On Predicting the Outcomes of Chemotherapy Treatments in Breast Cancer*

Agastya Silvina[1], Juliana Bowles[1][0000−0002−5918−9114], and Peter Hall[2][0000−0001−6015−7841]

[1] School of Computer Science, University of St Andrews
St Andrews KY16 9SX, UK
{as362|jkfb}@st-andrews.ac.uk
[2] Edinburgh Cancer Research Centre, University of Edinburgh
Western General Hospital, Crewe Road South, Edinburgh EH4 2XR, UK
p.s.hall@ed.ac.uk

Abstract. Chemotherapy is the main treatment commonly used for treating cancer patients. However, chemotherapy usually causes side effects some of which can be severe. The effects depend on a variety of factors including the type of drugs used, dosage, length of treatment and patient characteristics. In this paper, we use a data extraction from an oncology department in Scotland with information on treatment cycles, recorded toxicity level, and various observations concerning breast cancer patients for three years. The objective of our paper is to compare several different techniques applied to the same data set to predict the toxicity outcome of the treatment. We use a Markov model, Hidden Markov model, Random Forest and Recurrent Neural Network in our comparison. Through analysis and evaluation of the performance of these techniques, we can determine which method is more suitable in different situations to assist the medical oncologist in real-time clinical practice. We discuss the context of our work more generally and further work.

Keywords: Breast Cancer Data· Toxicity Prediction · Modelling · Machine Learning

1 Introduction

Cancer is a vast medical problem and a major cause of mortality in the UK and worldwide. Each year, one in every 250 men and one in every 300 women get diagnosed with cancer [12]. Cancer itself includes more than 200 different diseases which are characterised by the uncontrolled proliferation of cells. The rapid and abnormal reproduction of the cells can happen in several different organs and tissues within the human body (e.g., breast, lungs, bone, etc.) [12].

* This work is supported by the DataLab and EU H2020 SU-TDS-02-2018 project Serums (No. 826278).

In this paper, we focus on chemeotherapy-based treatments for patients with breast cancer.

Chemotherapy in breast cancer is considered one of the major therapeutic treatments. Although introduced only fairly recently, it has gained increasing use both in primary management (also known as adjuvant therapy) and for patients with metastatic disease (for palliative care). Since the treatments are toxic and expensive, it is important to gain further insight into the consequences of their use when treating patients with cancer. One methodology to obtain knowledge about chemotherapy is by using a digital system (e.g., trained model or simulation). This system can evaluate the treatments applied to patients throughout several cycles.

Today, machine learning enables us to create a system which can be used to observe the outcome of the chemotherapy by feeding the data into several different learning algorithms [3]. With the right combination of data and techniques, we can improve the performance of the system and gain new insights that can guide and improve patient treatment in the future.

In this paper, we compare several different techniques, including Markov model (MM), Hidden Markov model (HMM), Random Forest (RF) and Recurrent Neural Network (RNN), to predict the outcome (e.g., toxicity) of chemotherapy treatments for breast cancer. The toxicity level is a scale obtained by measuring the condition of a patient based on several side effects of chemotherapy treatments (e.g., vomiting, diarrhoea, constipation, hand/foot and skin conditions). By comparing the result of several different techniques, we can find the connection between the treatment and its side effect. Finding this correlation among the recorded patient data can help guide clinicians and patients to decide which treatment is the most suitable for them when treating breast cancer.

This paper is structured as follows. We present related work in Section 2, describe the data and its features in Section 3, and our models in Section 4. We discuss our results in Section 5, and conclude with suggestions for further work in Section 6.

2 Related Work

In the past decade, many multivariate programs have been used to help diagnose and stage cancers, such as prostate cancer, as well as forecast the prognoses of patients [5]. As more facts about cancer are known, some cancer experts argue that every patient cancer is unique which explains why treating cancer is so difficult.

Motivated by this issue, there has been a lot of ongoing research to develop a multivariate system for personalised cancer treatment, e.g., IBM Watson [7], Microsoft Research [11], NHS [13]. Most of these approaches treat cancer as a data problem and should only be used for guidance.

Hui-Ling Chen et al. [2] used the Breast Cancer Wisconsin (Diagnostic) Data Set, which describes characteristics of the cell nuclei in an image of a fine needle aspirate (FNA) of a breast mass [1], to train a support vector machine classifier for breast cancer diagnosis. Other studies by Nguyen et al. [10] used random forest to predict breast cancer diagnosis and prognostic. By using another machine learning technique, namely Bayesian logistic regression, Subramani et al. [8] investigated the application of machine learning techniques to imaging data for predicting the eventual therapeutic response of breast cancer patients after a single cycle of neoadjuvant chemotherapy.

In our case, our data extraction consists of sequence data, and that makes it possible to explore other techniques commonly used in Natural Language Processing (NLP) such as Hidden Markov Model (HMM) [6] and Recurrent Neural Network (RNN) [4].

HMM is a sequence model for part-of-speech tagging. A sequence model, aka sequence classification-sequence model, is one whose job is to assign a label or class to each unit in a sequence, thus mapping a sequence of observations to a sequence of labels. Given a sequence of units (words, letters, morphemes, sentences, and so on), a HMM computes the probability distribution over possible sequences of labels and chooses the best label sequence [6].

RNN is an enhancement to a neural network. There is a known limitation with artificial neural networks (ANNs) and convolution neural networks (CNNs) that constrain their API. Both CNN and ANN only accept a fixed size of input or output (one sequence) [3]. RNN instead consists of several layers of ANNs, which allows us to process sequence data for which the input can be longer than one sequence [4].

In this paper, we adjust our data extraction, which is time series data, to create models using HMM and RNN and then we compare the result with the other machine learning classifiers to predict the toxicity level of a patient.

3 Data Analysis

3.1 Data Characteristics

In this paper, we use a data extraction from an oncology department in Scotland with information on treatment cycles, recorded side effects (here, toxicity level), and various observations concerning breast cancer patients for three years (from 2014 to 2016).

The extraction has data for 51661 treatments of which 13030 are of breast cancer treatments. There are 933 unique patients, and some patients have two or three different treatments/regimes during the time period. Each regime has several cycles ranging from one to more than 50 cycles (e.g., 85). Table 1 shows the number of patients for different intentions. We exclude the Curative regime because we do not have enough data for training our model.

Along with an extraction of general patient characteristics, we received the toxicity level and measurement of the patients in separate flat files. We combine the data by connecting the treatment appointment date with the date when the toxicity and other measurements (i.e., weight, height, surface area) were obtained. In this paper, we ignore patient data with no toxicity information.

Table 1: The treatment's Intentions

Intention	Total patients
Adjuvant	620
Neo-Adjuvant	427
Palliative	483
Curative	17

After we performed data cleansing, we are left with 2752 instances (i.e., 213 patients) for the palliative treatment, 1855 instances (i.e., 382 patients) for the adjuvant treatment, and 1209 instances (i.e., 205 patients) for the neo-adjuvant treatment.

3.2 Feature Analysis

Before we feed the data into the model, we analyse our datasets. First, we order the data by the cycles to make sequences. We then determine the target answer (i.e., toxicity) and predictors. After we categorised the fields, we check the relation between each predictor in the dataset to the toxicity outcome. Fig. 1 (a) shows that at the beginning of the treatment, most of the patients have low toxicity which is to be expected.

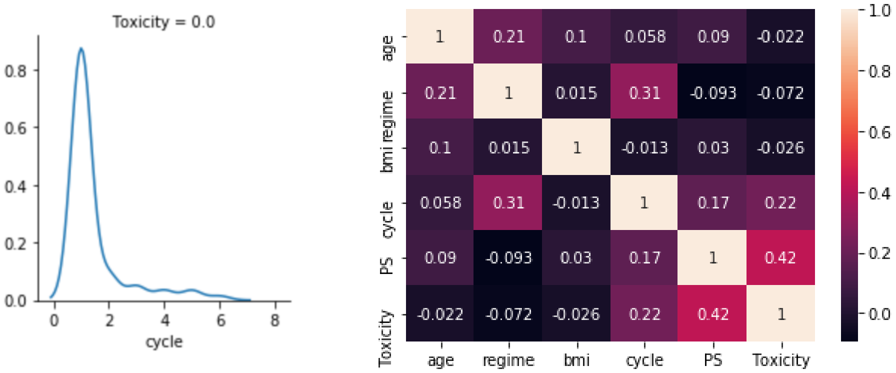

Fig. 1: Features analysis and correlation: (a) Patients' proportion against low toxicity (b) Adjuvant therapies fields' correlation map

Next, we calculate the correlation between all the predictors to the target answers as shown in Fig. 1(b). High correlation implies that there is a relationship between the variable and the target class. We want to include variables with high correlation because they are the ones with higher predictive power (signal), and leave out variables with low correlation because they are likely less relevant [9]. Even though including more relevant features during the training helps to improve the prediction power, we still include all features in the model training and then gradually exclude the irrelevant features as it is not always possible to know the features that have high predictive influence in advance.

Finally, we clean the data by replacing the missing/invalid data in our predictors. We use the mean average for fields like age or body mass index (BMI) while we use regression for the performance status (PS). To avoid the class imbalance problem, when some regime has more data than the others, we create a new dataset by duplicating some of the data. We perform this only for the RF model training because, unlike for the other models used (in our case HMM and RNN), our RF model is not dependent on the previous observation. For example, we have 141 patients treated with *FEC (D)* while only 80 patients treated with *PACLITAX*. Here, we duplicate some of the data from the *PACLITAX* to match the number of patients treated with *FEC (D)*.

4 Model Creation

As usual after analysis, we split the data into training and evaluation subset. The split ratio is 90% for training and 10% for evaluation. Hence, we randomly choose 20 patients as the test data for both adjuvant and neo-adjuvant treatments and 30 patients for the palliative treatments. All others are used to train the models.

4.1 Markov Model (MM)

A Markov model is a stochastic model with the assumption that a future state only depends on the current state [6]. Based on the toxicity in the data extractions, we created a discrete time Markov chain shown in Fig. 2 where the states represent the different levels of toxicity (e.g., T_0 corresponds to no toxicity, and T_3 is very high toxicity) and transitions reflect the treatment effects over time.

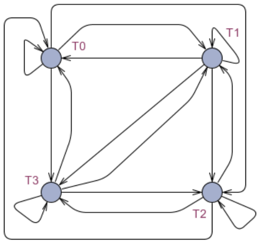

Fig. 2: The diagram representing the Markov chain for patients' toxicity outcome

Table 2 denotes the transition probability matrix for Fig. 2. From our data extraction, we calculate both the transition probability matrix and the initial probability distribution.

Table 2: The transition probability for all adjuvant treatment regimes

	T0	T1	T2	T3
T0	0.06177606	0.4980695	0.40926641	0.03088803
T1	0.03555556	0.63407407	0.30962963	0.02074074
T2	0.00524934	0.44356955	0.51968504	0.03149606
T3	0	0.32142857	0.64285714	0.03571429

We have three different Markov models for each Intention (i.e., adjuvant, neo-adjuvant, palliative). We have the model for all regimes, individual regimes, and the patient's body mass index (BMI).

4.2 Hidden Markov Model (HMM)

A HMM is based on augmenting a Markov chain to observe the hidden states of events. In our case, we want to infer/predict the toxicity level based on the patient's characteristics. Table 3 specifies the components of our HMM.

Table 3: The HMM components for predicting the toxicity outcome

Component	Description
States	The toxicity level of the patients (i.e., T0, T1, T2, T3)
Transition probabilities	The transition from one toxicity level to another toxicity level (e.g., from T0 to T1, from T1 to T3, etc).
Observations	The observed events obtained from the data extraction (i.e., cycle, age, BMI, regime). We categorise the value of each observation to simplify the process of training our HMM. For example, 1-2-3-1 denotes the observation for an overweight patient who gets the *FEC-D (D)* in their first cycle and is aged less than 50 years old.
Emission Probabilities	Each member represents the probability of the observations generated from the toxicity state.

To predict the toxicity from the sequence of the patients' events, and as is usual for HMM, we use the *Viterbi* algorithm. The *Viterbi* algorithm is a dynamic programming algorithm used for finding the most likely sequence of hidden states (aka path) [6].

Table 4: The HMM classification result example

Observed events	Toxicity Outcome
1-2-3-2/1-2-3-2/2-2-3-2	T0/T1/T1
1-2-3-4/1-2-3-4/2-2-3-4	T3/T3/T2
1-2-3-4/1-2-3-4/2-2-3-4/2-2-3-4	T3/T3/T2/T2

Table 4 shows an example of using HMM to predict the toxicity outcome for patients.

4.3 Random Forest (RF)

Random forest (RF) is an ensemble of decision trees for solving classification problems. The random forest classifier uses several features to predict the outcome [3]. For our RF model, we use the following features: *age*, *BMI*, *cycle*, *Regime*, *previous performance status*, *previous toxicity level* to predict the toxicity outcome of the treatment. We created three RF models for each treatment intention (i.e., adjuvant, neo-adjuvant, palliative), and categorised most of the features (except age) for training our model. After we created our first RF model, we manipulate the hyperparameters to get a better prediction result. Those hyperparameters are *number of estimators*, *minimum sample leafs*, *minimum sample splits*, and *the maximum depth of each tree*.

Lastly, we observe the feature importance of each field. We get an estimate of the importance of a feature by computing the average depth at which it appears across all trees in the forest [3]. The RF libraries we used for this work allowed us to compute the feature importance automatically for every feature after training. Fig. 3 shows the graph of the feature importance for the fields used to predict the toxicity outcome.

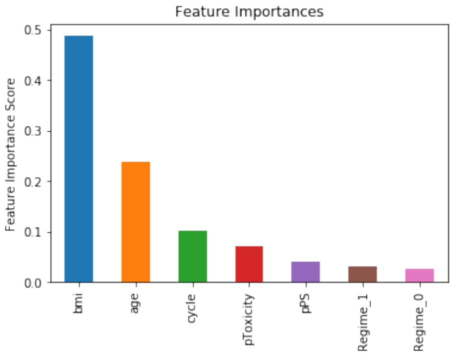

Fig. 3: The feature importance in neo-adjuvant treatments

4.4 Recurrent Neural Network (RNN)

The RNN models we created take several inputs and produce one output for each input based on the treatment cycle. During the training, we used similar features as for our RF model. However, we do not use the previous performance status and previous toxicity fields because an RNN model preserves states across time steps (in other words, has memory cells) [3]. For both models, we use the Long short-term memory (LSTM) [4] units.

5 Model Evaluation

For the MM, we observe the general pattern of the treatment outcome for each cycle and then compare it with the outcome distribution obtained from the data extraction. Fig. 4 shows both datasets plotted together. The dashed line represents the value obtained from the Markov chain. From that we get the steady-state probability after 5/6 cycles. The distribution obtained from the MM resembles the distribution obtained from the real data.

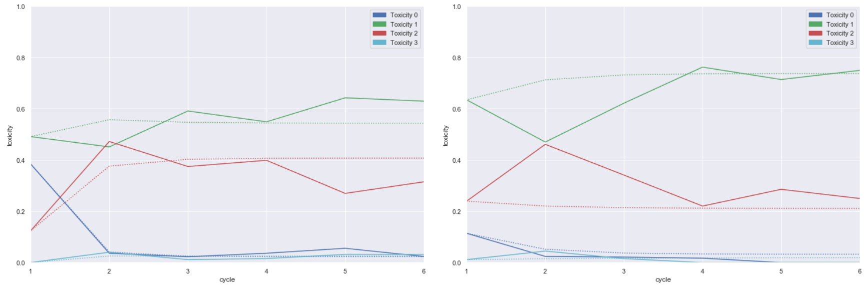

Fig. 4: The Distributions for Chemotherapy treatments: (a) Adjuvant treatments (b) Palliative treatments

We measure the performance of our classifier models by using several metrics (i.e., precision, recall, accuracy, and f1-score) after performing the cross-validation test [3]. We choose 5-fold cross validation (instead of 10-fold CV) to get more records for the validation (i.e., around 20% of total records). By duplicating the data for tackling the class imbalance issues, our model, especially the Random Forest model, is susceptible to overfitting. Here, we need more sample in the validation set to evaluate our models confidently. We do not perform the cross-validation for the HMM because the performance measured with the splitting train-test method is much lower compared to the RF or RNN models. Table 5 shows the result of the evaluation for all classifier models.

We need more data to train the corner cases (i.e., initial and end of the treatments) for the HMM models. The accuracy for the corner cases is significantly lower than the middle/transition case because the dataset has more transition cases than the initial (cycle 1) or end cases. Similarly, we can see the same characteristic for the F1-score for each treatment outcome. The T1 and T2 have higher F1-score compared to the extreme case, T0 and T3 because our datasets have more data with T1/T2 as its outcome. The RNN models outperform the RF models because, unlike RF, the RNN has LSTM units which allow the model to consider all the observations since the first treatment. Since our datasets are given as a time series, the previous treatments may affect the result of the current treatment. Hence, the RNN has an advantage compared to the RF that only considers the current state and limited information about the previous treatment result.

Table 5: Model test result (mean-std)

Model	Regime	Accuracy	Precision	Recall	F1-score
RF	Adjuvant	0.81(+/-0.11)	T0:0.55(+/-0.50)	0.92(+/-0.32)	0.65(+/-0.46)
			T1:0.85(+/-0.15)	0.83(+/-0.09)	0.84(+/- 0.09)
			T2:0.82(+/-0.09)	0.78(+/-0.20)	0.80(+/-0.13)
			T3:0.57(+/-0.52)	0.83(+/-0.67)	0.67(+/-0.55)
	Neo-Adjuvant	0.72(+/-0.09)	T0:0.53(+/-0.80)	0.60(+/-0.88)	0.52(+/-0.74)
			T1:0.77(+/-0.11)	0.80(+/-0.08)	0.79(+/-0.07)
			T2:0.63(+/-0.16)	0.61(+/-0.22)	0.62(+/-0.17)
			T3:0.33(+/-0.77)	0.23(+/-0.61)	0.23(+/-0.49)
	Palliative	0.78(+/-0.08)	T0:0.43(+/-0.23)	1.00(+/-0.00)	0.60(+/-0.24)
			T1:0.96(+/-0.03)	0.73(+/-0.09)	0.83(+/-0.06)
			T2:0.56(+/-0.17)	0.88(+/-0.10)	0.68(+/-0.15)
			T3:0.55(+/-0.81)	0.70(+/-0.92)	0.61(+/-0.83)
RNN	Adjuvant	0.85(+/-0.09)	T0:0.70(+/-0.22)	0.96(+/-0.08)	0.80(+/-0.15)
			T1:0.87(+/-0.11)	0.86(+/-0.14)	0.86(+/-0.10)
			T2:0.94(+/-0.10)	0.79(+/-0.16)	0.85(+/-0.11)
			T3:0.85(+/-0.64)	0.72(+/-0.63)	0.77(+/-0.61)
	Neo-Adjuvant	0.81(+/-0.09)	T0:0.58(+/-0.35)	0.82(+/-0.25)	0.67(+/-0.31)
			T1:0.84(+/-0.10)	0.84(+/-0.11)	0.84(+/-0.09)
			T2:0.85(+/-0.17)	0.77(+/-0.12)	0.81(+/-0.13)
			T3:0.95(+/-0.30)	0.78(+/-0.47)	0.82(+/-0.34)
	Palliative	0.85(+/-0.09)	T0:0.67(+/-0.94)	0.24(+/-0.44)	0.33(+/-0.57)
			T1:0.85(+/-0.12)	0.94(+/-0.05)	0.89(+/-0.07)
			T2:0.83(+/-0.17)	0.75(+/-0.20)	0.79(+/-0.15)
			T3:0.53(+/-0.96)	0.56(+/-0.99)	0.54(+/-0.97)
HMM (corner)	Adjuvant	0.53(+/-0.00)	NA	NA	NA
HMM (middle)		0.70(+/-0.00)	NA	NA	NA
HMM (corner)	Neo-Adjuvant	0.62(+/-0.00)	NA	NA	NA
HMM (middle)		0.70(+/-0.00)	NA	NA	NA
HMM (corner)	Palliative	0.4(+/-0.00)	NA	NA	NA
HMM (middle)		0.72(+/-0.00)	NA	NA	NA

6 Conclusion

The real value of predicting outcome/toxicity for individual patients in real-time is to help the patient and clinician understand the potential consequences of the treatment, where the patient needs to make a decision on whether to undergo treatment or not. Whereas attempts have been made to predict mortality from cancer, prediction of toxicity is much less common in the literature and where it has taken place has used simple logistic regression. The novelty of our approach is to explore the use of machine learning for these purposes.

With our classifiers, we can predict the toxicity outcome of the chemotherapy with around 0.8/0.85 accuracy. The RNN model performed better overall, because it considers all patient's treatments. Both RF and HMM only have limited observations (one previous state). However, RF has advantages because it does not differentiate between corner cases (first/last treatment) and the middle cases. Furthermore, the datasets we use for our RF models have a less class-imbalance problem than HMM. In comparison to the MM, the classifiers are more tailored for an individual patient. The MM shows the general pattern of the treatment while the classifiers can help predict the toxicity outcome of the patient. Both the MM and the classifiers complement each other.

We can improve the accuracy of our models further with more data regarding cancer characteristics or comorbidities. In our datasets, the information regarding the cancer stage is limited. We presently lack crucial information (e.g., TNM, ER/HER2 status [13]), which makes it difficult to reliably recommend suitable regimes for different patients, as we need both the toxicity outcome and cancer TNM to evaluate the treatment efficacy. For instance, some treatments might more effectively inhibit cancer growth but give higher toxicity in the short term. We are currently retraining our models with richer data extractions for more informed results on the suitability of different regimes for individual patients.

References

1. Breast cancer wisconsin (diagnostic) data set. https://data.world/health/breast-cancer-wisconsin, last accessed 15 Jan 2019
2. Chen, H., Yang, B., Liu, J., Liu, D.: A support vector machine classifier with rough set-based feature selection for breast cancer diagnosis. Expert Systems with Applications **38**, 9014–9022 (2011)
3. Geron, A.: Hands-On Machine Learning with Scikit-Learn and TensorFlow. O'Reilly Media, USA, 1 edn. (2017)
4. Graves, A.: Supervised Sequence Labelling with Recurrent Neural Networks. Studies in Computational Intelligence (2012)
5. Hu, X., Cammann, H., Meyer, H., Miller, K., Jung, K., Stephan, C.: Artificial neural networks and prostate cancer tools for diagnosis and management. Nature Reviews Urology **10**, 174–182 (2013)
6. Jurafsky, D., Martin, J.: Speech and language processing. Pearson Education, Upper Saddle River, NJ (2009)
7. Malin, J.: Envisioning Watson as a rapid-learning system for oncology. Journal of Oncology Practice **9**, 155–157 (2013)
8. Mani, S., Chen, Y., Li, X., Arlinghaus, L., Chakravarthy, A., Abramson, V., Bhave, S., Levy, M., Xu, H., Yankeelov, T.: Machine learning for predicting the response of breast cancer to neoadjuvant chemotherapy. Journal of the American Medical Informatics Association **20**, 688–695 (2013)
9. Machine learning concepts. http://docs.aws.amazon.com/machine-learning/latest/dg/machine-learning-concepts.html, last accessed 12 Oct 2017
10. Nguyen, C., Wang, Y., Nguyen, H.: Random forest classifier combined with feature selection for breast cancer diagnosis and prognostic. Journal of Biomedical Science and Engineering **6**, 551–560 (2013)
11. Project hanover. https://hanover.azurewebsites.net/, last accessed: 15 Jan 2019
12. Souhami, R., Tobias, J.: Cancer and its management. Blackwell Publishing, 5th edn. (2005)
13. Wishart, G., Azzato, E., Greenberg, D., Rashbass, J., Kearins, O., Lawrence, G., Caldas, C., Pharoah, P.: PREDICT: a new UK prognostic model that predicts survival following surgery for invasive breast cancer. Breast Cancer Research **12** (2010)