What sorts of people were more likely to survive on the Titanic?

– Team members and student IDs:
Peilin.Wu (11593587)
Chien-Min Lee (11744116)
Yi Chou (011744816)

– Abstract:

The Titanic disaster is one of the most famous shipwreck disasters in the history of the world. In 1912, the Titanic hit an iceberg and sank in the North Atlantic. There were 2,224 people on board at the time, and about 1,500 died in the shipwreck.

There are different analyzes of the survival rate of passengers in shipwreck disasters. If we can more accurately predict what factors can improve the survival rate of passengers through technology, it will significantly help society in future disaster relief. Our team wants to use machine learning to build and train our model through the passenger's surviving and dead passenger information such as sex, age, cabin class, etc.

Finally, we will use the model to predict what passenger conditions have a higher or lower chance of surviving when a shipwreck occurs. And to obtain a better model and prediction of the traffic, we will rely heavily on neural networks to train with many examples.

– Introduction

Our goal for this project is to build a predictive model that answers the question, "What sorts of people were more likely to survive?". We chose this project because Kaggle will give us comprehensive data to train, and they also have other user notebooks for us to reference. And also, the members in our group are all beginners at machine learning, so this project also fits our level.

The project will have two basic steps. First, download the data on the Kaggle and do the EDA on it. Second, we must train, tune, and ensemble some machine learning models. And finally, we will upload our prediction as a submission on Kaggle and get the accuracy score. Then we will try different machine-learning models to find the best.

– Literature review

Walter Lord conducted one of the earliest and most influential studies in his 1955 book "A Night to Remember." Lord interviewed many disaster survivors and pieced together a detailed account of what happened that night. His book helped to establish the enduring myth that women and children were prioritized for lifeboats and that crew members stayed behind to enable passengers to safety.

However, subsequent research has challenged some of these assumptions. For example, a study by John Maxtone-Graham in 1991 found that first-class passengers were more likely to survive than those in lower classes. Similarly, a study published in the journal "Demography" in 2011 found that the age and gender of passengers were not the essential factors in determining who survived. Instead, the study suggested that passengers' social networks and ability to navigate the ship significantly affected survival.

In recent years, machine learning techniques have been used to explore the factors that influenced survival on the Titanic. One of the most well-known examples is the "Titanic: Machine Learning from Disaster" competition hosted by Kaggle in 2012. In this competition, participants were given a dataset of passenger information and asked to use machine-learning algorithms to predict who survived and who did not. The competition attracted thousands of participants, and many innovative approaches were developed.

Some of the most successful techniques used in the Kaggle competition included ensemble methods, such as random forests and gradient boosting, and feature engineering techniques, such as creating new variables based on existing data. A tutorial on approaching the Titanic dataset using machine learning can be found on Kaggle's website, along with many other helpful resources for those interested in further exploring this topic.

Overall, the literature suggests that factors such as gender, age, social class, and passenger class played a significant role in determining survival on the Titanic. In addition, machine learning techniques, such as logistic regression, decision trees, and neural networks, have been successfully used to predict survival and outperform traditional statistical methods.

– Technical plan

Based on the Literature review, there are three methods that we want to try. These three methods are random forests, gradient boosting, and feature extraction.

Random forests(RF) are the ensemble of decision trees, and each decision tree will process the sample and predict the output label. And the Rf only works with tabular data, which is also why it will fit this project.

The difference between the random forest and gradient boosting is the decision trees are built additively. Each new tree is made to improve on the deficiencies of the previous trees. The gradient will give us a lot of flexibility. For example, it can optimize different loss functions.

The third method we want to try is the feature extraction method. The reason is that the feature extraction method can delete the uncorrelated or extra features. Since the data that

Kaggle provides has many aspects, ignoring the data that have little effect on the results can make the learning time faster and make our model more accurate.

– Intermediate results

In this part, our team tried to explore the dataset by analyzing the features and their relationships with the " Survived " target variable. Also, we used the data from CSV to attempt to clean the dataset by removing any missing or irrelevant values that might affect the accuracy of the prediction model.

In this process, we have learned about the dataset structure, the characteristics of the features, and their relationships with the target variable from many others' comments, such as blaming missing values and feature scaling.

And we have experimented with different machine learning algorithms and models to develop a prediction model that accurately predicts the survival of passengers in the test dataset—and evaluated the model's performance using various metrics, such as accuracy, precision, recall, and F1-score.

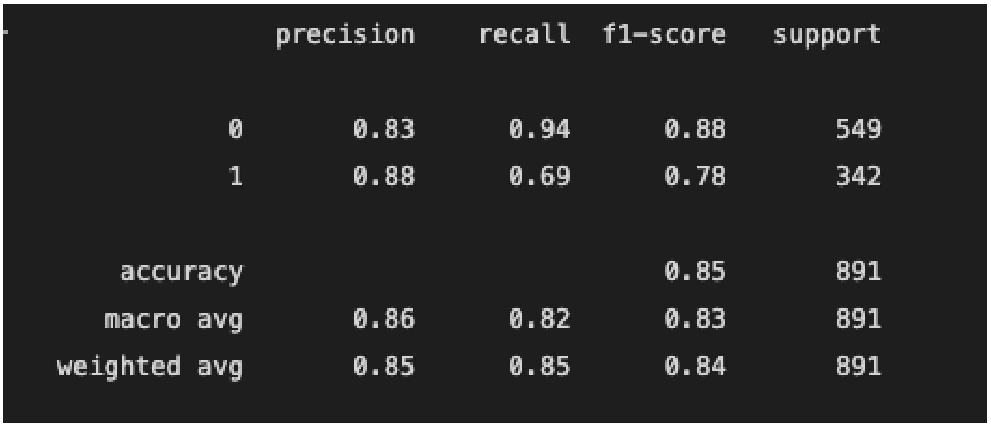
-Complete results

In our result, we used code: print(classification_report(y_train,pred)) to make the output more precise and show the difference between different Alg.

Precision, recall, and F1-score are useful metrics in machine learning to evaluate the performance of different models. Precision can evaluate the accuracy of the model. The higher the Precision, the less the wrong prediction of the model. Recall stands for all actual positive instances in the dataset, and a high recall rate means that the model has captured most of the positive instances. F1-score ranges from 0 to 1, which is the harmonic mean of precision and recall. Usually, we use F1-score as the overall evaluation of the model.

In the table, the reported averages include the macro average (averaging the unweighted mean per label), weighted average (averaging the support-weighted mean per label), and sample average (only for multilabel classification). Micro average (averaging the total true positives, false negatives, and false positives) is only shown for multi-label or multi-class with a subset of classes because it corresponds to accuracy otherwise and would be the same for all metrics.

RF:

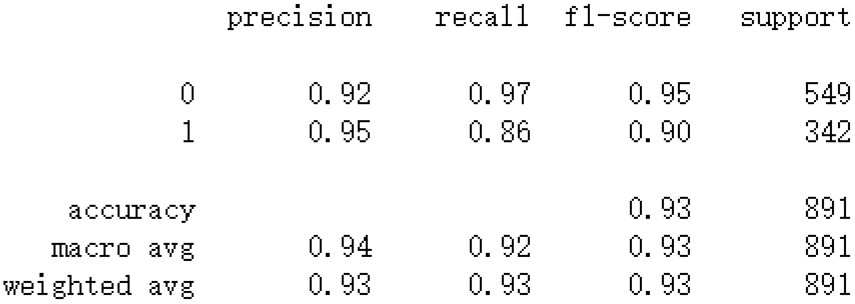


```
              precision    recall  f1-score   support

           0       0.83      0.94      0.88       549
           1       0.88      0.69      0.78       342

    accuracy                           0.85       891
   macro avg       0.86      0.82      0.83       891
weighted avg       0.85      0.85      0.84       891
```

We found that people with certain characteristics were likely to be survivors with high survival rates, such as women, children, and higher fare. Specific initial data such as sex, fare, and age are particularly likely to affect survival. While some parameters, like siblings and parents, don't contribute significantly to survival, we still need to include them in the model for training.

The Random Forest algorithm is a powerful machine learning model that is well suited for dealing with noise and missing data. Therefore it is suitable to use in this problem. However, we do note that it can be slow and computationally expensive when dealing with large datasets and many decision trees. Besides that, when there are too many trees in the Random Forest algorithm, it will still overfit.

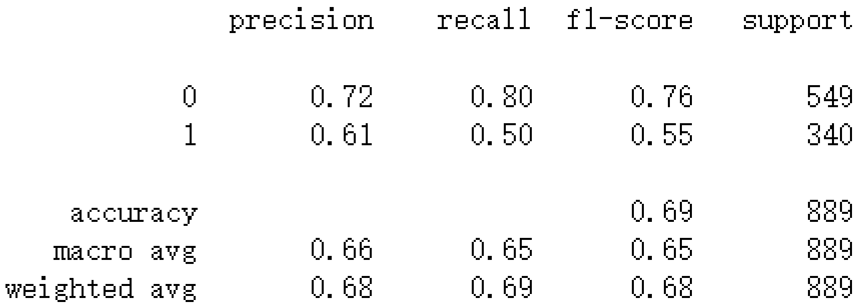
Gradient boosting:



```
              precision    recall  f1-score   support

           0       0.92      0.97      0.95       549
           1       0.95      0.86      0.90       342

    accuracy                           0.93       891
   macro avg       0.94      0.92      0.93       891
weighted avg       0.93      0.93      0.93       891
```

The advantage of gradient boosting first is that the accuracy usually is higher than others. We can see that the gradient boosting is the highest for all three methods. The disadvantage is that Gradient Boosting Models will continue improving to minimize all errors. This can overemphasize outliers and cause overfitting.

Cnn:



```
              precision    recall  f1-score   support

           0       0.72      0.80      0.76       549
           1       0.61      0.50      0.55       340

    accuracy                           0.69       889
   macro avg       0.66      0.65      0.65       889
weighted avg       0.68      0.69      0.68       889
```

The purpose of CNN is to extract features of things with a specific model and then classify, identify, predict, or make decisions based on the parts. The most crucial step is feature extraction, extracting the features that can distinguish things to the greatest extent. To realize this great model, CNN needs to be iteratively trained.

After finishing the project, I think the advantage of CNN is that it shares the convolution kernel, optimizes the amount of calculation, and does not need to manually select features, train the weights, and get the parts. Moreover, the deep network extracts images with rich information and good expression effects.

However, while parameter adjustment is required, a large sample size is also needed, resulting in a longer demand time.

Rf vs GB

Gradient boosting trees can be more accurate than random forests. Because we train them to correct each other’s errors, they can capture complex patterns in the data. However, if the data are noisy, the boosted trees may overfit and start modeling the noise.

Two main differences exist between the gradient-boosting trees and the random forests. First, we train the former sequentially, one tree at a time, each to correct the errors of the previous ones. In contrast, we construct the trees in a random forest independently. Because of this, we can train a forest in parallel but not the gradient-boosting trees.

The other principal difference is in how they output decisions. Since the trees in a random forest are independent, they can determine their outputs in any order. Then, we aggregate the individual predictions into a collective one: the majority class in classification problems or the average value in regression. On the other hand, the gradient-boosting trees run in a fixed order, and that sequence cannot change. For that reason, they admit only sequential evaluation.

Gb vs Cnn

These are two machine learning algorithms used for different tasks and data types, with varying structures of the model, feature extraction methods, and training methods.

Gradient boosted trees are an ensemble learning algorithm that belongs to supervised learning and is used to solve classification and regression problems. It works by gradually training a series of weak learners and combining them into a strong learner. CNN is a deep learning algorithm that belongs to unsupervised learning and is mainly used for image recognition and processing tasks.

Cnn vs Rf

Random Forest is less computationally expensive and requires no GPU to finish training. A random forest can give you a different interpretation of a decision tree but with better performance. Neural Networks will need much more data than an everyday person might have to be effective. The neural network will simply decimate the interpretability of your features to the point where it becomes meaningless for the sake of performance.

– Future work after this course

Now that we have completed three computing models, we will do the following things to improve our project in the next task.

Model improvement: According to the experimental results and evaluation indicators in the project, you can consider further optimizing the used algorithms or try other more advanced machine learning algorithms, such as deep learning models, to improve the performance and prediction accuracy of the models. For example, one can improve predictive models using ensemble learning methods such as Random Forest, Gradient Boosted Trees, etc.

Feature engineering: In the Titanic survivor data, there may be other unutilized features, such as passenger family member information, cabin class, ticket price, etc. These features can be further explored and processed better to reflect passengers' personalized features and background information and be added as input features to the predictive model.

Data expansion: Consider collecting more data on Titanic survivors or other similar ship disaster events to expand the scale and diversity of the data set, thereby improving the generalization performance and reliability of the model.

Explain model predictions: For the prediction results of the model, the reasons and explanations behind them can be further explored. For example, explanatory machine learning techniques, such as LIME, SHAP, etc., can explain the model's prediction decisions, thereby increasing the interpretability and credibility of the model's predictions.

Model application: You can consider applying the trained model to actual scenarios, such as real-time passenger survival prediction, ship safety assessment, etc., to verify the effect and usability of the model in real applications.

Sub-problem research: Some interesting sub-problems can be excavated from the problems involved in the project for in-depth analysis. For example, the impact of different passenger characteristics on the survival rate can be explored, such as gender, age, cabin class, etc.; or the interaction and association between other traits can be studied, such as the relationship between family members and survival rate, etc.

– Reference:

Farag, & Hassan, G. (2018). Predicting the Survivors of the Titanic Kaggle, Machine Learning From Disaster. Proceedings of the 7th International Conference on Software and Information Engineering, 32–37. https://doi.org/10.1145/3220267.3220282

Shetty, Pallavi, S., & Ramyashree. (2018). Predicting the Survival Rate of Titanic Disaster Using Machine Learning Approaches. 2018 4th International Conference for Convergence in Technology (I2CT), 1–5. https://doi.org/10.1109/I2CT42659.2018.9058280

The University. (1978). What is a neural network? Amazon. Retrieved February 18, 2023, from https://aws.amazon.com/what-is/neural-network/#:~:text=A%20neural%20network%20is%20a,that%20resembles%20the%20human%20brain.

Singh, Saraswat, S., & Faujdar, N. (2017). Analyzing the Titanic disaster using machine learning algorithms. 2017 International Conference on Computing, Communication, and Automation (ICCCA), 406–411. https://doi.org/10.1109/CCAA.2017.8229835

Singh, Nagpal, R., & Sehgal, R. (2020). Exploratory Data Analysis and Machine Learning on Titanic Disaster Dataset. 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 320–326. https://doi.org/10.1109/Confluence47617.2020.9057955

What is logistic regression? IBM. (n.d.). Retrieved February 18, 2023, from https://www.ibm.com/topics/logistic-regression#:~:text=Resources-,What%20is%20logistic%20regression%3F,given%20dataset%20of%20independent%20variables.

Rokach, L., &amp; Maimon, O. (2015). Data mining with decision trees: Theory and applications. World Scientific.

Géron Aurélien. (2023). Hands-on machine learning with sci-kit-learn, Keras, and TensorFlow: Concepts, tools, and techniques to build Intelligent Systems. O'Reilly.

Lie&, szlig, Mareike, Glaser, B., & Huwe, B. (2012). Uncertainty in the spatial prediction of soil texture: Comparison of regression tree and Random Forest models. Geoderma, 170, 70–79. https://doi.org/10.1016/j.geoderma.2011.10.010

Mecikalski, Sandmael, T. N., Murillo, E. M., Homeyer, C. R., Bedka, K. M., Apke, J. M., & Jewett, C. P. (2021). A Random-Forest Model to Assess Predictor Importance and Nowcast Severe Storms Using High-Resolution Radar-GOES Satellite-Lightning Observations. Monthly Weather Review, 149(6), 1725–1746. https://doi.org/10.1175/MWR-D-19-0274.1

https://github.com/ValentinFigue/Sklearn_PyTorch