, as the imbalanced rate increases, the confidence bands are getting wider. For instance, the width of the confidence band of XGBoost's F1-score for the 100% dataset is 0.0058 for predicting asthma; in contrast, it is 0.0202 for predicting lung cancer.

## 3.4 Promote hit curve as a particular visual tool

To summarize all the metric results we have found, a hit curve is promoted as a particular visual tool to compare the prediction models. In a biomedical study, it is impossible for a prediction model to accurately predict all cases, and a model can be effective without

necessarily accurately predicting all cases. For example, in our research, a prediction model is considered to be doing an excellent job if it chose a relatively small number of subjects, and correctly labeled the majority of the condition group. Therefore, hit curve is used to prioritize case. In this situation, cases with the largest prediction probabilities are chosen first. As we select cases according to the prediction probabilities, a "hit" occurs whenever the case is a success (people we selected are in a certain disease condition). Say we choose $m_1$ subjects and $m_2$ are diseased, and we can visually assess a prediction model by plotting $m_2$ against $m_1$, a so-called hit curve. A good prediction model will have $m_2$ increasing rapidly with $m_1$, as shown in Figure B.23 (only the hit curve plots for 10%, 50%, and 100% of the dataset are shown here, and the result plots for the remaining percentage of the dataset are visible in the Appendix B).

The elastic net curve and XGBoost curve are nearly identical, but they cross over each other at some points and are significantly higher than the others in predicting Asthma and COPD. For lung cancer condition prediction, XGBoost does not maintain a good performance. However, elastic net and SVM models are still superior to the LSTM model. DNN model is inferior to other models in all cases. Therefore, evidence supports that DL models often cannot overperform non-deep ML models. The brown bar appears in the 10% and 50% datasets on predicting asthma conditions. However, there are no brown bars in the 100% dataset plot. It implies the performance gap between DL and non-deep ML decreases as the sample size increases. And the difference will be trivial when the data sample size is as large as the biobank level. However, it is difficult to obtain such a large dataset. Hence, DL models often underperform non-deep ML models.

With lung cancer data's highly imbalanced data structure, The model performance is as good as that of other model B.24. None of the five models perform well when the data sample size is small. Their shaded areas are relatively broad, making the difference hard to tell. As the data sample size increases, their hit curves increase with different slopes. As a

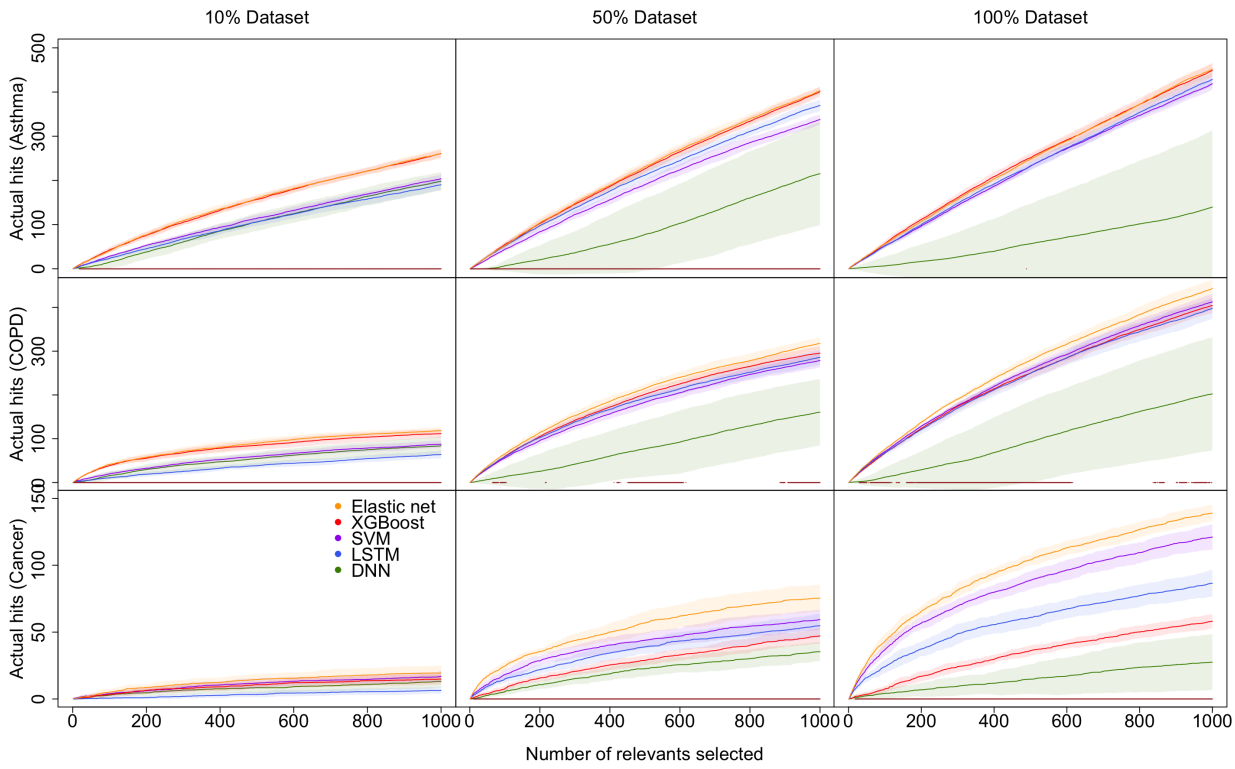


Figure 3.2: Hit curve graphs of AsthmaStatus, COPDStatus and CancerStatus classification by five models on 10%-100% data sets. The x-axis represents the number of test subjects we selected by sorting the estimated probability up to down. The y-axis of the hit curve chart represents the number of subjects with certain conditions which are correctly diagnosed in the test set. The point ($m_1$, $m_2$) indicates there are $m_2$ patients in the first $m_1$ selected subjects are correctly predicted as diseased. The curves show the average hit curves of five models, and the shaded area denotes the confidence bounds constructed using 10-fold cross-validation (i.e. $\pm$ one standard error). The brown bar at the bottom means non-deep ML models are significantly better than DL models.

consequence, the performance differences become substantial. In other words, imbalanced data is also called weighted data. The effective sample size of a weighted data is smaller than its original sample size. It is almost impossible to evaluate those five models' performances due to a lack of a sufficient sample size. As the effective sample size gradually increases, the model performance differences become apparent. However, if the effective sample size reaches a certain large amount, the differences among all models are not significant again.

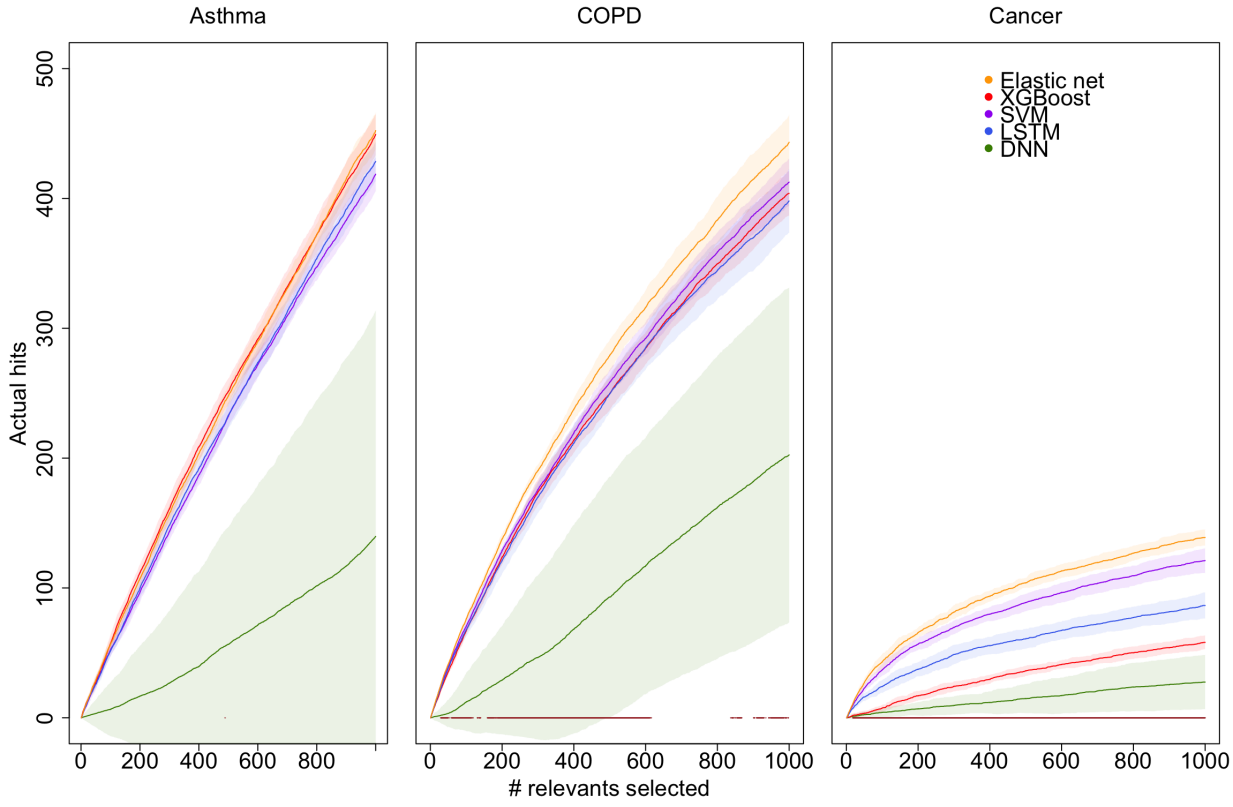


Figure 3.3: Hit curve graphs of AsthmaStatus, COPDStatus and CancerStatus classification by five models on 100% data sets. The y axis are from 0 to 500 for all three models.

# Chapter 4

# Discussion

This study evaluated the potential of DL models (DNN and LSTM) in predicting asthma, COPD, and lung cancer with various sample sizes from the UK Biobank dataset, compared with non-deep ML models (elastic net, XGBoost, and SVM). Besides the most popular performance metrics, such as the F1-score, the hit curve, as a particular visual tool, is promoted to describe the performance of predicting rare events. The results suggest that we should not apply DL methods in most GWAS studies unless we have data with biobank-level sample sizes. We conclude not recommending standard deep learning methods for GWAS studies based on the following two facts we observed in our study. First, the prediction performances of non-deep machine learning methods vastly outperform deep learning methods in small datasets (e.g., 10% and 20% random subsets of UK Biobank). Second, we observed that deep learning could not outperform non-deep methods in huge data like the entire cohort of UK Biobank (500k participants), although increasing sample size leads to the improvement of the deep learning method's performance, and its improvement is faster than non-deep methods. Therefore, we need more data than UK Biobank to prefer deep learning methods. However, the sample sizes of most publicly available genomic data cannot meet this requirement, which ranges from tens to thousands.

Although deep learning methods achieved outstanding performance in image, video, and

natural language analysis, we found their performance is not attractive in analyzing GWAS studies. We believe this is the result of two characteristics of genomic data: (1) genomic data typically has small sample sizes to fit a complex network; and (2) genomic data lacks common structural patterns like images to use pre-trained networks or take advantage of convolution layers.

Besides comparing deep learning methods with non-deep methods, the following are other important messages we learned from this study and would like to share with the audience.

We found that cancer status is much harder to predict than the other two diseases. The results show that the uneven data structure also affects the model's performance. The control/case ratio is 6.5:1 for asthma, 30.8:1 for COPD, and 169.6:1 for lung cancer, respectively. We notice that all three disease conditions are imbalanced, and the imbalanced ratio of lung cancer conditions is particularly extreme, which leads to model overfitting and underperforming prediction. Therefore, the imbalanced rate between cases and controls is also a critical influencing factor. Although we operate by regulation, rare events are harder to predict. We would do the data augmentation to prevent the imbalance problem in the future.

Our predictions of disease status are based on genomic information but not the specific diagnostic tests of related diseases. Therefore, we don't expect high accuracy in the predictions. This prediction aims to segment the patients by their predicted risk of conditions and manage them differently (e.g., following up with a different visit frequency or using follow-up disease-specific diagnostic tests).

There are two types of classification mistakes: (1) incorrectly labeling a patient as low-risk or healthy; and (2) incorrectly labeling a healthy individual as a patient or high-risk. In our case, the first type of mistake is much more harmful than the second type. Follow-up diagnosis can fix the second mistake. The first mistake may cause a delay in treatment, while the timing of treatment can be the most critical factor in treating diseases like cancer.

These two types of mistakes can be summarised by precision and recall, respectively. The most popular metric, F1-score, is the harmonic average of precision and recall, which regards these two prediction mistakes as costing equally. Fn-score can weigh two types of mistakes using user-defined weights. However, it isn't easy to define weights objectively. Hence, we introduced our preferred metric, the hit curve, for rare event detection, which focus on detecting true positive rate. Different points on the curve correspond to different decision rules about who should be labeled as patients. Users can compare many decision rules between the two methods using their hit curves. Users can also use this visual tool to decide which decision rule is best (subjectively).

# Contribution

This is joint work with collaborators of our lab, including Dr. Yao Dong, Li Xing, Yumeng Chen, Ziyu Ren, Dr. Yongfeng Dong. I am a key team member and make a substantial contribution to the entire development process, including data preparation, data analysis, preparing figures and tables, and writing the manuscript.

# Bibliography

[1] Guan Wang, Yanbo Zhang, Sijin Li, Jun Zhang, Dongkui Jiang, Xiuzhen Li, Yulin Li, and Jie Du. A machine learning-based prediction model for cardiovascular risk in women with preeclampsia. *Frontiers in cardiovascular medicine*, 8(736491), 2021.

[2] Henock M. Deberneh and Intaek Kim. Prediction of type 2 diabetes based on machine learning algorithm. *International Journal of Environmental Research and Public Health*, 18(6), 2021.

[3] Young Min Park and Byung-Joo Lee. Machine learning-based prediction model using clinico-pathologic factors for papillary thyroid carcinoma recurrence. *Scientific Reports*, 11(1), 2021.

[4] Ali Hussain, Hee-Eun Choi, Hyo-Jung Kim, Satyabrata Aich, Muhammad Saqlain, and Hee-Cheol Kim. Forecast the exacerbation in patients of chronic obstructive pulmonary disease with clinical indicators using machine learning techniques. *Diagnostics*, 11(5):829, 2021.

[5] Charlie F Rowlands, Diana Baralle, and Jamie M Ellingford. Machine learning approaches for the prioritization of genomic variants impacting pre-mrna splicing. *Cells*, 8(12):1513, 2019.

[6] Katerina Placek, Michael Benatar, Joanne Wuu, Evadnie Rampersaud, Laura Hennessy, Vivianna M Van Deerlin, Murray Grossman, David J Irwin, Lauren Elman, Leo McCluskey, Colin Quinn, Volkan Granit, Jeffrey M Statland, Ted M Burns, John Ravits, Andrea Swenson, Jon Katz, Erik P Pioro, Carlayne Jackson, James Caress, Yuen So, Samuel Maiser, David Walk, Edward B Lee, John Q Trojanowski, Philip Cook, James Gee, Jin Sha, Adam C Naj, Rosa Rademakers, The CReATe Consortium, Wenan Chen, Gang Wu, J Paul Taylor, and Corey T McMillan. Machine learning suggests polygenic risk for cognitive dysfunction in amyotrophic lateral sclerosis. *EMBO Molecular Medicine*, 13(1):e12595, 2021.

[7] Hanqing Chao, Hongming Shan, Fatemeh Homayounieh, Ramandeep Singh, Ruhani Khera, Hengtao Guo, Timothy Su, Ge Wang, Mannudeep Kalra, and Pingkun Yan. Deep learning predicts cardiovascular disease risks from lung cancer screening low dose computed tomography. *Nature Communications*, 12:2963, 05 2021.

[8] Hongyu Zhou, Lu Li, Zhenyu Liu, Kankan Zhao, Xiuyu Chen, Minjie Lu, Gang Yin, Lei Song, Shihua Zhao, Hairong Zheng, and Jie Tian. Deep learning algorithm to improve hypertrophic cardiomyopathy mutation prediction using cardiac cine images. *European Radiology*, 31, 11 2020.

[9] Shen Lin, Zhigang Li, Bowen Fu, Sipeng Chen, Xi Li, Yang Wang, Xiaoyi Wang, Bin Lv, Bo Xu, Xiantao Song, Yao-Jun Zhang, Xiang Cheng, Weijian Huang, Jun Pu, Qi Zhang, Yunlong Xia, Bai Du, Xiangyang Ji, and Zhe Zheng. Feasibility of using deep learning to detect coronary artery disease based on facial photo. *European Heart Journal*, 41(46):4400–4411, 08 2020.

[10] Cheng Jin, Heng Yu, Jia Ke, Peirong Ding, Yongju Yi, Xiaofeng Jiang, Xin Duan, Jinghua Tang, Daniel Chang, Xiaodong Wu, Feng Gao, and Ruijiang Li. Predicting

treatment response from longitudinal images using multi-task deep learning. *Nature Communications*, 12, 03 2021.

[11] Yilan Sun, Stephen Milne, Jen Erh Jaw, Chen Yang, Feng Xu, Xuan Li, Ma'en Obeidat, and Don Sin. Bmi is associated with fev1 decline in chronic obstructive pulmonary disease: A meta-analysis of clinical trials. *Respiratory Research*, 20, 10 2019.

[12] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

[13] Baoshan Ma, Ge Yan, Bingjie Chai, and Xiaoyu Hou. XGBLC: an improved survival prediction model based on XGBoost. *Bioinformatics*, 38(2):410–418, 09 2021.

[14] Shuanglong Fan, Zhiqiang Zhao, Yanbo Zhang, Hongmei Yu, Chuchu Zheng, Xueqian Huang, Zhenhuan Yang, Meng Xing, Qing Lu, and Yanhong Luo. Probability calibration-based prediction of recurrence rate in patients with diffuse large b-cell lymphoma. *BioData Mining*, 14, 08 2021.

[15] Ammar H. Elsheikh, Amal I. Saba, Mohamed Abd Elaziz, Songfeng Lu, S. Shanmugan, T. Muthuramalingam, Ravinder Kumar, Ahmed O. Mosleh, F.A. Essa, and Taher A. Shehabeldeen. Deep learning-based forecasting model for covid-19 outbreak in saudi arabia. *Process Safety and Environmental Protection*, 149:223–233, 2021.

[16] Junhua Ye, Shunfang Wang, Xin Yang, and Tang Xianjun. Gene prediction of aging-related diseases based on dnn and mashup. *BMC Bioinformatics*, 22, 12 2021.

# Appendix A

# Tables

Table A.1: Average numbers of case for 10 datasets of same percentages.

| | 10% | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 100% |
|---|---|---|---|---|---|---|---|---|---|---|
| Asthmastatus | 2772 | 5556 | 8336 | 11108 | 13852 | 16613 | 19396 | 22163 | 24941 | 27692 |
| COPDstatus | 656 | 1309 | 1944 | 2581 | 3227 | 3864 | 4515 | 5154 | 5799 | 6449 |
| Cancerstatus | 119 | 242 | 363 | 485 | 600 | 720 | 838 | 957 | 1084 | 1202 |

Table A.2: Number of patients who have asthma condition in each dataset.

| | Data1 | Data2 | Data3 | Data4 | Data5 | Data6 | Data7 | Data8 | Data9 | Data10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 10% | 2765 | 2788 | 2654 | 2736 | 2781 | 2781 | 2757 | 2744 | 2866 | 2848 |
| 20% | 5499 | 5611 | 5429 | 5515 | 5576 | 5576 | 5638 | 5539 | 5612 | 5567 |
| 30% | 8302 | 8427 | 8171 | 8304 | 8445 | 8445 | 8353 | 8316 | 8301 | 8295 |
| 40% | 11112 | 11203 | 10909 | 11129 | 11191 | 11191 | 11085 | 11054 | 11137 | 11064 |
| 50% | 13917 | 13963 | 13699 | 13880 | 13892 | 13892 | 13907 | 13759 | 13872 | 13742 |
| 60% | 16622 | 16769 | 16465 | 16597 | 16601 | 16601 | 16698 | 16561 | 16616 | 16598 |
| 70% | 19381 | 19509 | 19253 | 19344 | 19414 | 19414 | 19411 | 19394 | 19494 | 19344 |
| 80% | 22160 | 22240 | 22072 | 22143 | 22166 | 22166 | 22219 | 22121 | 22203 | 22136 |
| 90% | 24944 | 24965 | 24912 | 24934 | 24946 | 24946 | 24961 | 24943 | 24901 | 24961 |
| 100% | 27692 | 27692 | 27692 | 27692 | 27692 | 27692 | 27692 | 27692 | 27692 | 27692 |

Table A.3: Number of patients who have COPD condition in each dataset.

| | Data1 | Data2 | Data3 | Data4 | Data5 | Data6 | Data7 | Data8 | Data9 | Data10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 10% | 681 | 617 | 650 | 663 | 634 | 646 | 680 | 643 | 686 | 659 |
| 20% | 1330 | 1249 | 1305 | 1326 | 1305 | 1337 | 1306 | 1285 | 1321 | 1321 |
| 30% | 1957 | 1942 | 1953 | 1972 | 1955 | 1978 | 1905 | 1891 | 1929 | 1960 |
| 40% | 2569 | 2608 | 2583 | 2626 | 2591 | 2623 | 2532 | 2554 | 2570 | 2560 |
| 50% | 3212 | 3259 | 3222 | 3267 | 3225 | 3284 | 3237 | 3202 | 3201 | 3164 |
| 60% | 3845 | 3861 | 3884 | 3897 | 3872 | 3878 | 3896 | 3846 | 3829 | 3832 |
| 70% | 4505 | 4472 | 4520 | 4604 | 4513 | 4499 | 4555 | 4527 | 4483 | 4470 |
| 80% | 5159 | 5159 | 5171 | 5208 | 5156 | 5133 | 5174 | 5134 | 5136 | 5107 |
| 90% | 5816 | 5789 | 5827 | 5807 | 5805 | 5784 | 5800 | 5809 | 5766 | 5785 |
| 100% | 6449 | 6449 | 6449 | 6449 | 6449 | 6449 | 6449 | 6449 | 6449 | 6449 |

Table A.4: Number of patients who have lung cancer condition in each dataset.

| | Data1 | Data2 | Data3 | Data4 | Data5 | Data6 | Data7 | Data8 | Data9 | Data10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 10% | 128 | 122 | 109 | 122 | 125 | 117 | 113 | 109 | 123 | 117 |
| 20% | 259 | 231 | 233 | 230 | 262 | 244 | 240 | 233 | 240 | 244 |
| 30% | 372 | 373 | 349 | 368 | 384 | 363 | 353 | 353 | 350 | 364 |
| 40% | 498 | 483 | 479 | 500 | 508 | 483 | 484 | 465 | 466 | 488 |
| 50% | 622 | 602 | 600 | 598 | 614 | 612 | 593 | 582 | 572 | 604 |
| 60% | 730 | 739 | 711 | 729 | 729 | 724 | 701 | 702 | 703 | 736 |
| 70% | 830 | 849 | 837 | 843 | 856 | 826 | 837 | 814 | 821 | 865 |
| 80% | 969 | 964 | 951 | 958 | 968 | 948 | 968 | 940 | 936 | 969 |
| 90% | 1089 | 1079 | 1073 | 1090 | 1093 | 1094 | 1086 | 1082 | 1070 | 1080 |
| 100% | 1202 | 1202 | 1202 | 1202 | 1202 | 1202 | 1202 | 1202 | 1202 | 1202 |

Table A.5: Model evaluations of five prediction models on training 10%-100% datasets for predicting Asthma.

| Dataset | Model | Precision | Recall | F1_Score | AUC |
|---|---|---|---|---|---|
| 10% | ElasticNet | 0.2404±0.0127 | 0.4535±0.0307 | 0.3135±0.0098 | 0.6566±0.0118 |
| | XGBoost | 0.2343±0.0163 | 0.4505±0.0360 | 0.3097±0.0123 | 0.6543±0.0095 |
| | SVM | 0.1701±0.0092 | 0.4800±0.0126 | 0.2635±0.0129 | 0.5655±0.0111 |
| | LSTM | 0.2036±0.0125 | 0.1780±0.0234 | 0.1891±0.0142 | 0.5787±0.0099 |
| | DNN | 0.1907±0.0143 | 0.4635±0.0432 | 0.2828±0.0121 | 0.5886±0.0078 |
| 20% | ElasticNet | 0.2433±0.0087 | 0.4710±0.0238 | 0.3187±0.0071 | 0.6679±0.0070 |
| | XGBoost | 0.2412±0.0057 | 0.4501±0.0286 | 0.3137±0.0086 | 0.6638±0.0083 |
| | SVM | 0.1903±0.0057 | 0.4847±0.0131 | 0.2794±0.0036 | 0.5859±0.0063 |
| | LSTM | 0.2323±0.021 | 0.2787± 0.0232 | 0.2587±0.0118 | 0.6293±0.0084 |
| | DNN | 0.2235±0.0033 | 0.4654±0.0166 | 0.3034±0.0063 | 0.6063±0.0069 |
| 30% | ElasticNet | 0.2416±0.0078 | 0.4729±0.0197 | 0.3195±0.0046 | 0.6705±0.0053 |
| | XGBoost | 0.2424±0.0057 | 0.4537±0.0213 | 0.3155±0.0055 | 0.6637±0.0060 |
| | SVM | 0.2004±0.0047 | 0.4888±0.0132 | 0.2886±0.0054 | 0.5956±0.0065 |
| | LSTM | 0.2373±0.0090 | 0.3553±0.0259 | 0.2840±0.0096 | 0.6398±0.0061 |
| | DNN | 0.2235±0.0049 | 0.4706±0.0315 | 0.3072±0.0069 | 0.6112±0.0074 |
| 40% | ElasticNet | 0.2445±0.0091 | 0.4687±0.0270 | 0.3208±0.0047 | 0.6734±0.0038 |
| | XGBoost | 0.2448±0.0093 | 0.4550±0.0301 | 0.3177±0.0061 | 0.6707±0.0052 |
| | SVM | 0.2074±0.0066 | 0.4807±0.0180 | 0.2944±0.0057 | 0.6009±0.0043 |
| | LSTM | 0.2424±0.0096 | 0.3931±0.0305 | 0.2992±0.0095 | 0.6520±0.0063 |
| | DNN | 0.2239±0.0087 | 0.4665±0.0412 | 0.3106±0.0055 | 0.6120±0.0052 |
| 50% | ElasticNet | 0.2410±0.0070 | 0.4836±0.0189 | 0.3214±0.0047 | 0.6744±0.0030 |
| | XGBoost | 0.2419±0.0085 | 0.4755±0.0264 | 0.3201±0.0047 | 0.6726±0.0041 |
| | SVM | 0.2108±0.0049 | 0.4844±0.0112 | 0.2966±0.0043 | 0.6027±0.0037 |

Table A.5 continued from previous page

| | | | | | |
|---|---|---|---|---|---|
| | LSTM | 0.2382±0.0029 | 0.4398±0.0227 | 0.3088±0.0060 | 0.6593±0.0053 |
| | DNN | 0.2241±0.0062 | 0.4803±0.0286 | 0.3098±0.0032 | 0.6122±0.0044 |
| 60% | ElasticNet | 0.2416±0.0087 | 0.4801±0.0155 | 0.3211±0.0056 | 0.6743±0.0035 |
| | XGBoost | 0.2443±0.0084 | 0.4644±0.0214 | 0.3198±0.0051 | 0.6733±0.0042 |
| | SVM | 0.2175±0.0055 | 0.4850±0.0220 | 0.3006±0.0051 | 0.6058±0.0040 |
| | LSTM | 0.2415±0.0090 | 0.4356±0.0309 | 0.3101±0.0066 | 0.6608±0.0049 |
| | DNN | 0.2305±0.0072 | 0.4781±0.0299 | 0.3117±0.0048 | 0.6142±0.0047 |
| 70% | ElasticNet | 0.2436±0.0082 | 0.4751±0.0175 | 0.3217±0.0040 | 0.6743±0.0024 |
| | XGBoost | 0.2448±0.0049 | 0.4638±0.0148 | 0.3203±0.0015 | 0.6735±0.0031 |
| | SVM | 0.2181±0.0062 | 0.4851±0.0085 | 0.3028±0.0042 | 0.6080±0.0023 |
| | LSTM | 0.2367±0.0094 | 0.4575±0.0217 | 0.3115±0.0057 | 0.6637±0.0041 |
| | DNN | 0.2292±0.0104 | 0.4815±0.0353 | 0.3095±0.0053 | 0.6124±0.0044 |
| 80% | ElasticNet | 0.2453±0.0068 | 0.4710±0.0214 | 0.3222±0.0029 | 0.6750±0.0032 |
| | XGBoost | 0.2466±0.0058 | 0.4595±0.0182 | 0.3207±0.0030 | 0.6741±0.0027 |
| | SVM | 0.2229±0.0059 | 0.4856±0.0177 | 0.3048±0.0034 | 0.6089±0.0028 |
| | LSTM | 0.2430±0.0051 | 0.4416±0.0184 | 0.3133±0.0030 | 0.6641±0.0033 |
| | DNN | 0.2345±0.0120 | 0.4638±0.0347 | 0.3105±0.0051 | 0.6119±0.0036 |
| 90% | ElasticNet | 0.2459±0.0078 | 0.4671±0.0226 | 0.3218±0.0031 | 0.6745±0.0032 |
| | XGBoost | 0.2438±0.0045 | 0.4686±0.0102 | 0.3206±0.0026 | 0.6742±0.0024 |
| | SVM | 0.2229±0.0048 | 0.4880±0.0129 | 0.3061±0.0033 | 0.6103±0.0027 |
| | LSTM | 0.2377±0.0031 | 0.4457±0.0278 | 0.3098±0.0063 | 0.6595±0.0051 |
| | DNN | 0.2364±0.0045 | 0.4605±0.0168 | 0.3118±0.0047 | 0.6128±0.0041 |
| 100% | ElasticNet | 0.2444±0.0055 | 0.4657±0.0148 | 0.3204±0.0019 | 0.6739±0.0030 |
| | XGBoost | 0.2446±0.0044 | 0.4669±0.0141 | 0.3209±0.0029 | 0.6737±0.0025 |
| | SVM | 0.2257±0.0032 | 0.4904±0.0094 | 0.3066±0.0031 | 0.6107±0.0031 |

Table A.5 continued from previous page

| | | | | | |
|---|---|---|---|---|---|
| | LSTM | 0.2364±0.0073 | 0.4559±0.0243 | 0.3112±0.0032 | 0.6636±0.0026 |
| | DNN | 0.2380±0.0096 | 0.4542±0.0276 | 0.3117±0.0045 | 0.6128±0.0034 |

Table A.6: Model evaluations of five prediction models on training 10%-100% datasets for predicting COPD.

| Dataset | Model | Precision | Recall | F1_Score | AUC |
|---|---|---|---|---|---|
| 10% | ElasticNet | 0.2938±0.0415 | 0.3446±0.0524 | 0.3153±0.0386 | 0.8402±0.0154 |
| | XGBoost | 0.2863±0.0360 | 0.3447±0.0521 | 0.3053±0.0357 | 0.8261±0.0168 |
| | SVM | 0.1533±0.0180 | 0.2389±0.0464 | 0.1852±0.0227 | 0.5970±0.0198 |
| | LSTM | 0.1210±0.0886 | 0.0550±0.0443 | 0.0749±0.0191 | 0.6558±0.0176 |
| | DNN | 0.2149±0.0291 | 0.3055±0.0421 | 0.2451±0.0129 | 0.6334±0.0150 |
| 20% | ElasticNet | 0.3115±0.0264 | 0.3758±0.0286 | 0.3399±0.0222 | 0.8473±0.0119 |
| | XGBoost | 0.3067±0.0249 | 0.3679±0.0270 | 0.3338±0.0212 | 0.8378±0.0100 |
| | SVM | 0.1997±0.0160 | 0.3028±0.0448 | 0.2394±0.0198 | 0.6315±0.0199 |
| | LSTM | 0.2871±0.0274 | 0.2208±0.0330 | 0.2482±0.0268 | 0.8084±0.0146 |
| | DNN | 0.2402±0.0079 | 0.3169±0.0369 | 0.2885±0.0186 | 0.6495±0.0167 |
| 30% | ElasticNet | 0.3286±0.0226 | 0.3699±0.0312 | 0.3472±0.0206 | 0.8509±0.0110 |
| | XGBoost | 0.3099±0.0208 | 0.3544±0.0263 | 0.3296±0.0148 | 0.8396±0.0106 |
| | SVM | 0.2490±0.0270 | 0.3206±0.0330 | 0.2793±0.0239 | 0.6444±0.0156 |
| | LSTM | 0.3153±0.0222 | 0.2753±0.0233 | 0.2935±0.0199 | 0.8249±0.0143 |
| | DNN | 0.2687±0.0265 | 0.3348±0.0258 | 0.2974±0.0227 | 0.6523±0.0123 |
| 40% | ElasticNet | 0.3310±0.0136 | 0.3714±0.0264 | 0.3496±0.0155 | 0.8568±0.0093 |
| | XGBoost | 0.3166±0.0126 | 0.3460±0.0225 | 0.3301±0.0164 | 0.8437±0.0090 |
| | SVM | 0.2686±0.0177 | 0.3415±0.0290 | 0.2978±0.0138 | 0.6556±0.0132 |
| | LSTM | 0.3150±0.0223 | 0.3205±0.0269 | 0.3065±0.0202 | 0.8320±0.0088 |
| | DNN | 0.2753±0.0365 | 0.3360±0.0337 | 0.2999±0.0218 | 0.6533±0.0147 |
| 50% | ElasticNet | 0.3339±0.0160 | 0.3845±0.0222 | 0.3571±0.0154 | 0.8603±0.0083 |
| | XGBoost | 0.3137±0.0161 | 0.3570±0.0204 | 0.3335±0.0137 | 0.8429±0.0073 |
| | SVM | 0.2847±0.0187 | 0.3457±0.0201 | 0.3120±0.0172 | 0.6590±0.0100 |

Table A.6 continued from previous page

| | | | | | |
|---|---|---|---|---|---|
| | LSTM | 0.3237±0.0144 | 0.3213±0.0201 | 0.3171±0.0154 | 0.8417±0.0104 |
| | DNN | 0.2844±0.0268 | 0.3365±0.0371 | 0.3063±0.0191 | 0.6545±0.0166 |
| 60% | ElasticNet | 0.3353±0.0140 | 0.3839±0.0204 | 0.3576±0.0120 | 0.8635±0.0077 |
| | XGBoost | 0.3105±0.0081 | 0.3607±0.0206 | 0.3335±0.0110 | 0.8444±0.0063 |
| | SVM | 0.2874±0.0188 | 0.3557±0.0212 | 0.3175±0.0164 | 0.6637±0.0102 |
| | LSTM | 0.3236±0.0144 | 0.3322±0.0212 | 0.3267±0.0154 | 0.8464±0.0104 |
| | DNN | 0.2830±0.0177 | 0.3307±0.0231 | 0.3148±0.0165 | 0.6639±0.0139 |
| 70% | ElasticNet | 0.3455±0.0265 | 0.3802±0.0158 | 0.3612±0.0135 | 0.8646±0.0065 |
| | XGBoost | 0.3104±0.0087 | 0.3608±0.0249 | 0.3332±0.0109 | 0.8433±0.0057 |
| | SVM | 0.3033±0.0119 | 0.3546±0.0312 | 0.3261±0.0131 | 0.6643±0.0141 |
| | LSTM | 0.3235±0.0185 | 0.3330±0.0276 | 0.3272±0.0155 | 0.8453±0.0062 |
| | DNN | 0.2973±0.0041 | 0.3361±0.0248 | 0.3149±0.0122 | 0.6553±0.0114 |
| 80% | ElasticNet | 0.3512±0.0205 | 0.3776±0.0137 | 0.3637±0.0146 | 0.8649±0.0068 |
| | XGBoost | 0.3184±0.0115 | 0.3530±0.0231 | 0.3344±0.0131 | 0.8442±0.0056 |
| | SVM | 0.3090±0.0133 | 0.3671±0.0298 | 0.3347±0.0129 | 0.6704±0.0135 |
| | LSTM | 0.3234±0.0217 | 0.3387±0.0281 | 0.3294±0.0135 | 0.8455±0.0071 |
| | DNN | 0.2918±0.0126 | 0.3416±0.0211 | 0.3121±0.0081 | 0.6553±0.0092 |
| 90% | ElasticNet | 0.3498±0.0132 | 0.3864±0.0126 | 0.3671±0.0106 | 0.8658±0.0070 |
| | XGBoost | 0.3154±0.0180 | 0.3531±0.0180 | 0.3326±0.0109 | 0.8449±0.0059 |
| | SVM | 0.3162±0.0133 | 0.3567±0.0189 | 0.3349±0.0112 | 0.6660±0.0088 |
| | LSTM | 0.3203±0.0106 | 0.3387±0.0214 | 0.3287±0.0097 | 0.8446±0.0071 |
| | DNN | 0.2981±0.0237 | 0.3390±0.0265 | 0.3159±0.0150 | 0.6599±0.0118 |
| 100% | ElasticNet | 0.3549±0.0112 | 0.3870±0.0205 | 0.3699±0.0110 | 0.8660±0.0071 |
| | XGBoost | 0.3136±0.0141 | 0.3501±0.0148 | 0.3307±0.0130 | 0.8429±0.0066 |
| | SVM | 0.3247±0.0135 | 0.3562±0.0206 | 0.3394±0.0125 | 0.6661±0.0097 |

Table A.6 continued from previous page

| | | | | | |
|---|---|---|---|---|---|
| | LSTM | 0.3143±0.0172 | 0.3418±0.0223 | 0.3269±0.0145 | 0.8412±0.0092 |
| | DNN | 0.2907±0.0239 | 0.3359±0.0241 | 0.3106±0.0157 | 0.6546±0.0110 |

Table A.7: Model evaluations of five prediction models on training 10%-100% datasets for predicting lung cancer.

| Dataset | Model | Precision | Recall | F1_Score | AUC |
|---|---|---|---|---|---|
| 10% | ElasticNet | 0.0619±0.0282 | 0.1573±0.1306 | 0.0814±0.0357 | 0.8010±0.0460 |
| | XGBoost | 0.0319±0.0156 | 0.1490±0.1016 | 0.0490±0.0219 | 0.7134±0.0636 |
| | SVM | 0.0478±0.0272 | 0.1413±0.0985 | 0.0659±0.0310 | 0.5601±0.0420 |
| | LSTM | 0.0341±0.0250 | 0.0554±0.0003 | 0.0273±0.0030 | 0.5248±0.1088 |
| | DNN | 0.0398±0.0271 | 0.1036±0.1006 | 0.0402±0.0341 | 0.5513±0.0331 |
| 20% | ElasticNet | 0.1259±0.0459 | 0.2161±0.0471 | 0.1557±0.0474 | 0.8612±0.0291 |
| | XGBoost | 0.0507±0.0178 | 0.1280±0.0599 | 0.0701±0.0228 | 0.7830±0.0201 |
| | SVM | 0.1119±0.0278 | 0.1405±0.0679 | 0.1192±0.0392 | 0.5669±0.0329 |
| | LSTM | 0.0419±0.0270 | 0.0511±0.0012 | 0.0449±0.0053 | 0.5392±0.0183 |
| | DNN | 0.0426±0.0339 | 0.1144±0.0484 | 0.0437±0.0354 | 0.5531±0.0198 |
| 30% | ElasticNet | 0.1568±0.0357 | 0.1959±0.0413 | 0.1704±0.0264 | 0.8738±0.0209 |
| | XGBoost | 0.0547±0.0119 | 0.1394±0.0298 | 0.0765±0.0108 | 0.8027±0.0162 |
| | SVM | 0.1234±0.0295 | 0.1456±0.0365 | 0.1316±0.0295 | 0.5697±0.0179 |
| | LSTM | 0.1509±0.0340 | 0.0721±0.0322 | 0.0975±0.0354 | 0.8134±0.0195 |
| | DNN | 0.0463±0.0377 | 0.1387±0.0562 | 0.0430±0.0348 | 0.5791±0.0366 |
| 40% | ElasticNet | 0.1701±0.0428 | 0.2097±0.0343 | 0.1855±0.0325 | 0.8812±0.0231 |
| | XGBoost | 0.0602±0.0140 | 0.1432±0.0354 | 0.0839±0.0183 | 0.8109±0.0191 |
| | SVM | 0.1345±0.0311 | 0.1376±0.0447 | 0.1329±0.0301 | 0.5662±0.0219 |
| | LSTM | 0.1605±0.0494 | 0.0823±0.0364 | 0.1062±0.0415 | 0.8314±0.0251 |
| | DNN | 0.0437±0.0227 | 0.1517±0.0415 | 0.0508±0.0296 | 0.5600±0.0131 |
| 50% | ElasticNet | 0.2113±0.0353 | 0.2163±0.0320 | 0.2110±0.0235 | 0.8990±0.0149 |
| | XGBoost | 0.0616±0.0122 | 0.1702±0.0361 | 0.0885±0.0128 | 0.8168±0.0196 |
| | SVM | 0.1616±0.0238 | 0.1767±0.0397 | 0.1656±0.0239 | 0.5856±0.0193 |

Table A.7 continued from previous page

| | | | | | |
|---|---|---|---|---|---|
| | LSTM | 0.1655±0.0396 | 0.0929±0.0301 | 0.1146±0.0269 | 0.8376±0.0211 |
| | DNN | 0.0476±0.0131 | 0.1597±0.0406 | 0.0547±0.0187 | 0.5779±0.0197 |
| 60% | ElasticNet | 0.2197±0.0211 | 0.2504±0.0225 | 0.2327±0.0119 | 0.9090±0.0132 |
| | XGBoost | 0.0663±0.0104 | 0.1680±0.0373 | 0.0940±0.0127 | 0.8182±0.0179 |
| | SVM | 0.1758±0.0204 | 0.2027±0.0423 | 0.1854±0.0226 | 0.5985±0.0205 |
| | LSTM | 0.1754±0.0664 | 0.1139±0.0244 | 0.1338±0.0281 | 0.8480±0.0159 |
| | DNN | 0.0527±0.0198 | 0.1690±0.0577 | 0.0598±0.0262 | 0.5698±0.0260 |
| 70% | ElasticNet | 0.2500±0.0362 | 0.2291±0.0292 | 0.2381±0.0278 | 0.9130±0.0109 |
| | XGBoost | 0.0662±0.0271 | 0.1908±0.0400 | 0.0933±0.0115 | 0.8188±0.0160 |
| | SVM | 0.1804±0.0180 | 0.2132±0.0417 | 0.1938±0.0230 | 0.6038±0.0204 |
| | LSTM | 0.1680±0.0449 | 0.1198±0.0414 | 0.1352±0.0352 | 0.8443±0.0133 |
| | DNN | 0.0464±0.0254 | 0.1702±0.0496 | 0.0635±0.0319 | 0.5613±0.0216 |
| 80% | ElasticNet | 0.2546±0.0213 | 0.2579±0.0280 | 0.2551±0.0168 | 0.9181±0.0101 |
| | XGBoost | 0.0668±0.0068 | 0.1767±0.0290 | 0.0964±0.0092 | 0.8226±0.0142 |
| | SVM | 0.2001±0.0232 | 0.2271±0.0390 | 0.2104±0.0198 | 0.6108±0.0189 |
| | LSTM | 0.1606±0.0416 | 0.1261±0.0221 | 0.1393±0.0249 | 0.8395±0.0404 |
| | DNN | 0.0432±0.0176 | 0.1766±0.0440 | 0.0642±0.0199 | 0.5707±0.0242 |
| 90% | ElasticNet | 0.2736±0.0220 | 0.2560±0.0353 | 0.2621±0.0165 | 0.9231±0.0089 |
| | XGBoost | 0.0624±0.0079 | 0.1965±0.0567 | 0.0933±0.0098 | 0.8193±0.0134 |
| | SVM | 0.2287±0.0200 | 0.2147±0.0298 | 0.2201±0.0178 | 0.6052±0.0146 |
| | LSTM | 0.1708±0.0472 | 0.1278±0.0267 | 0.1393±0.0208 | 0.8541±0.0149 |
| | DNN | 0.0549±0.0209 | 0.1802±0.0589 | 0.0751±0.0204 | 0.5673±0.0202 |
| 100% | ElasticNet | 0.2783±0.0219 | 0.2704±0.0304 | 0.2731±0.0184 | 0.9247±0.0077 |
| | XGBoost | 0.0635±0.0063 | 0.1610±0.0392 | 0.0899±0.0101 | 0.8219±0.0135 |
| | SVM | 0.2310±0.0244 | 0.2358±0.0284 | 0.2313±0.0120 | 0.6155±0.0137 |

Table A.7 continued from previous page

| | | | | | |
|---|---|---|---|---|---|
| | LSTM | 0.1833±0.0290 | 0.1279±0.0320 | 0.1483±0.0250 | 0.8563±0.0142 |
| | DNN | 0.0551±0.0218 | 0.1840±0.0383 | 0.0857±0.0264 | 0.5685±0.0133 |

# Appendix B

# Figures

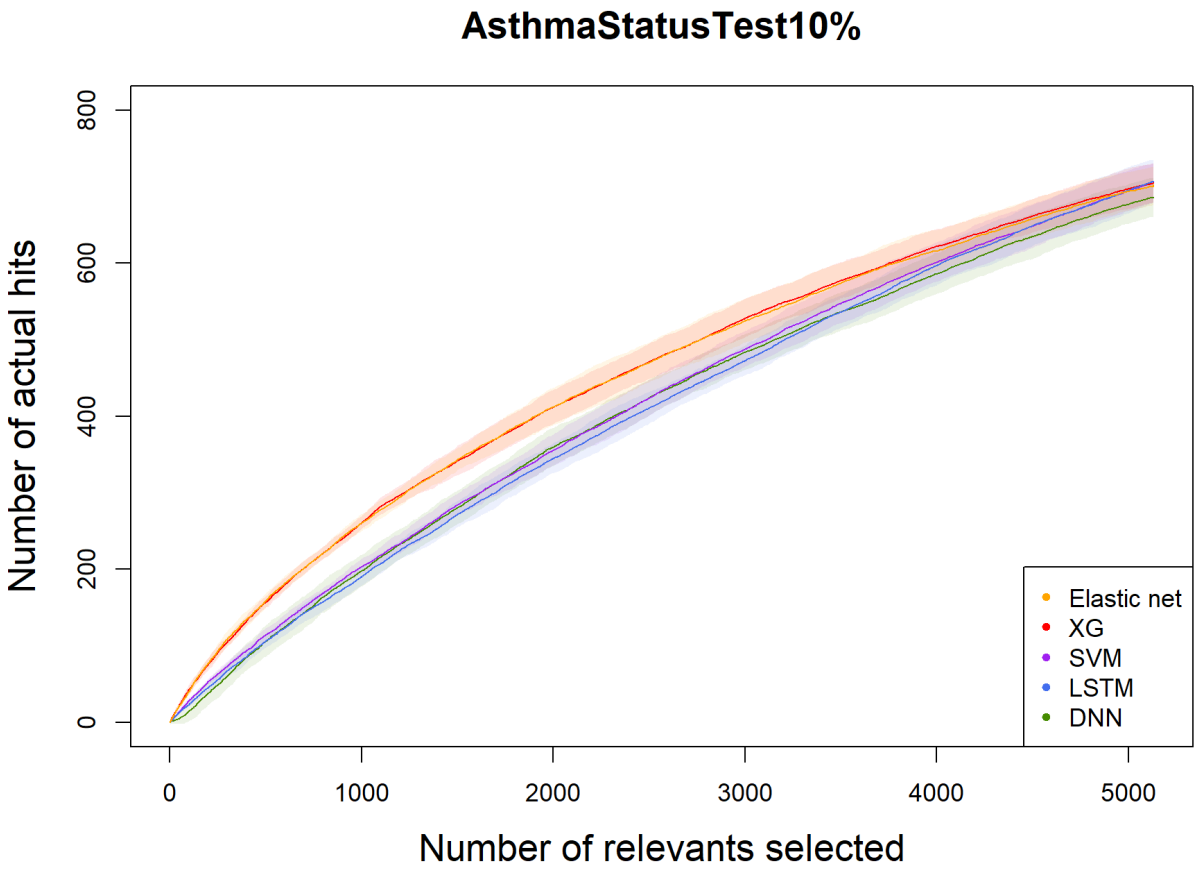


Figure B.1: Hit curve on asthma for 10% dataset.

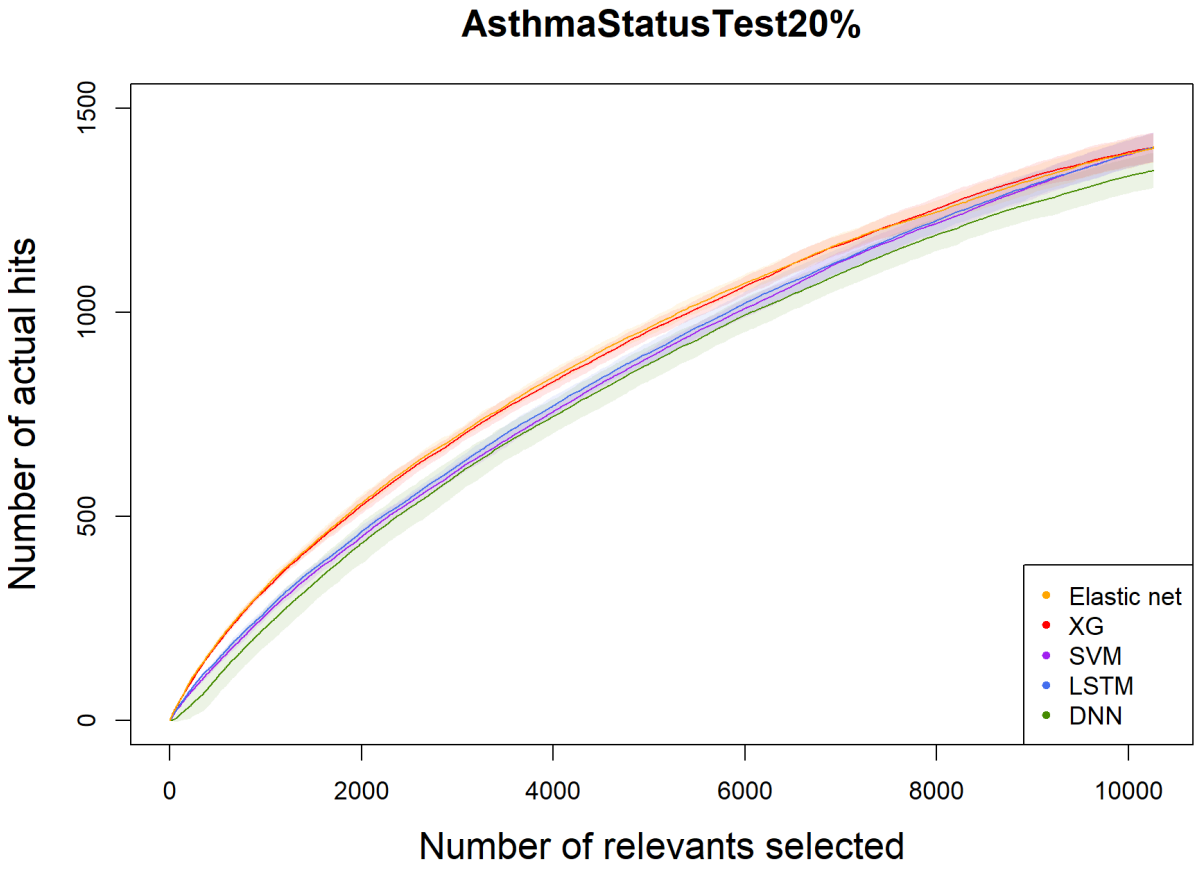


Figure B.2: Hit curve on asthma for 20% dataset.

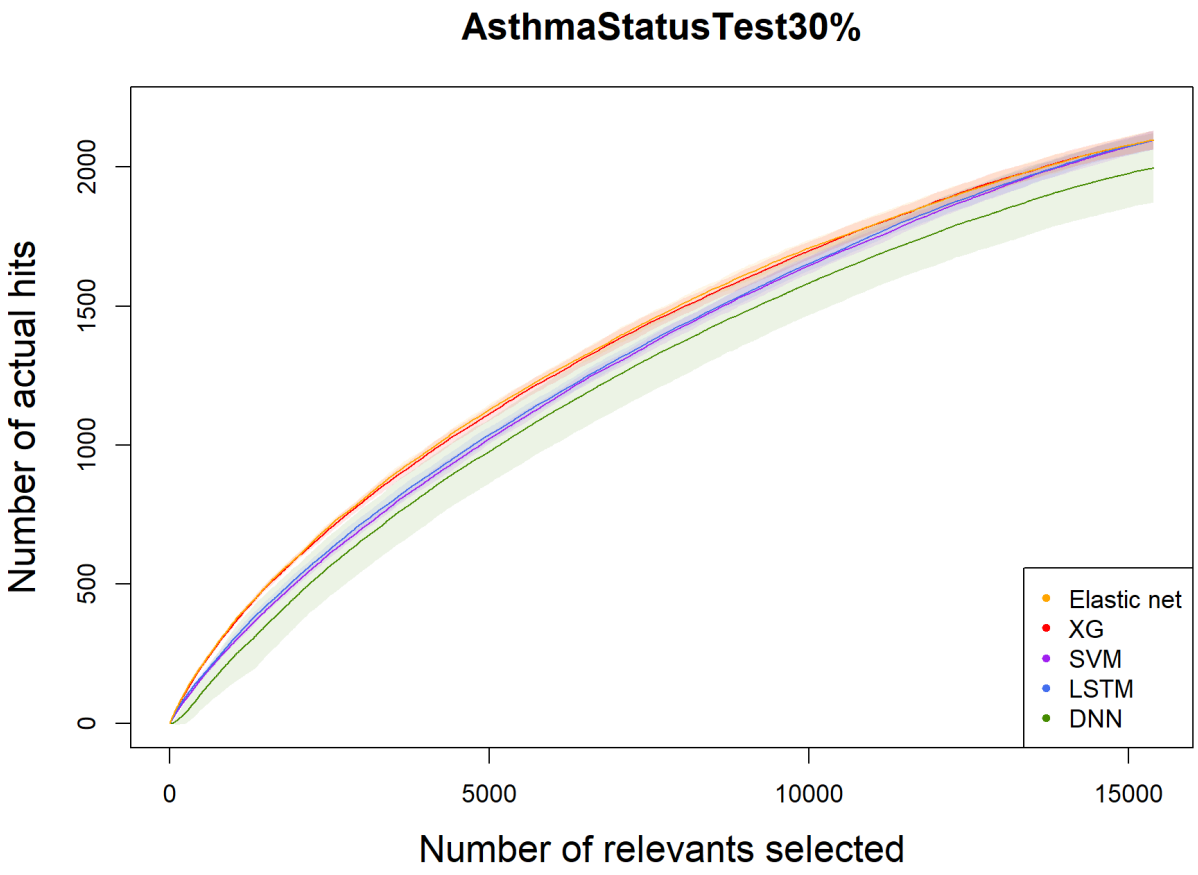


Figure B.3: Hit curve on asthma for 30% dataset.

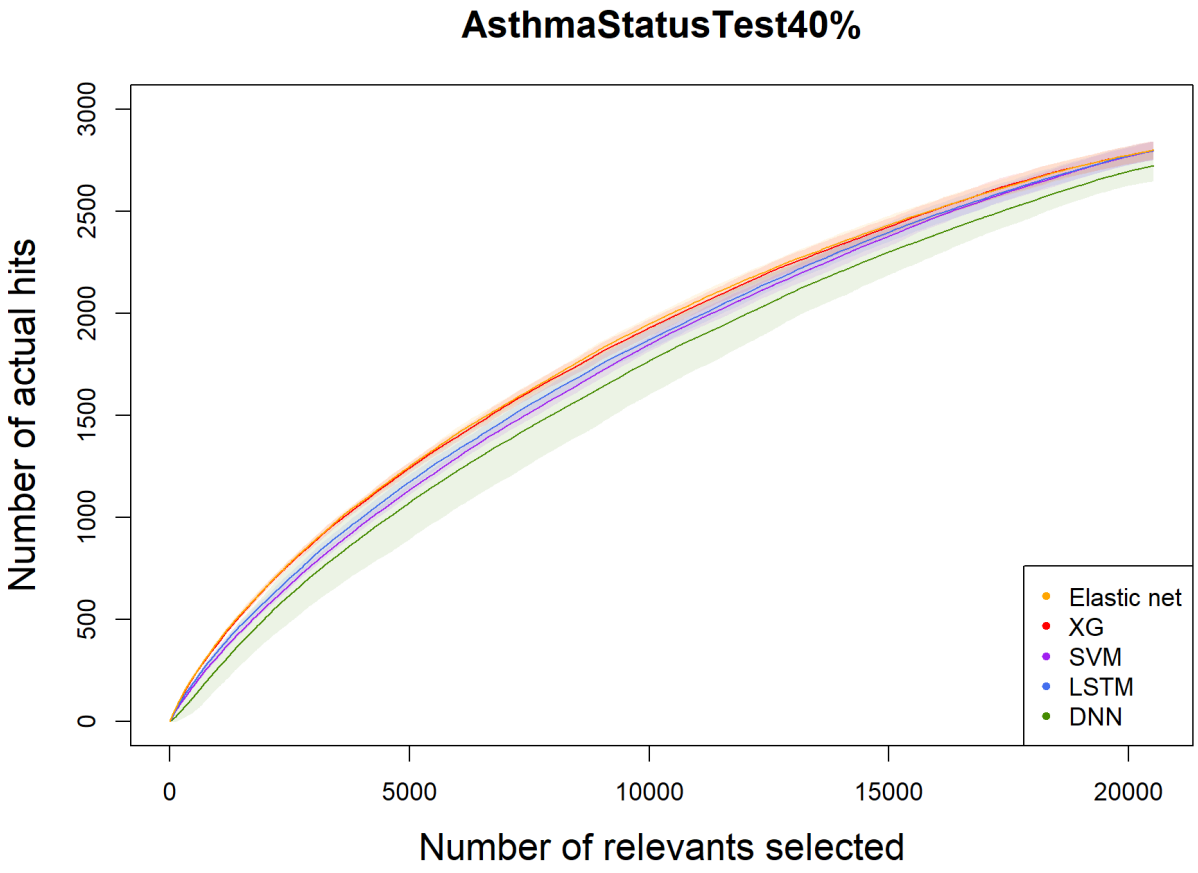


Figure B.4: Hit curve on asthma for 40% dataset.

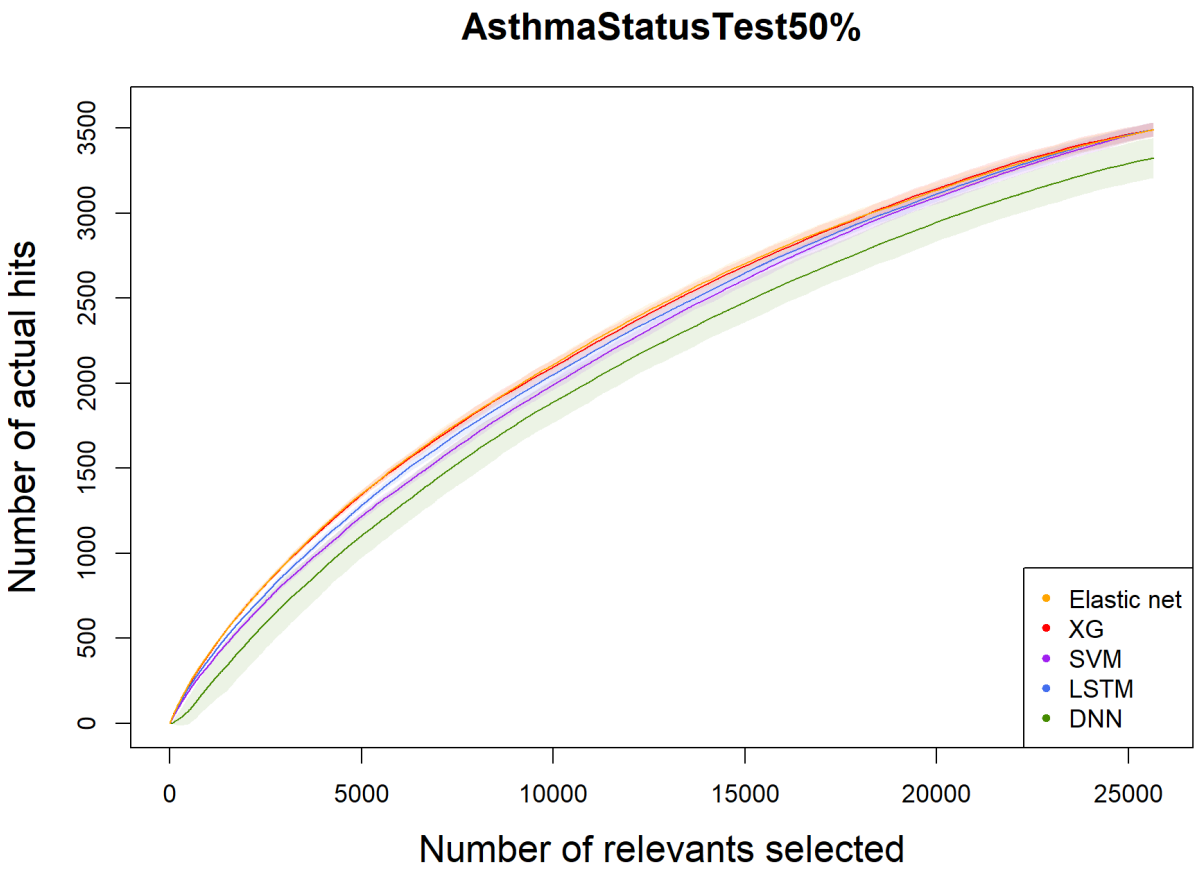


Figure B.5: Hit curve on asthma for 50% dataset.

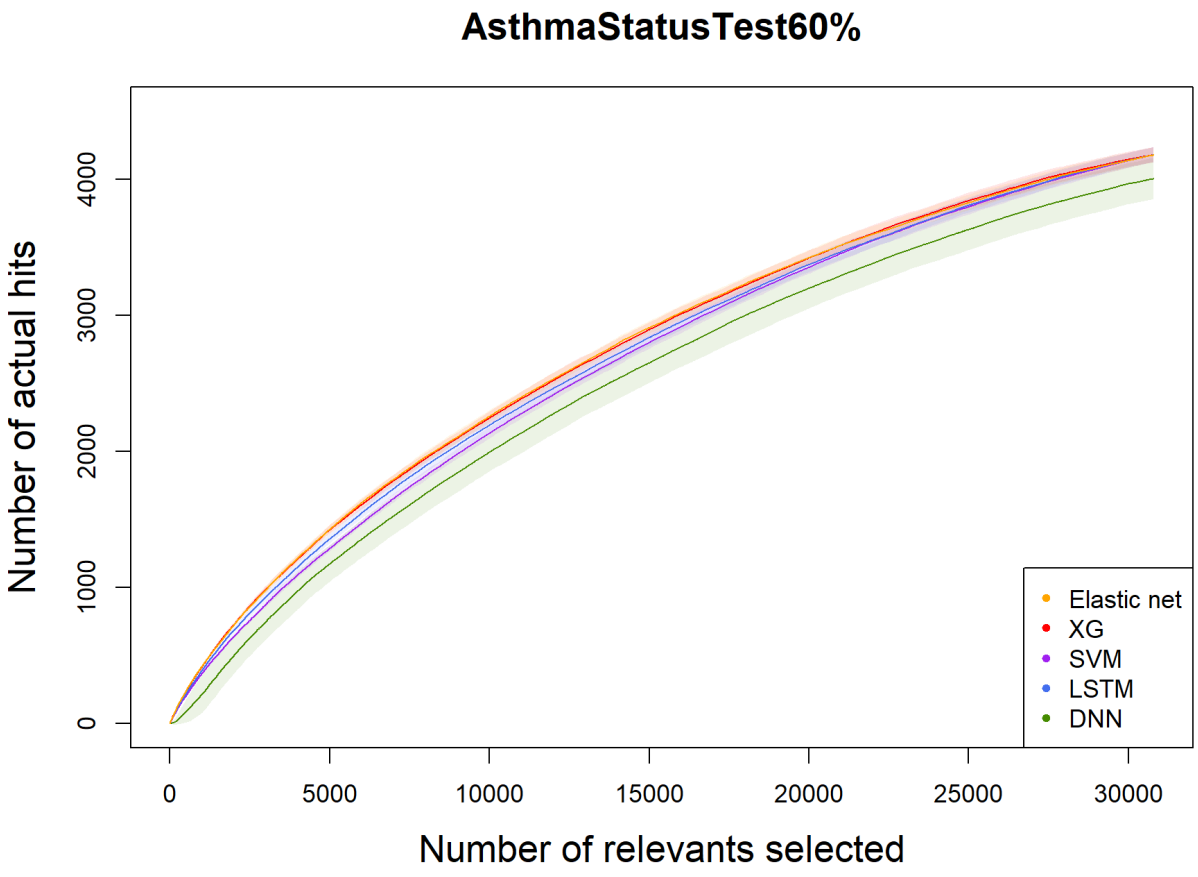


Figure B.6: Hit curve on asthma for 60% dataset.

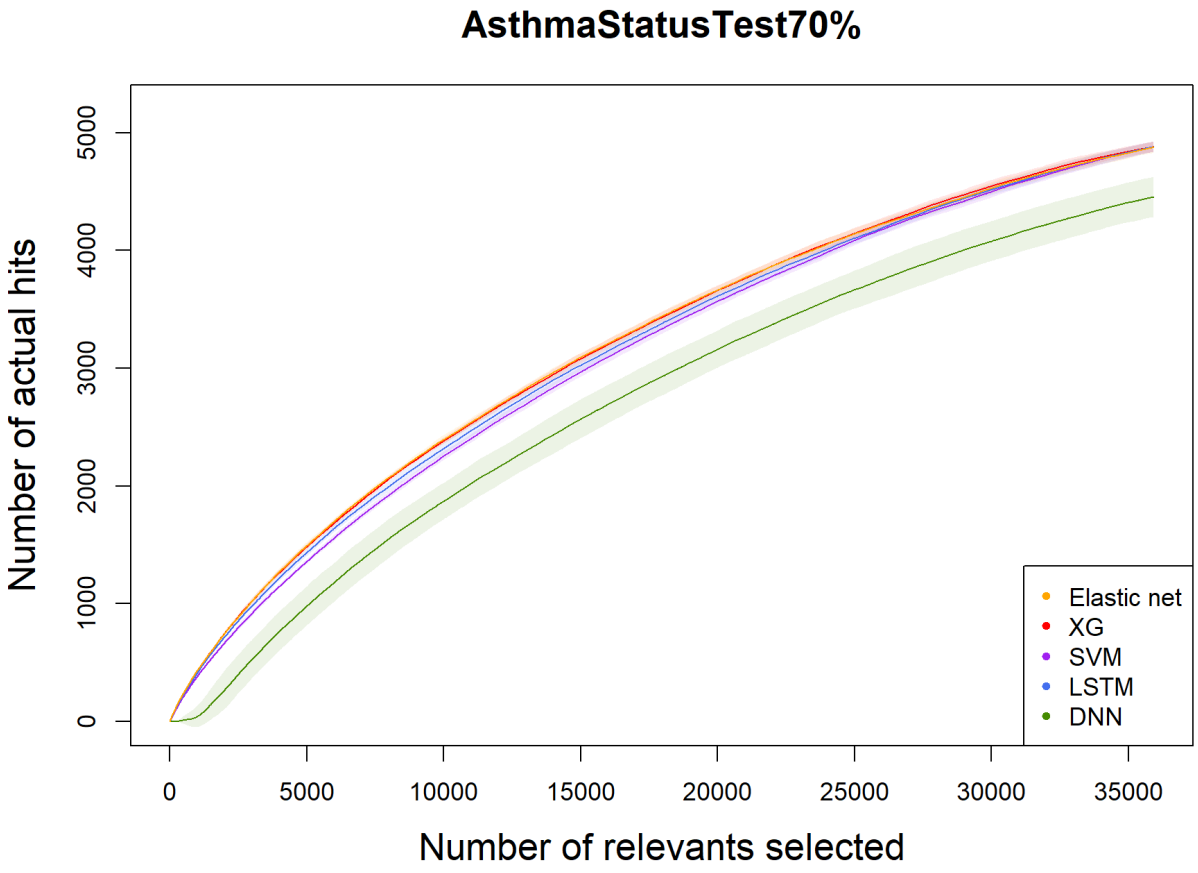


Figure B.7: Hit curve on asthma for 70% dataset.

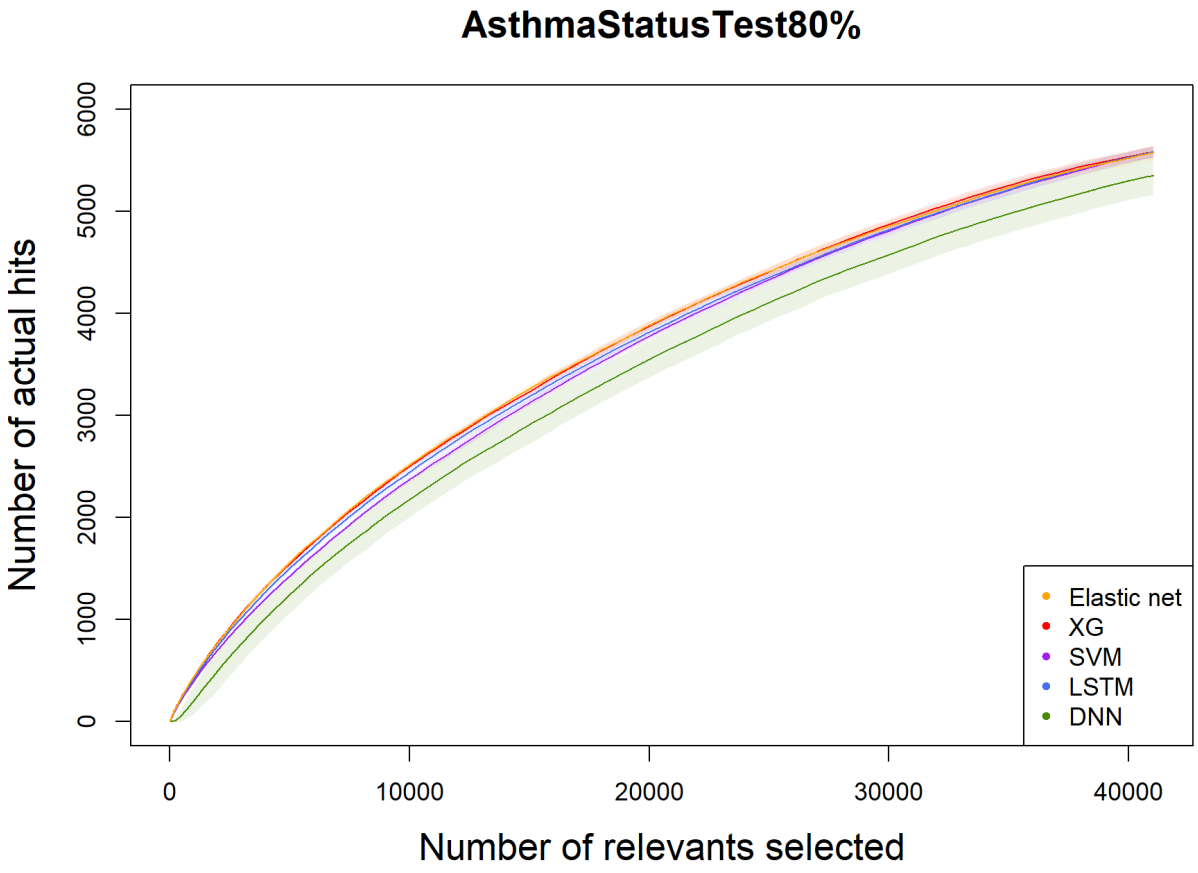


Figure B.8: Hit curve on asthma for 80% dataset.

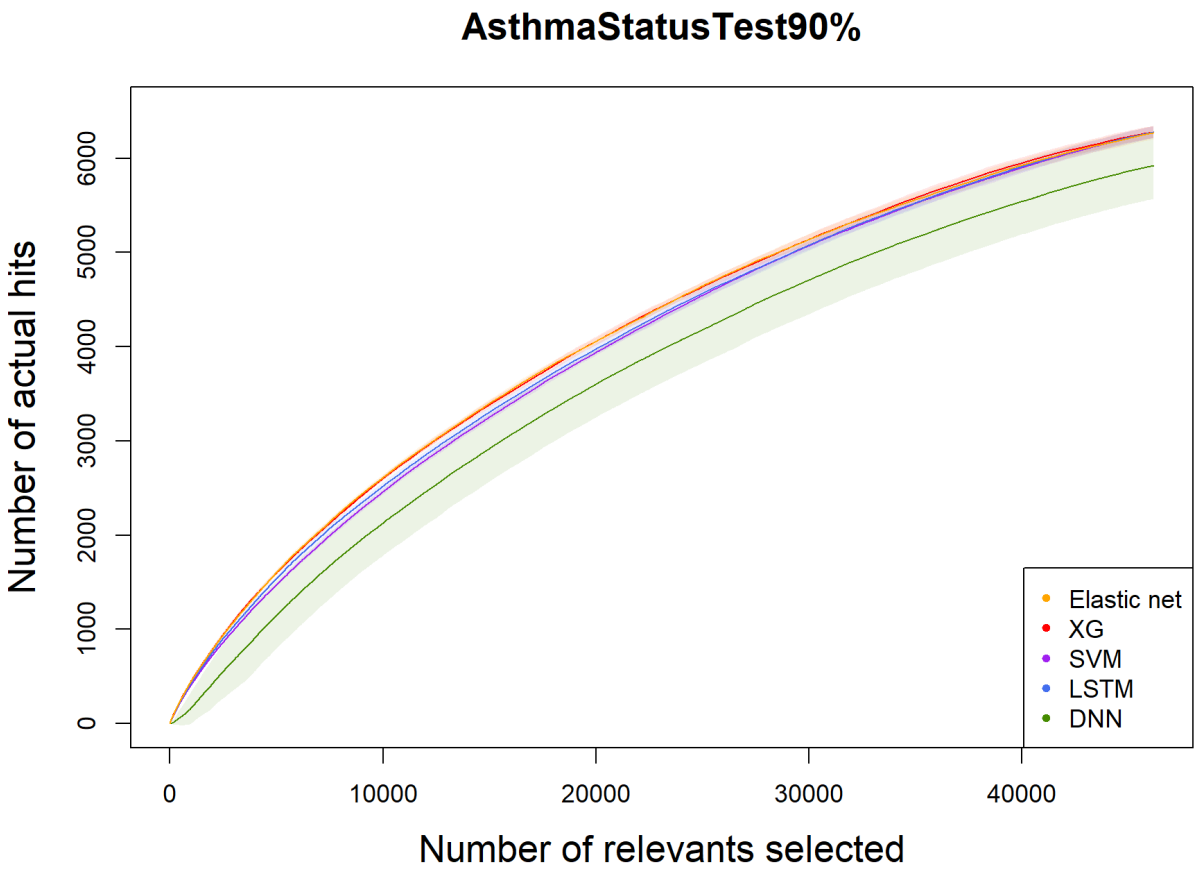


Figure B.9: Hit curve on asthma for 90% dataset.

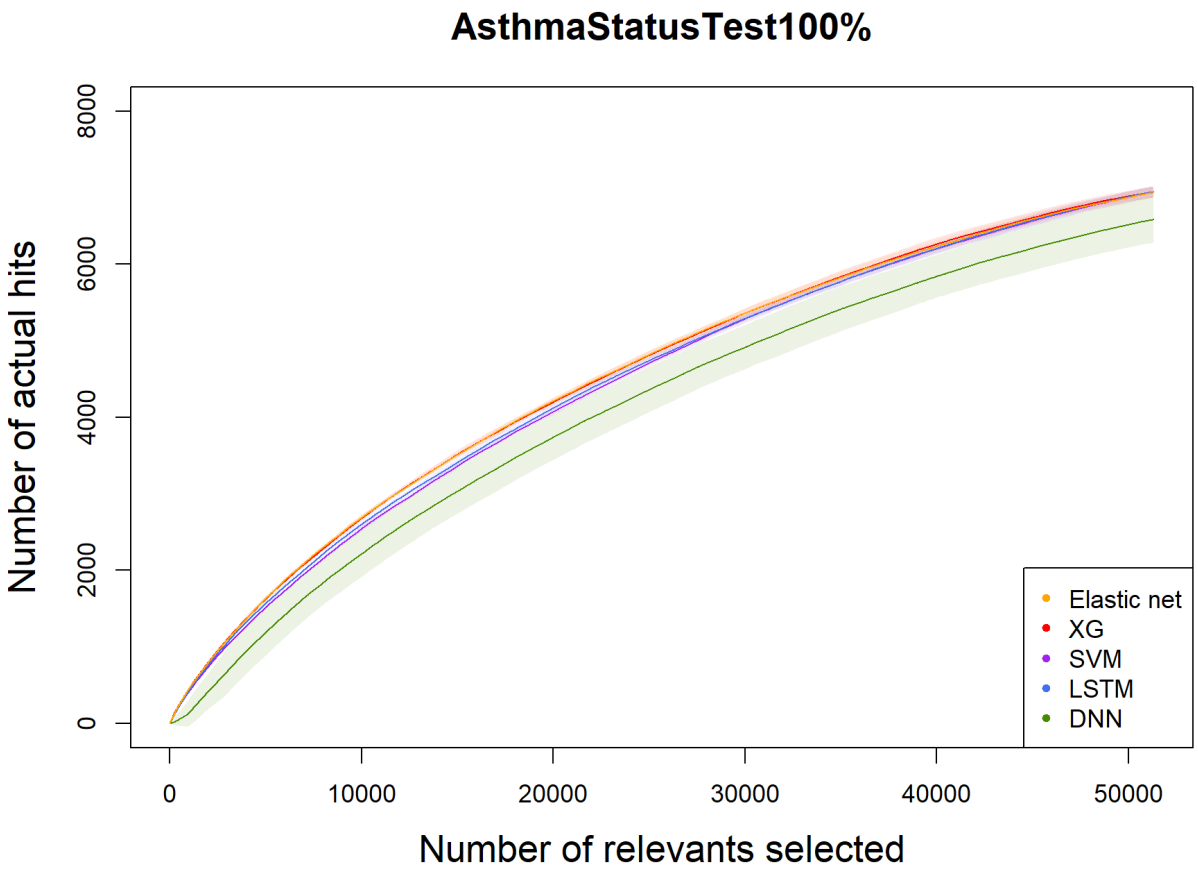


Figure B.10: Hit curve on asthma for 100% dataset.

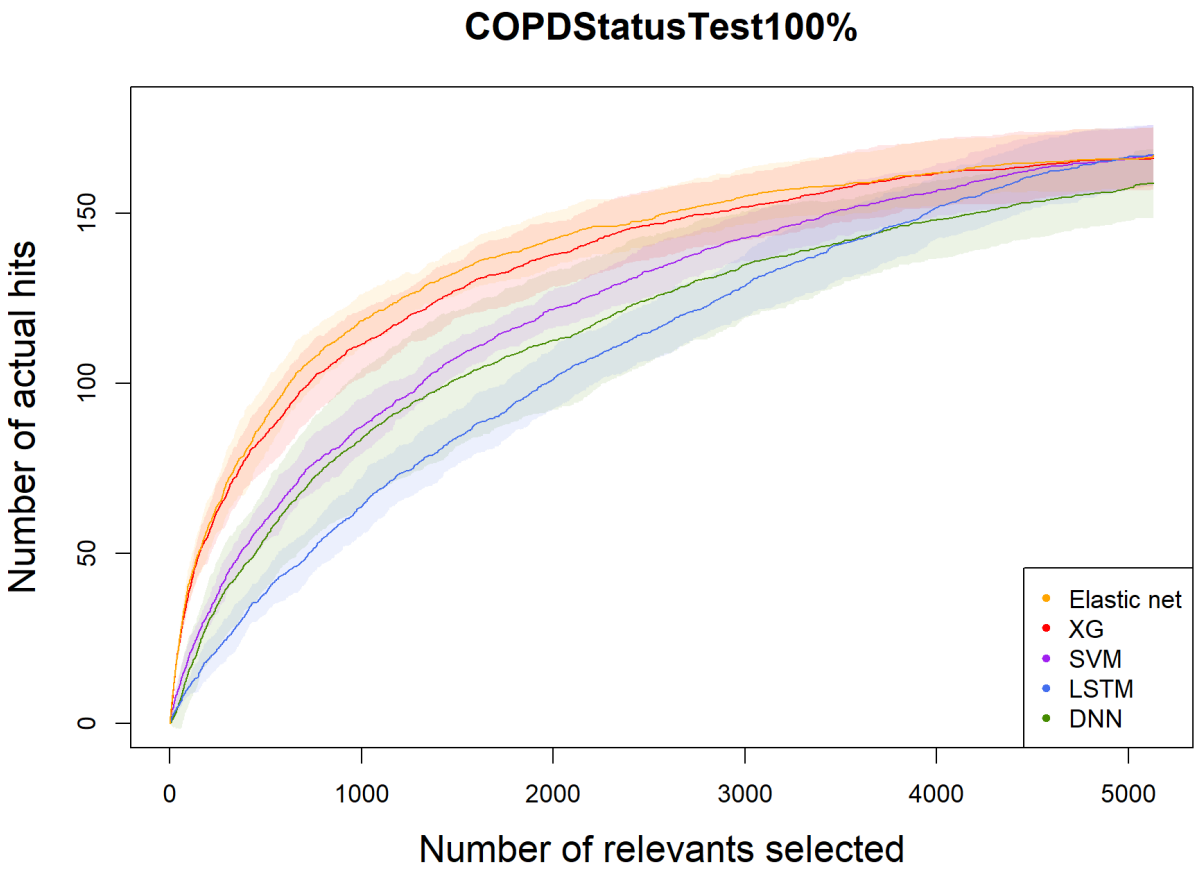


Figure B.11: Hit curve on COPD for 10% dataset.

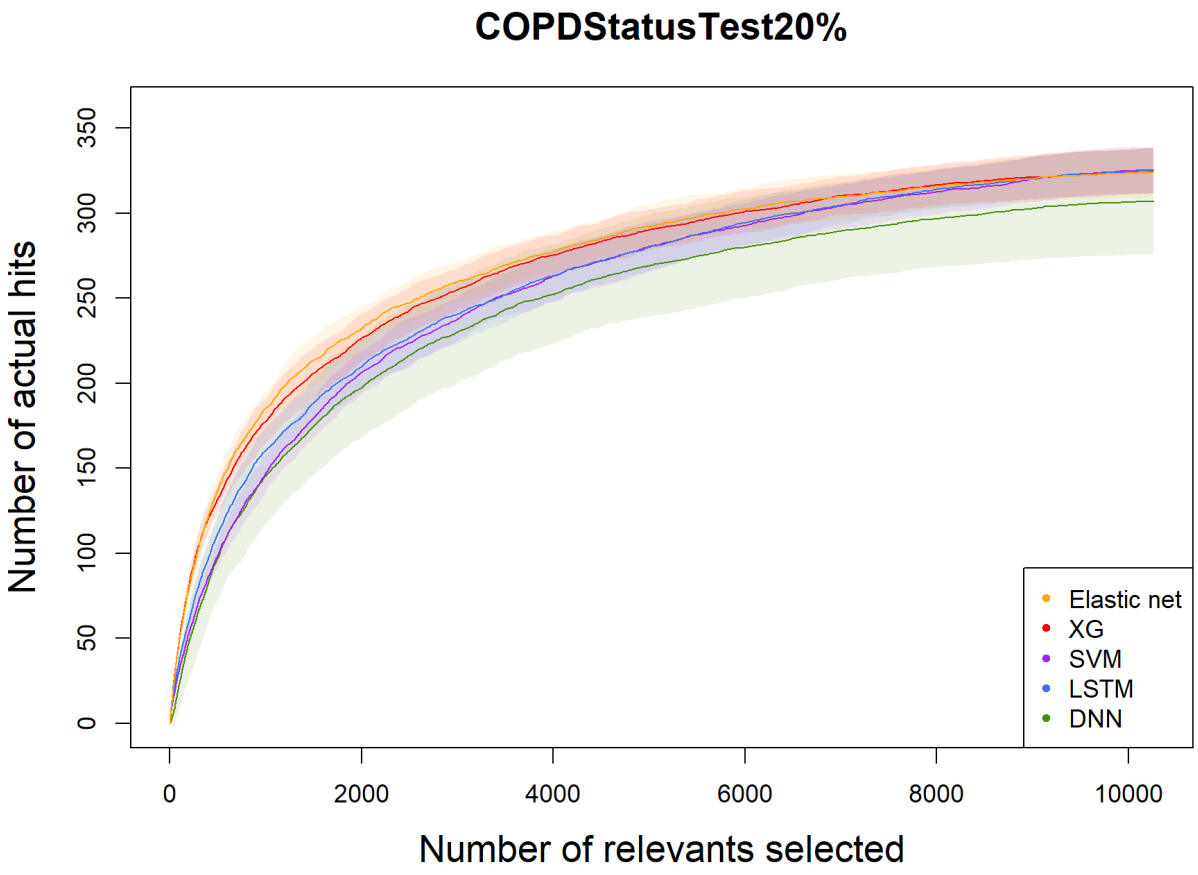


Figure B.12: Hit curve on COPD for 20% dataset.

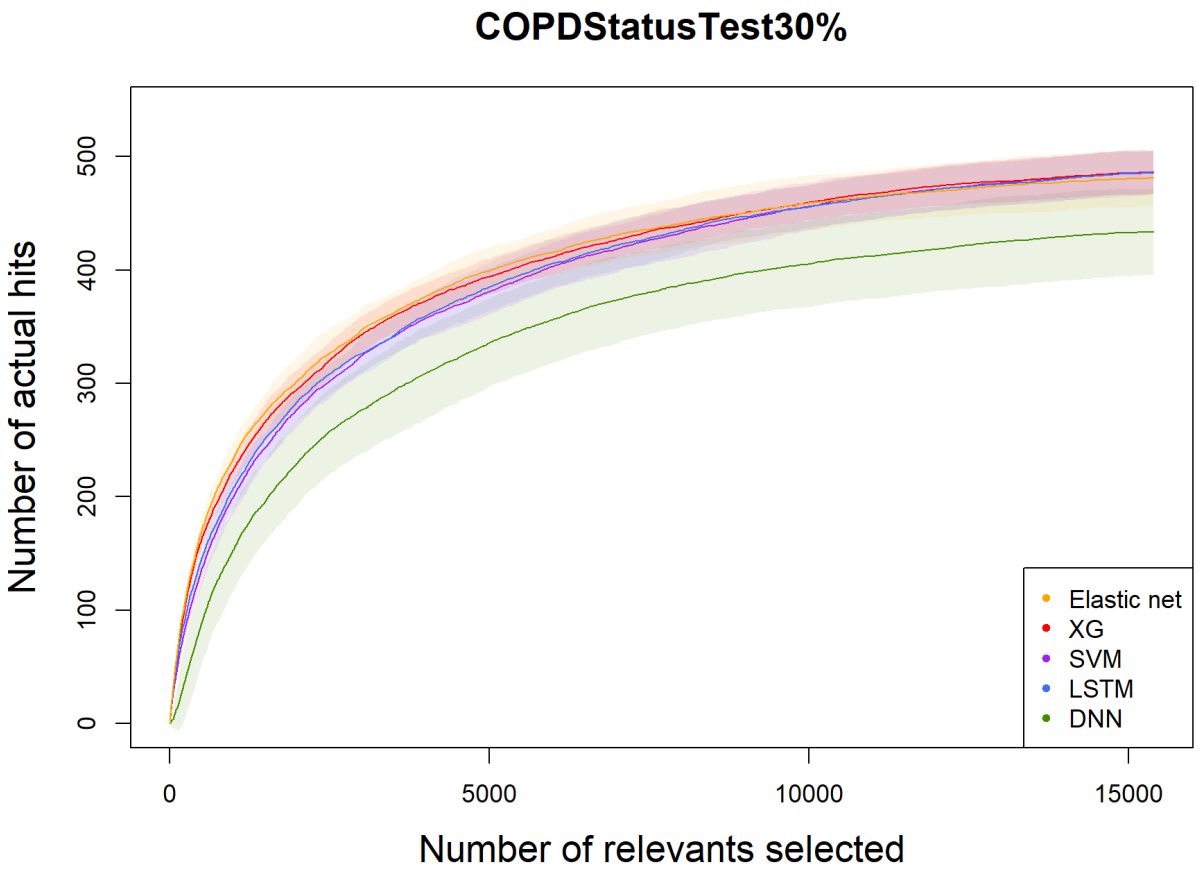


Figure B.13: Hit curve on COPD for 30% dataset.

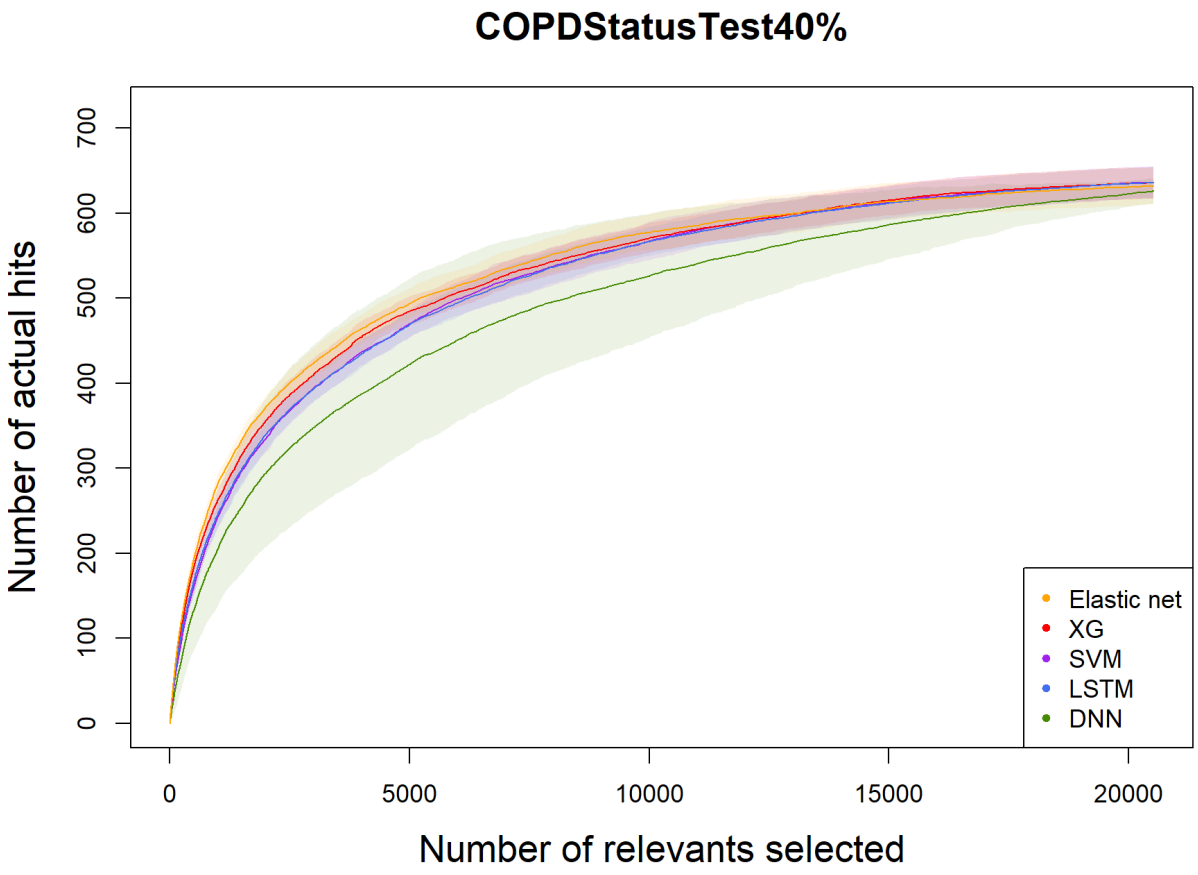


Figure B.14: Hit curve on COPD for 40% dataset.

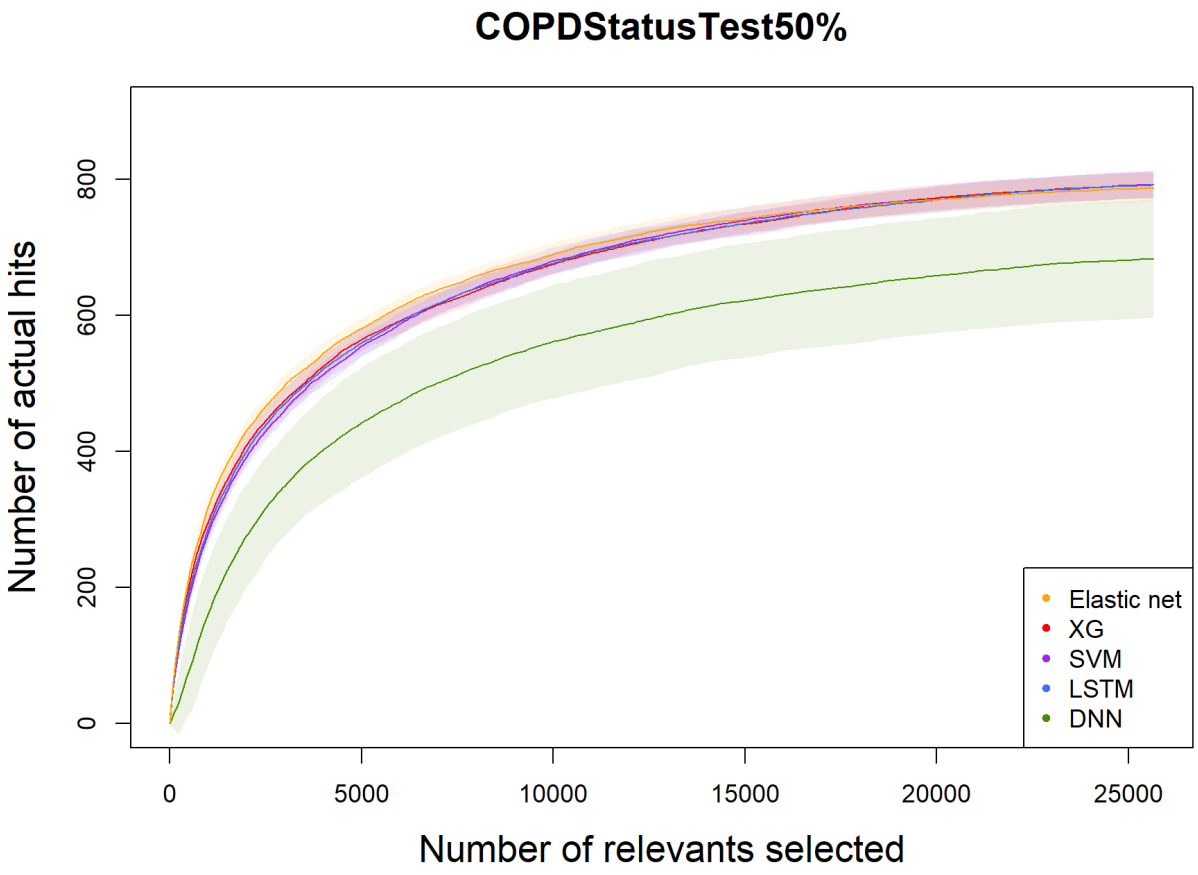


Figure B.15: Hit curve on COPD for 50% dataset.

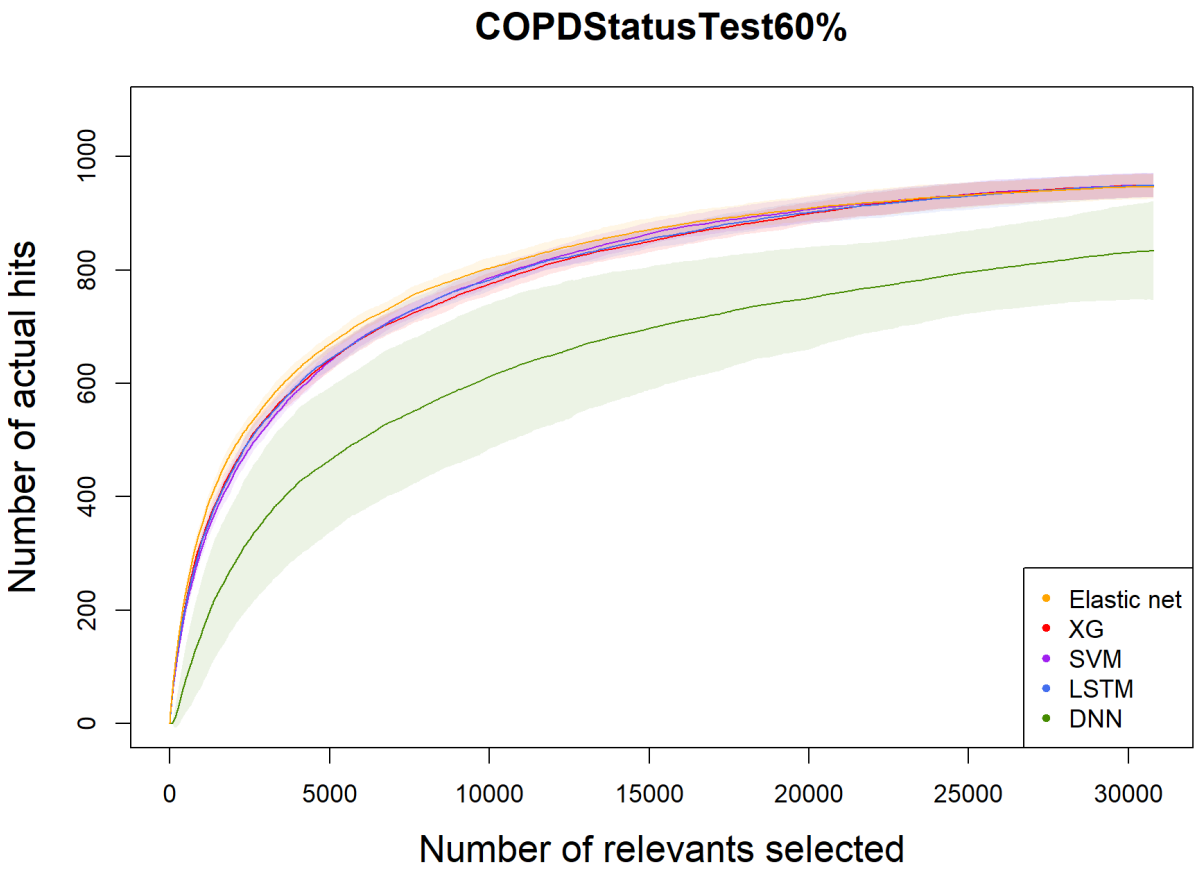


Figure B.16: Hit curve on COPD for 60% dataset.

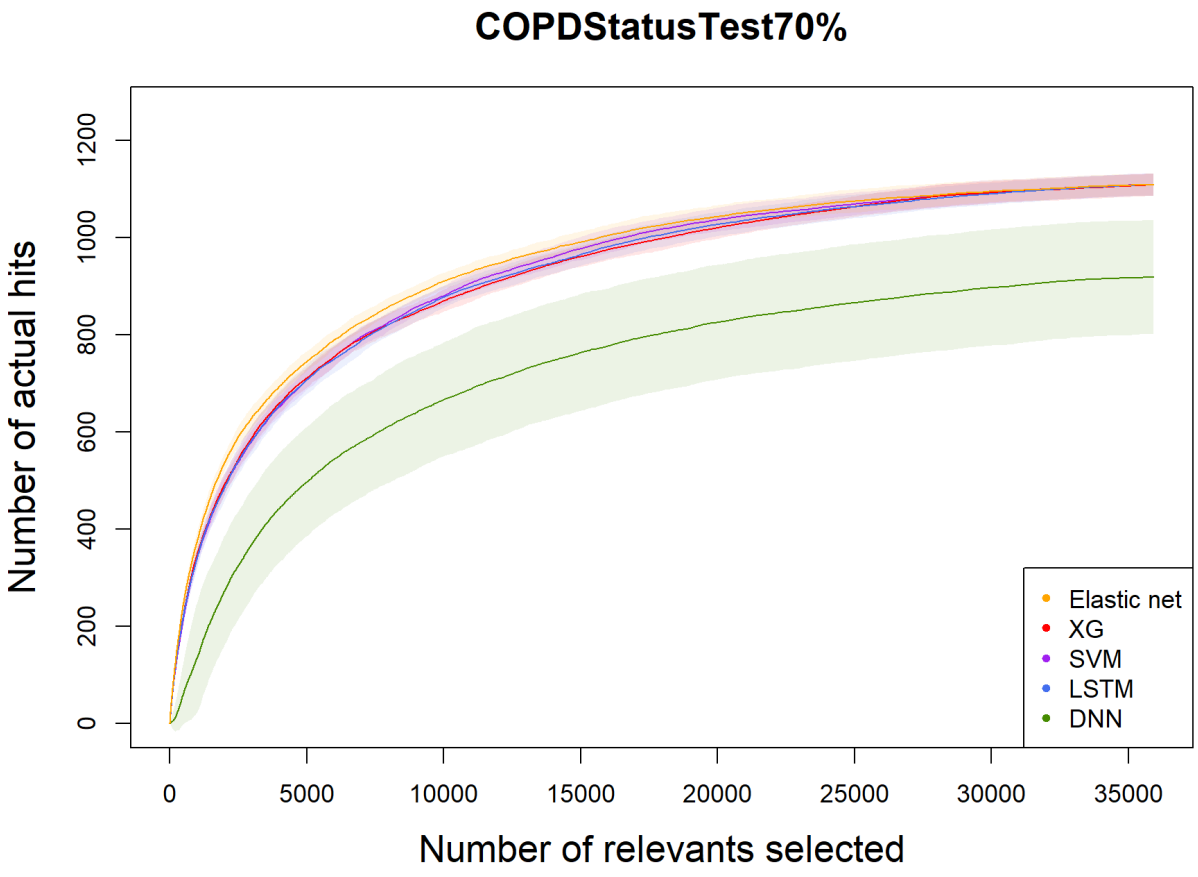


Figure B.17: Hit curve on COPD for 70% dataset.

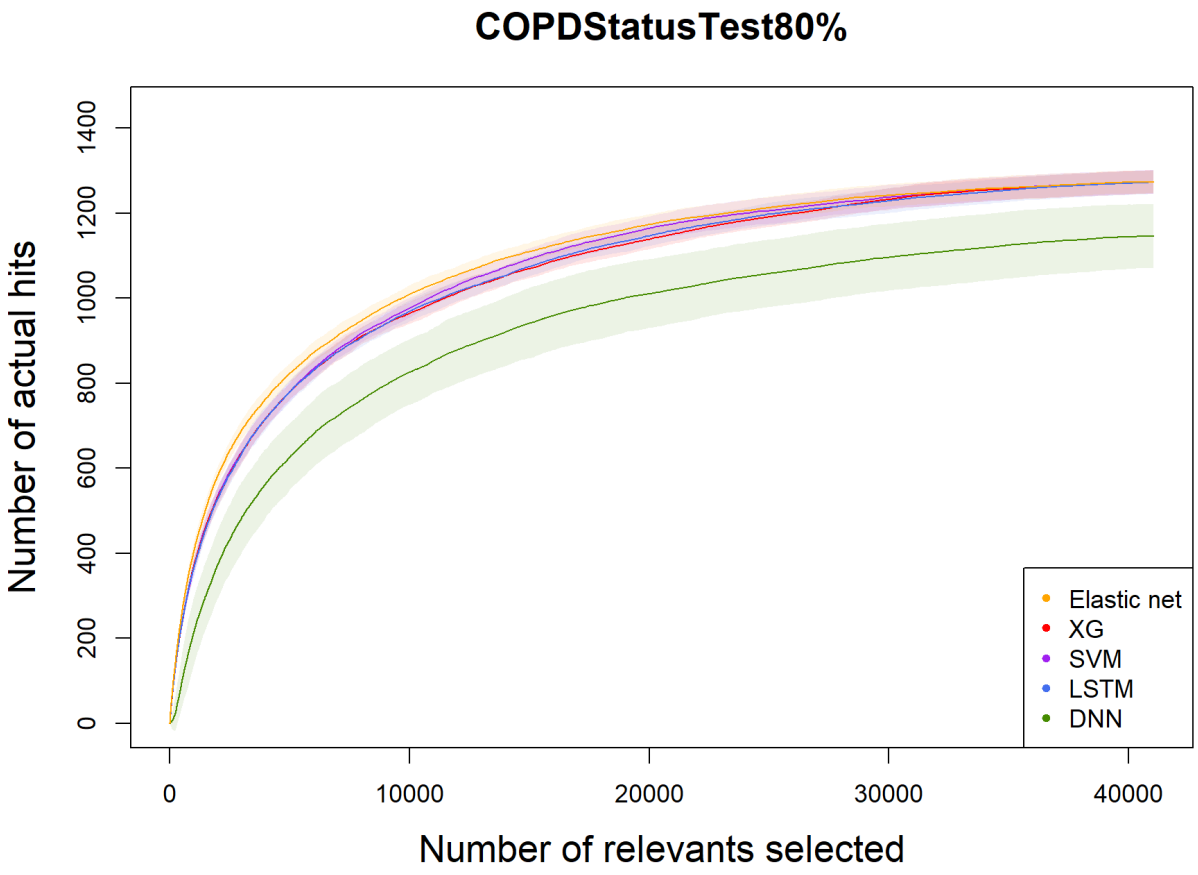


Figure B.18: Hit curve on COPD for 80% dataset.

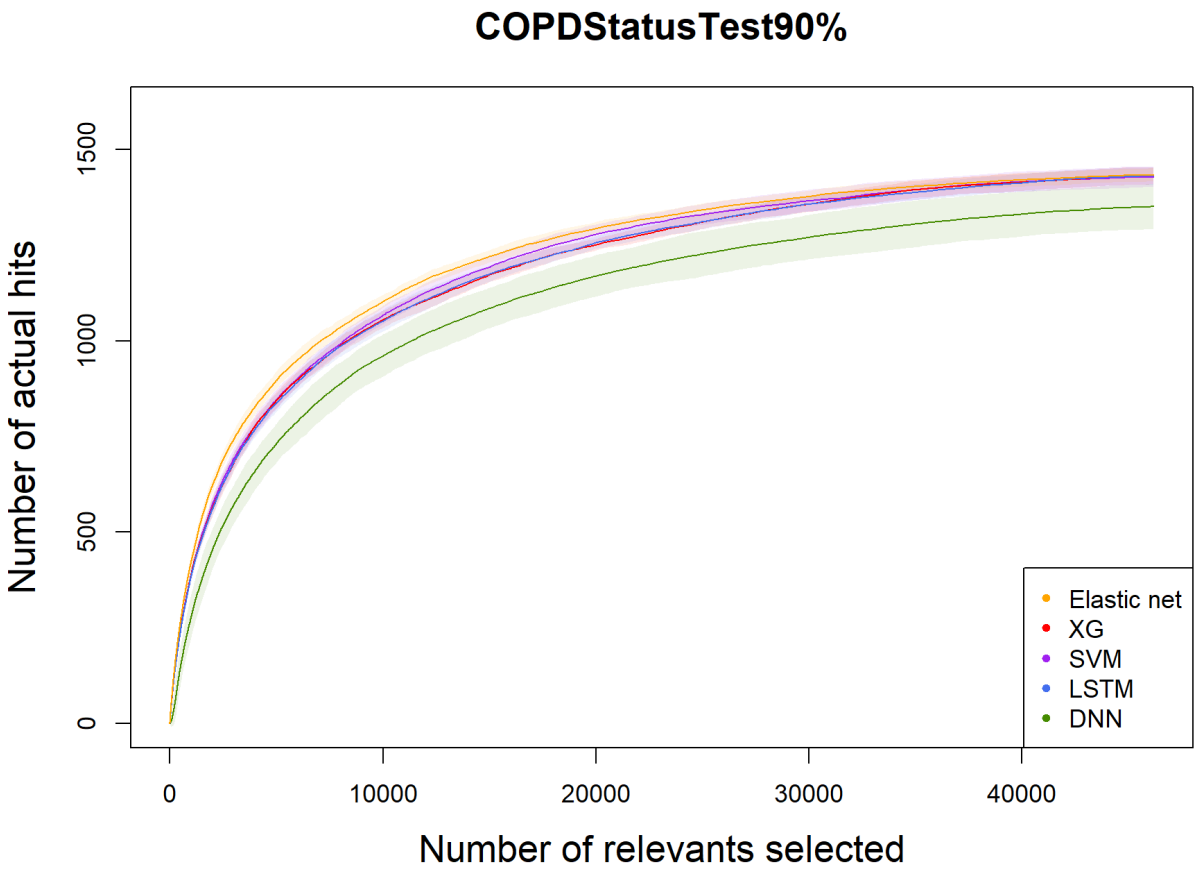


Figure B.19: Hit curve on COPD for 90% dataset.

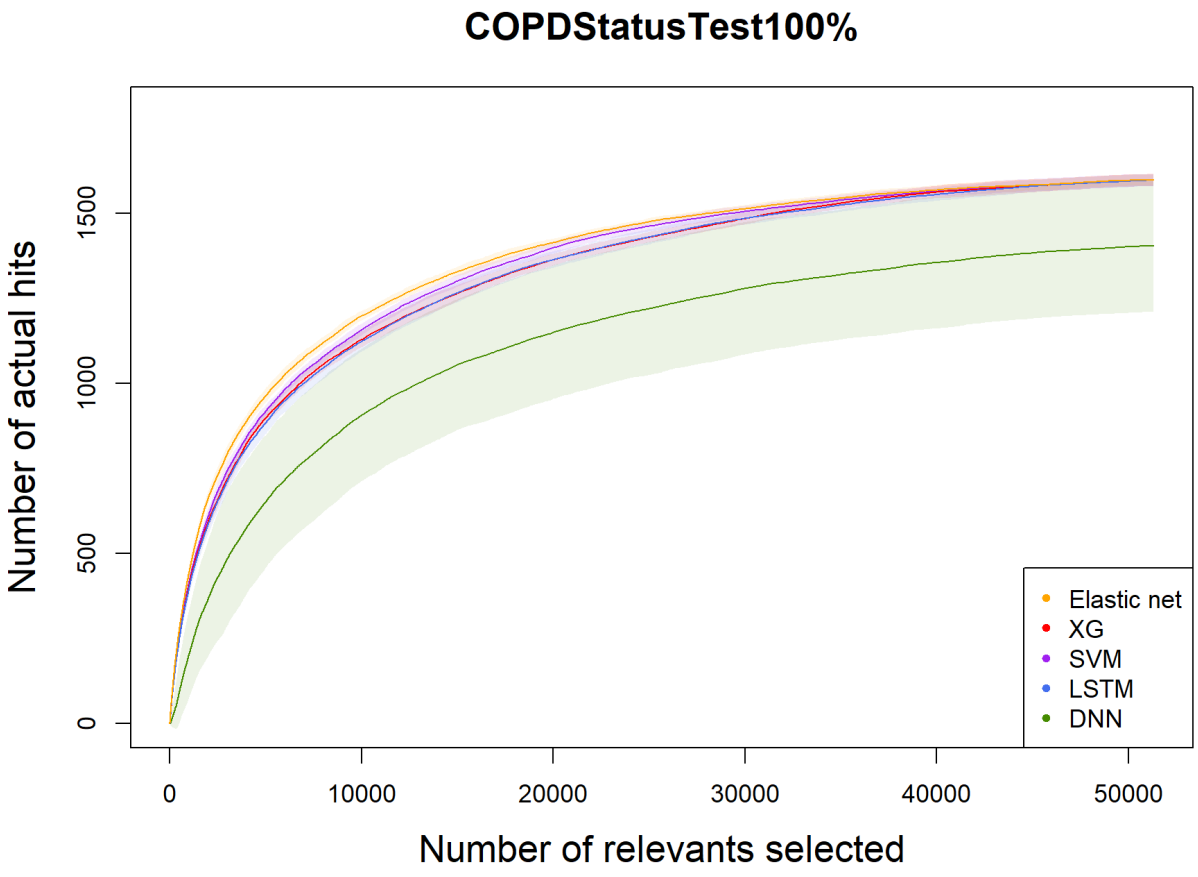


Figure B.20: Hit curve on COPD for 100% dataset.

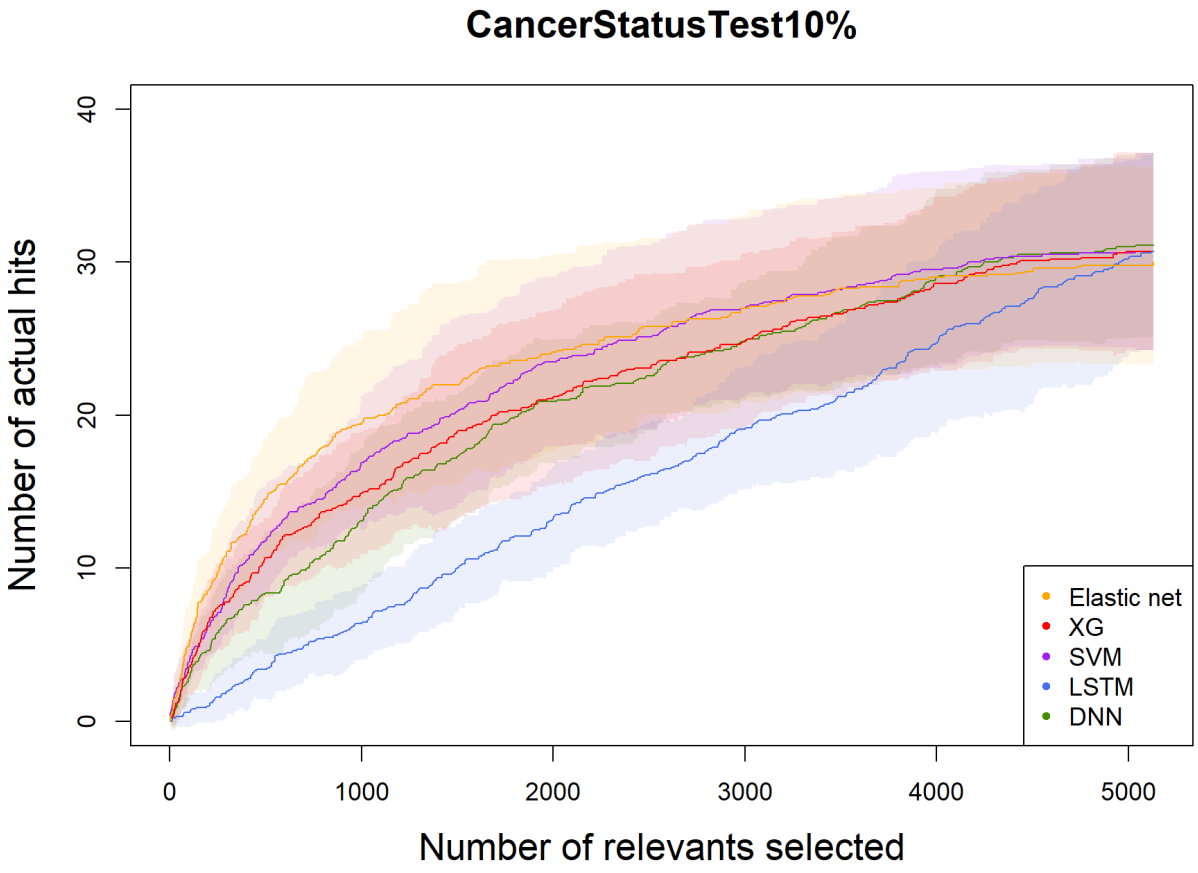


Figure B.21: Hit curve on Cancer for 10% dataset.

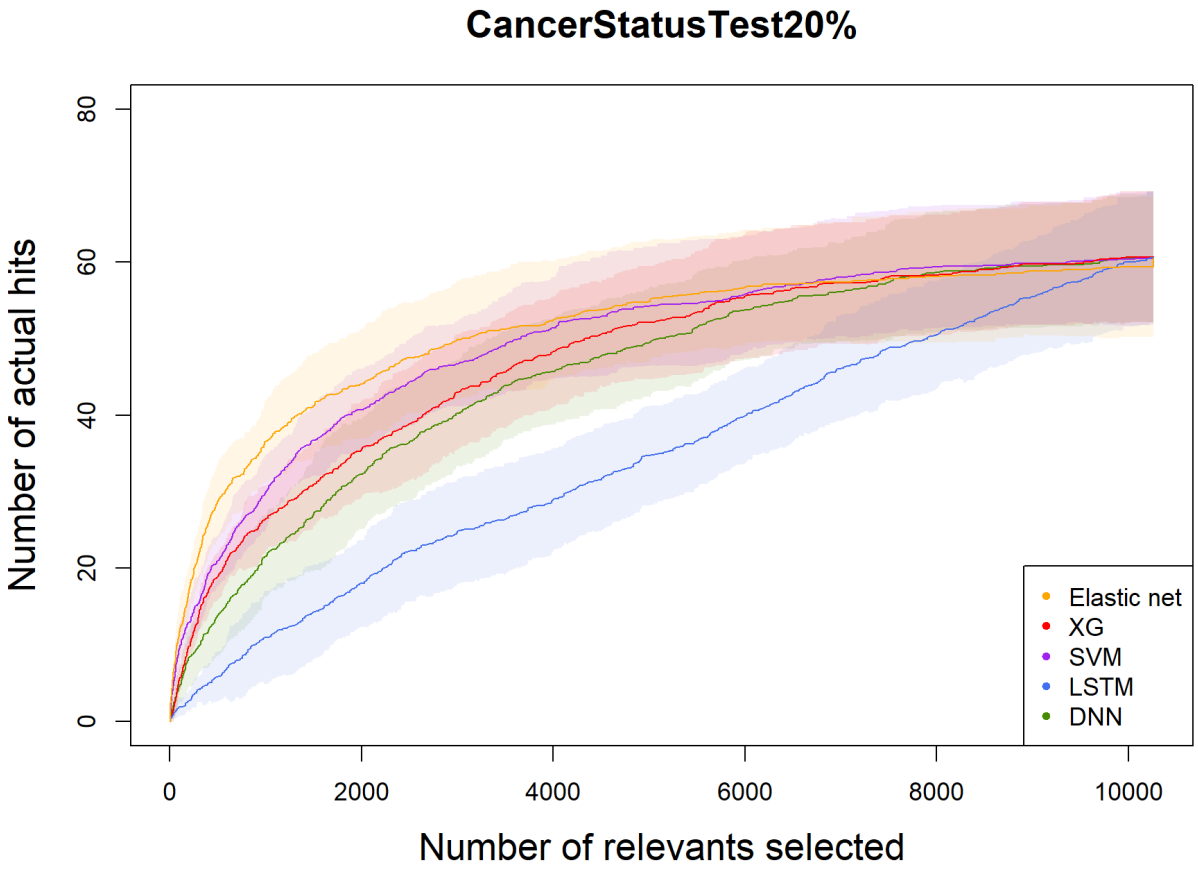


Figure B.22: Hit curve on Cancer for 20% dataset.

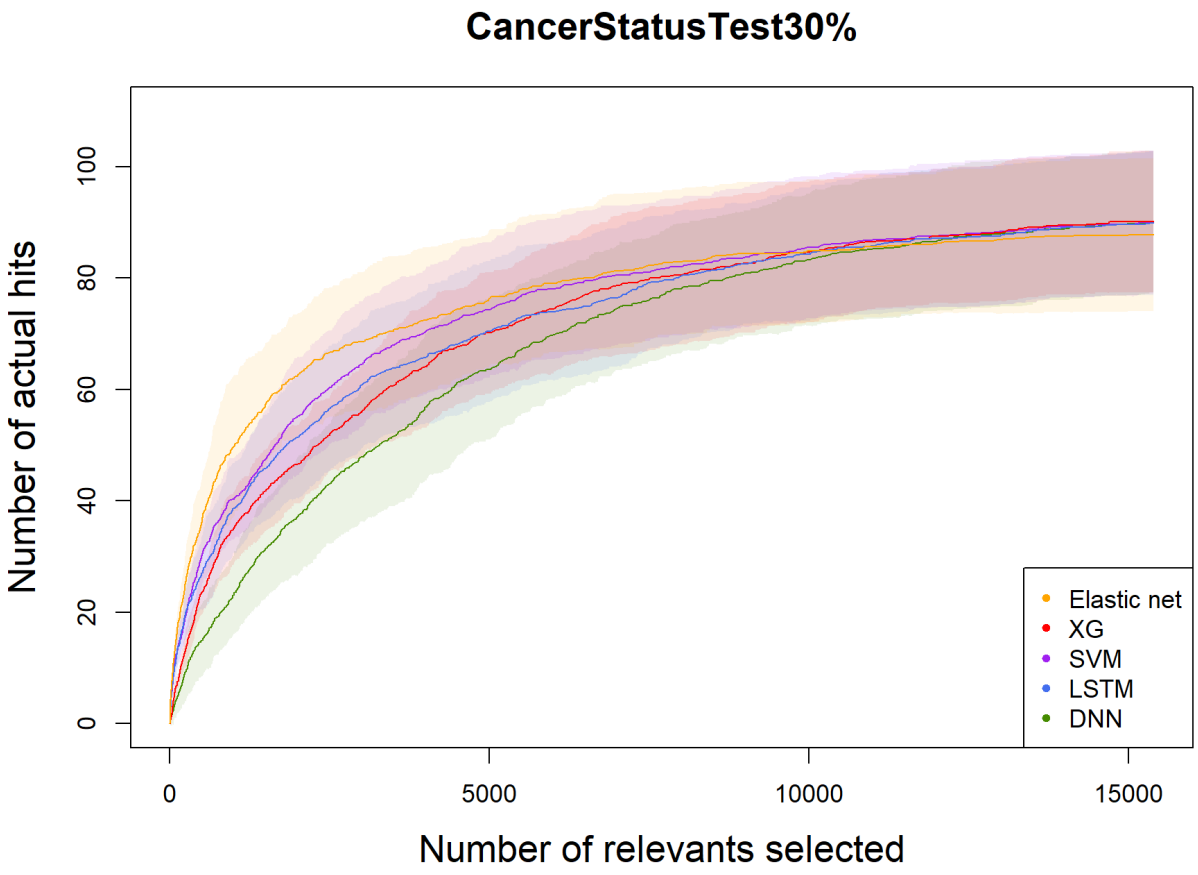


Figure B.23: Hit curve on Cancer for 30% dataset.

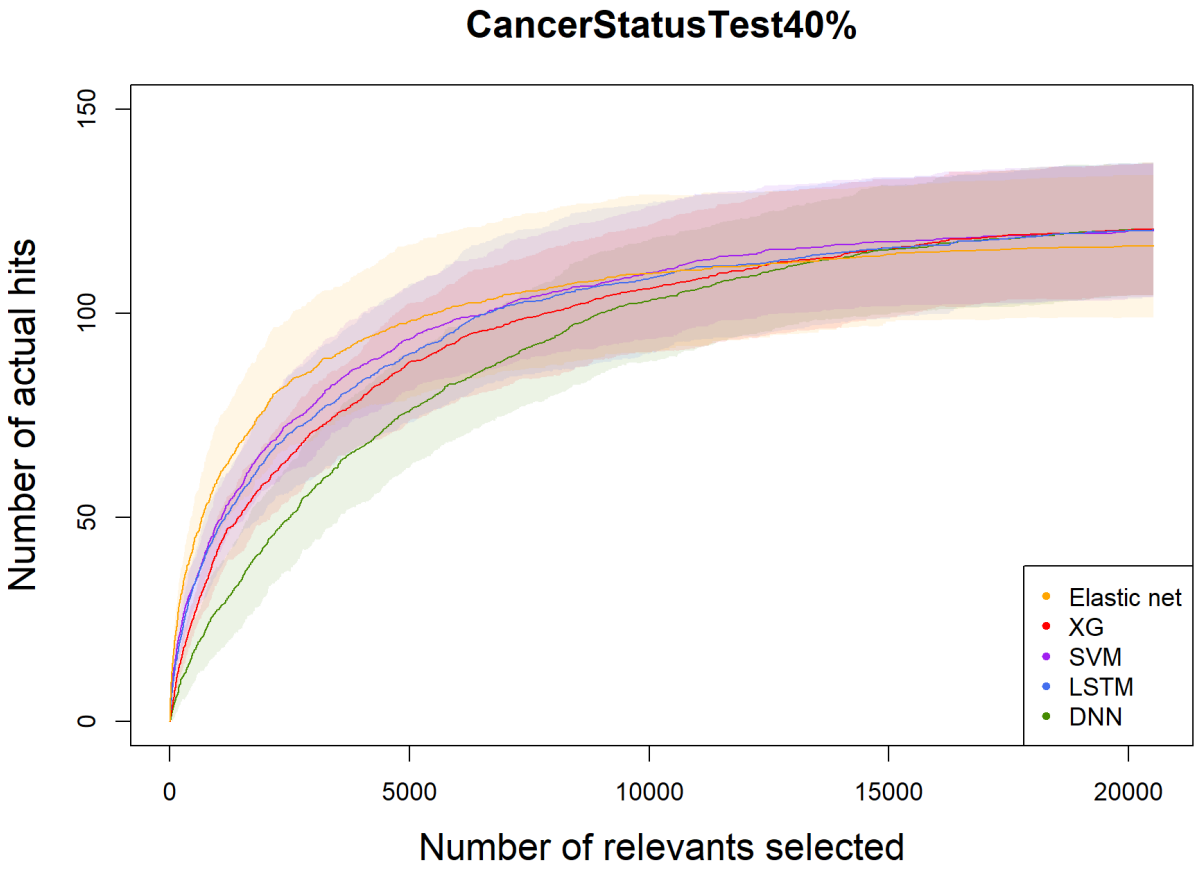


Figure B.24: Hit curve on Cancer for 40% dataset.

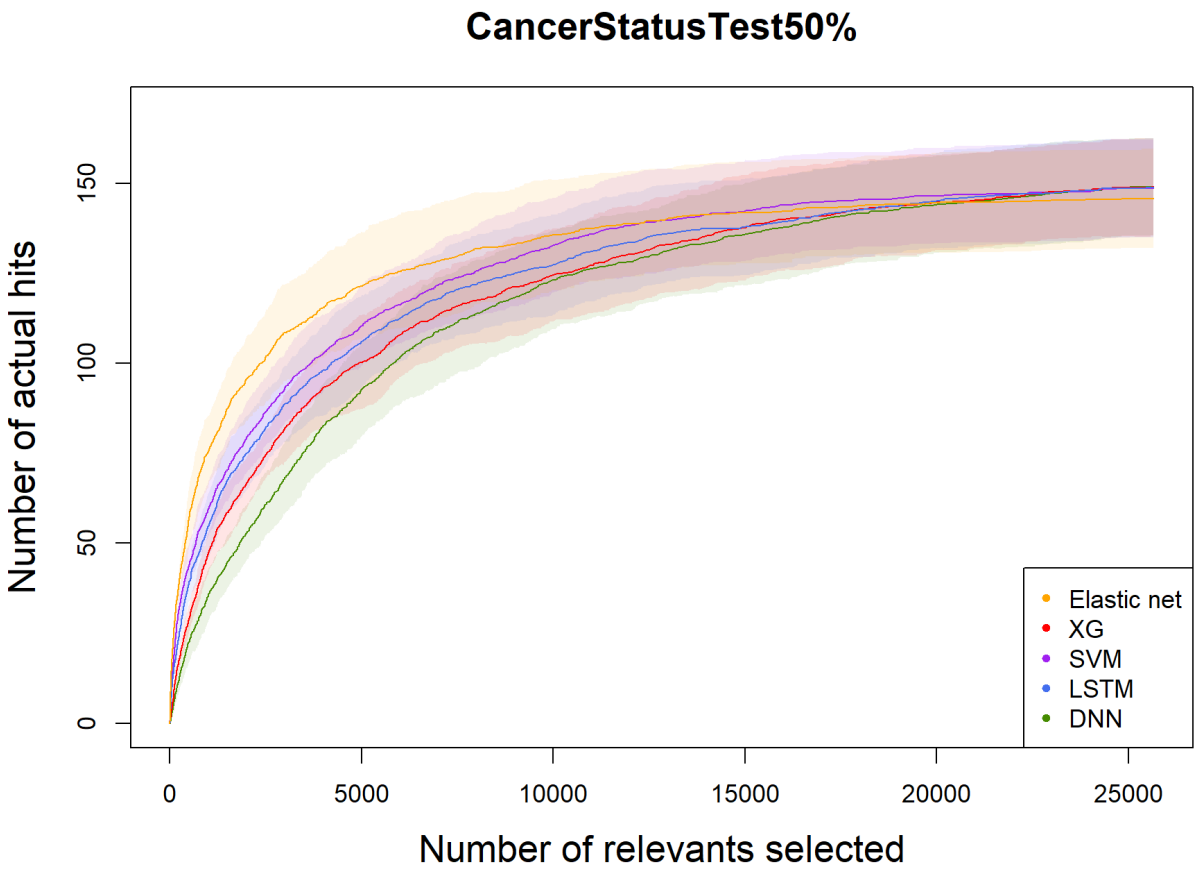


Figure B.25: Hit curve on Cancer for 50% dataset.

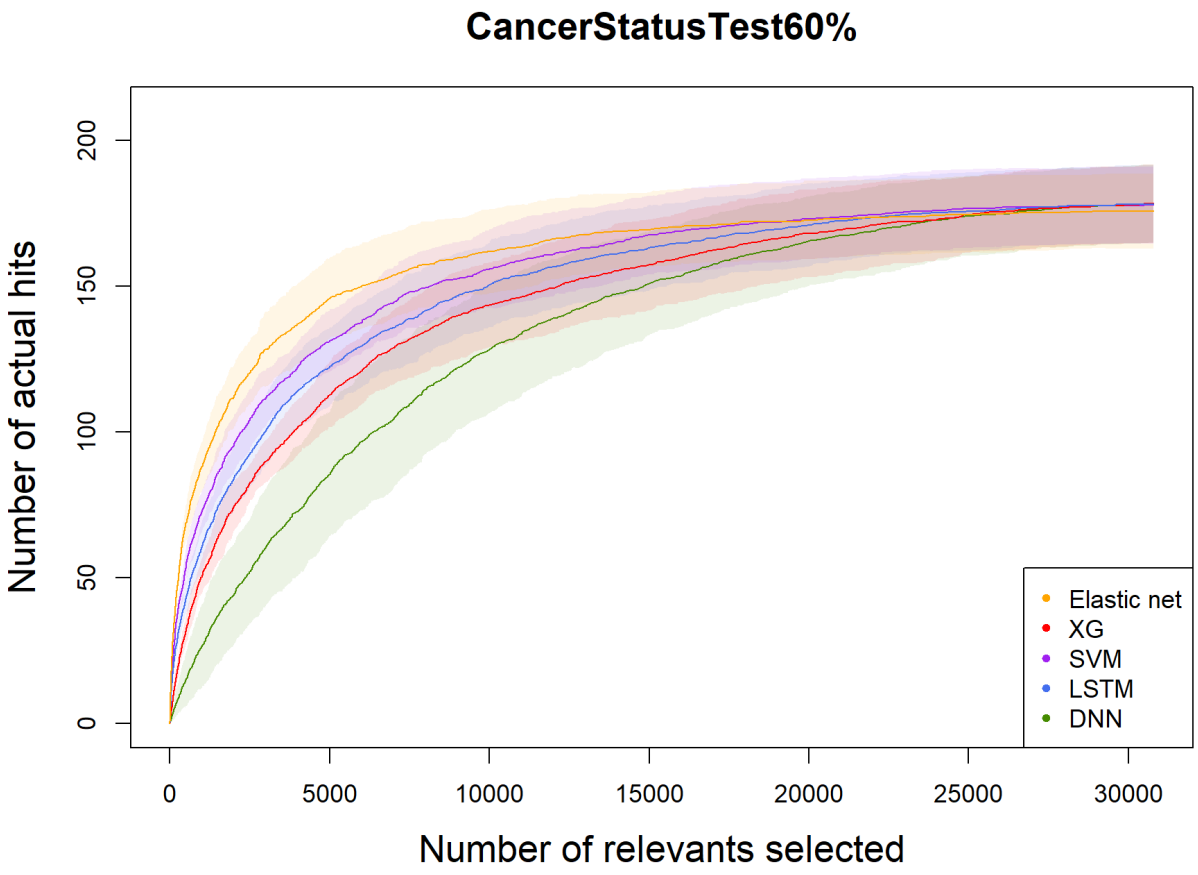


Figure B.26: Hit curve on Cancer for 60% dataset.

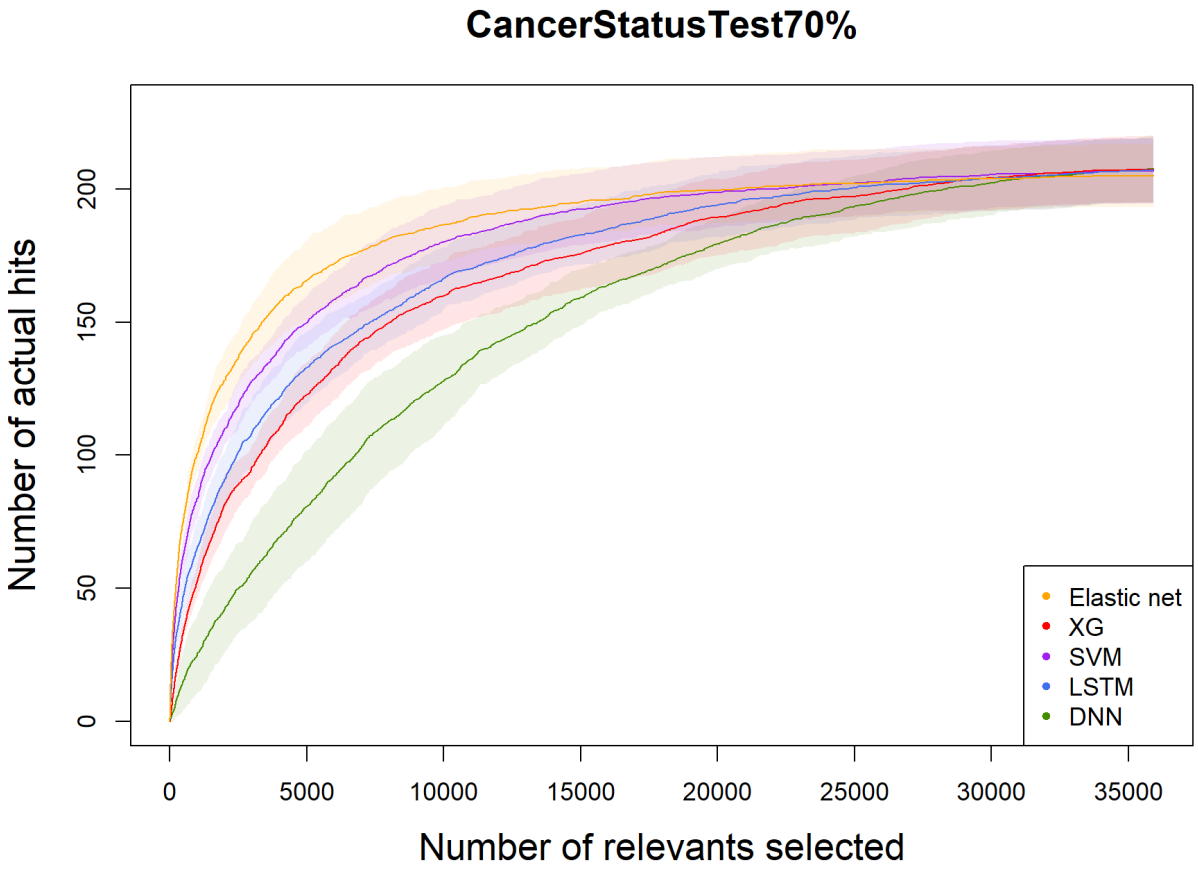


Figure B.27: Hit curve on Cancer for 70% dataset.

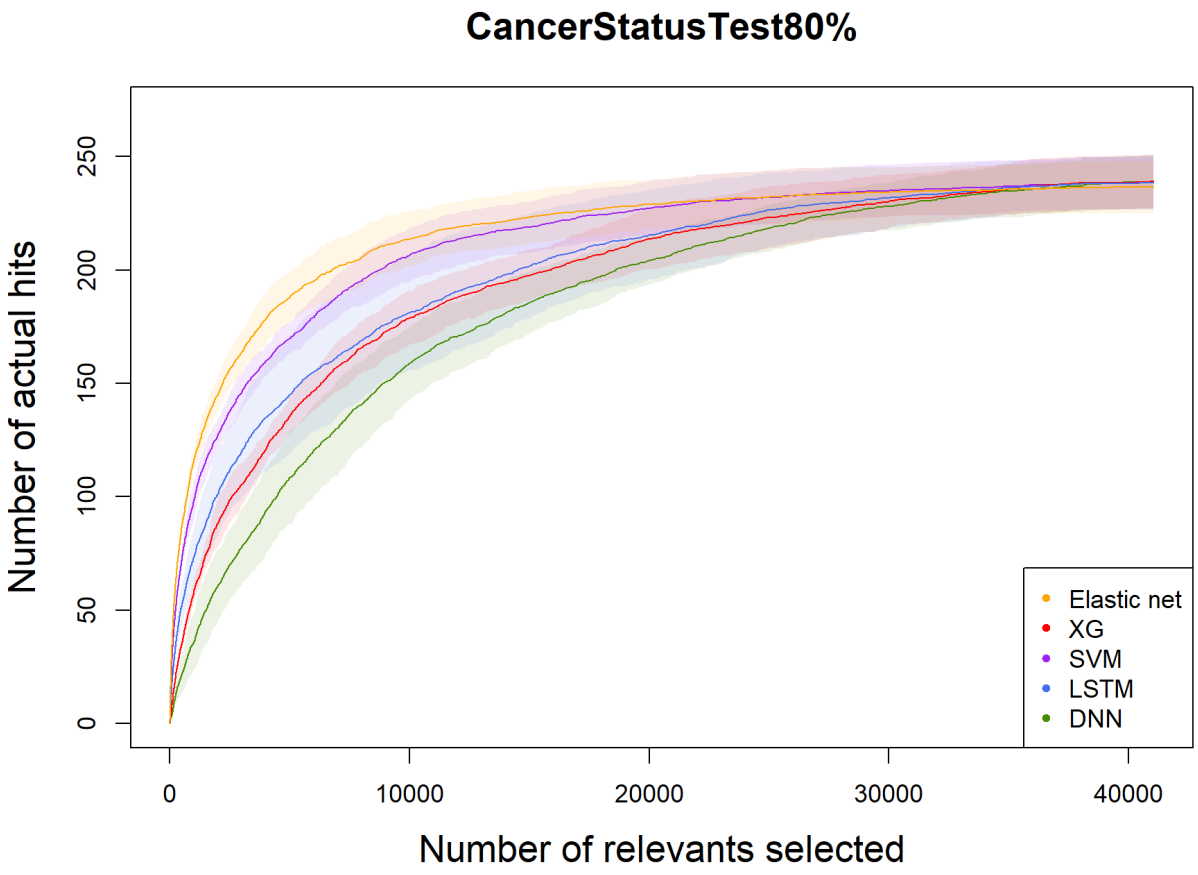


Figure B.28: Hit curve on Cancer for 80% dataset.

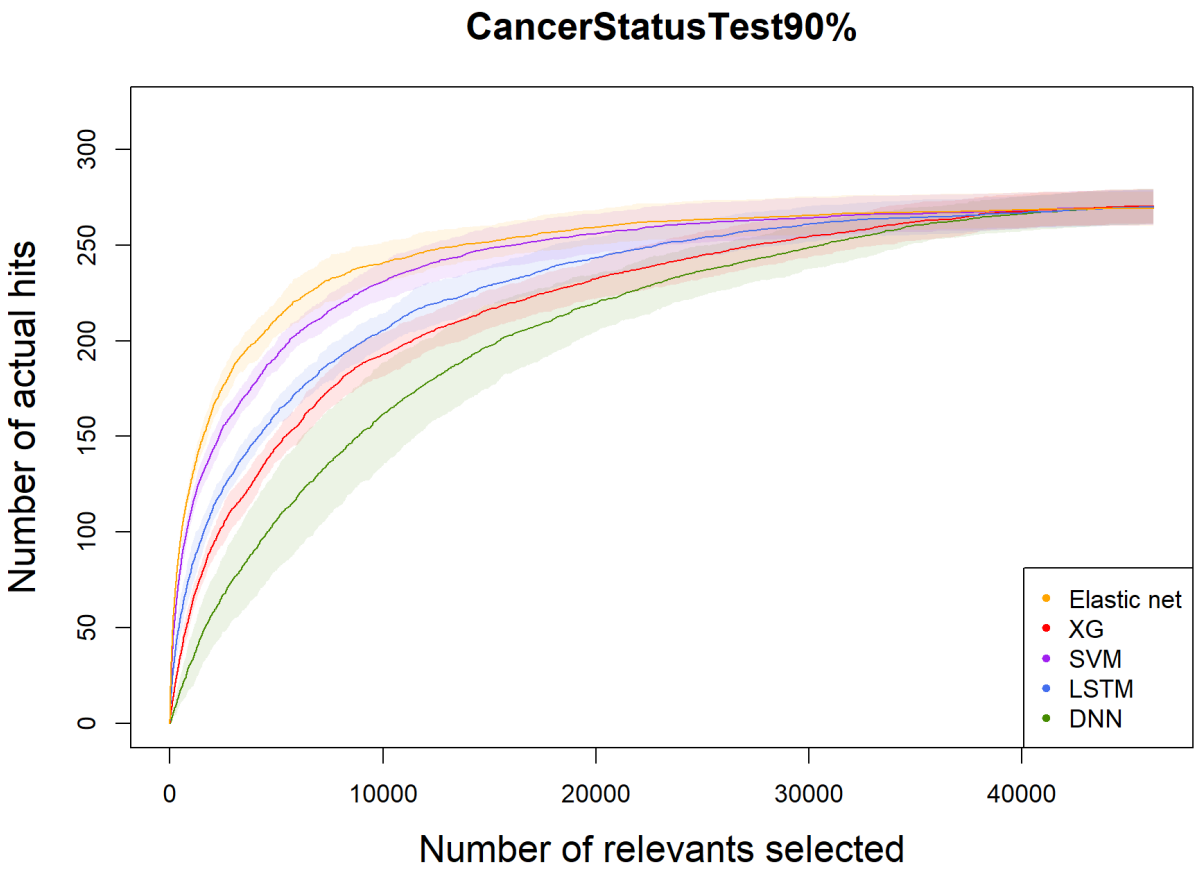


Figure B.29: Hit curve on Cancer for 90% dataset.

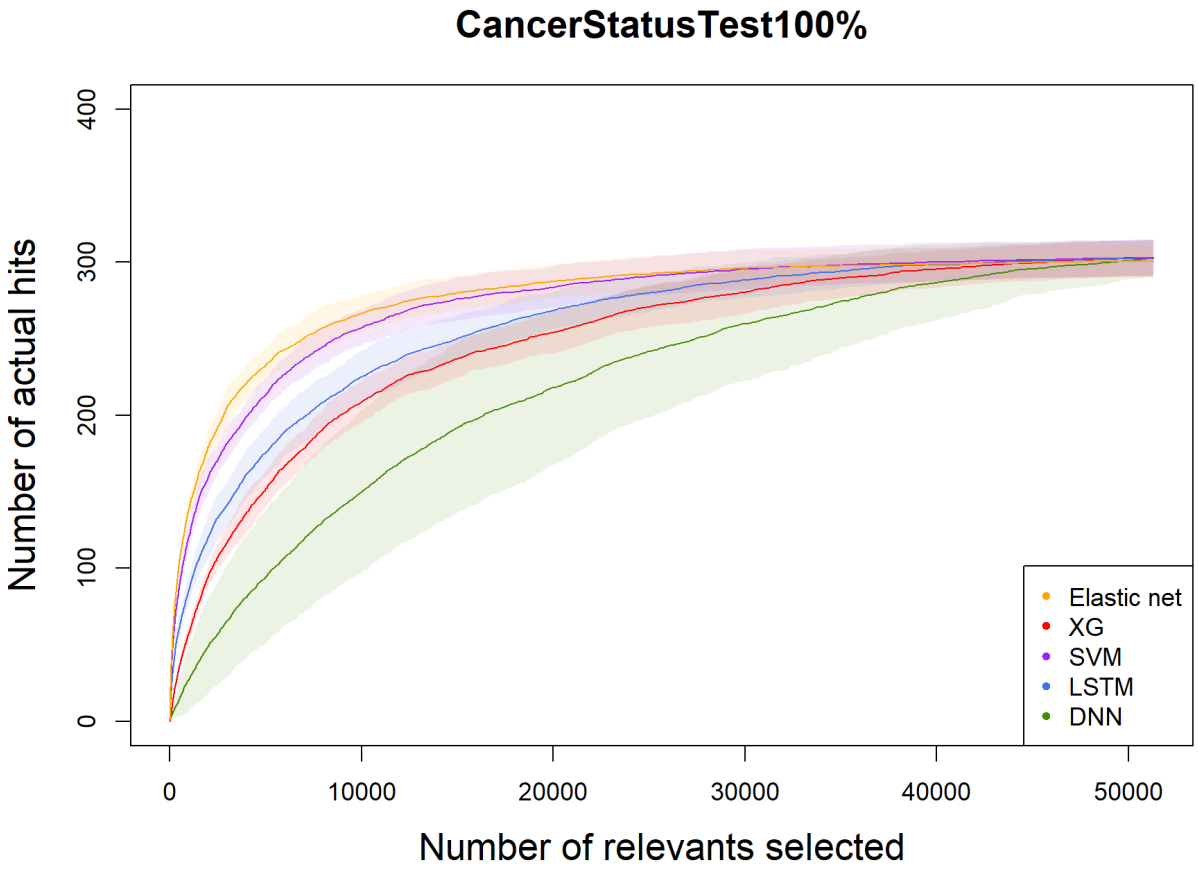


Figure B.30: Hit curve on Cancer for 100% dataset.