Capstone 1: Liver Disease Prediction

Final Report

Nils Madsen

Introduction

The liver performs many critical life functions in the context of digestion and detoxification of the blood. ‘Liver disease’ is a broad term which encompasses the many forms of liver malfunction that lead to illness. Liver disease needs to be diagnosed early and treated to prevent permanent liver damage (cirrhosis), which can lead to death or require drastic and risky medical intervention in the form of liver transplant.

The process of diagnosing liver disease begins with the identification of abnormal results on one or more blood tests of liver function, but blood tests alone are not considered sufficient to conclusively diagnose or rule out liver disease. The current gold standard for diagnosis of liver disease is liver biopsy, with inspection of the biopsied tissue under a microscope by a pathologist. However, liver biopsy is an invasive procedure which is not without risk, and can lead to substantial cost to the healthcare system.

Goal and Utility

The goal of this project is to predict the presence or absence of liver disease from a set of blood tests and other diagnostic measures. These measurements are much less invasive and costly than a liver biopsy, since at most they require blood to be drawn for a blood test.

The prediction of liver disease from these metrics can help physicians in multiple ways during the diagnosis process. If the algorithm can give a confident positive determination (that a patient has liver disease), it may allow doctors to reach a faster diagnosis, and allow patients to avoid unnecessary testing for other diseases. If the algorithm is able to make a confident negative determination (that a patient does not have liver disease), doctors can avoid putting patients through costly and invasive liver biopsies. In any case, this algorithm could help reduce some of the uncertainty doctors face with ambiguous or mixed test results, by learning from a much larger number of cases than could ever be seen by a single physician.

This algorithm may also be useful to insurance companies during the prior-authorization process for invasive medical tests. This could help to protect patients from unnecessary risk, and lower the burden of disease diagnosis on the healthcare system.

Data

The data set for this project consists of 583 observations of patients near Andhra Pradesh, India. The observations include 416 liver disease patients and 167 controls. The data set has 10 diagnostic metrics: age, sex, total bilirubin, direct bilirubin, alkaline phosphatase (ALP), alanine aminotransferase (ALT), aspartate aminotransferase (AST), total protein, albumin, and albumin-globulin ratio (a-g ratio). The final column is the group membership of the patient (i.e. liver patient or control).

The data set was discovered on Kaggle, and gathered from the UCI Machine Learning Repository. This page contains information about the data set, as well as the academic source and relevant scientific publications:

https://archive.ics.uci.edu/ml/datasets/ILPD+(Indian+Liver+Patient+Dataset)#

This approach is not restricted simply to liver disease, however. Any disease that has blood tests or other metrics used in its diagnosis could be approached in this way.

Data Cleaning

Missing Values

There are 4 observations that have incomplete data; specifically, these patients are missing data for a-g ratio. Since there are only 4 patients with incomplete information, dropping these observations entirely would be unlikely to bias the dataset in a meaningful way. However, since only one field is missing, dropping entire rows seems like a waste of data. Therefore, I elected to fill the missing values by imputing from the mean of the population. This should give the model more information to learn from, while not affecting the population distribution of a-g ratio.

Duplicate Rows

There are 13 duplicate rows in the data set. Unfortunately, there is no unique identifier for each patient, so there is no way to determine whether these rows are actually duplicated, or if they refer to multiple patients who have the same measures across all the variables. Because of this, no duplicate rows have been removed.

Outliers

An inspection of the variable distributions (see Appendix) reveals a significant amount of outliers present in total bilirubin, direct bilirubin, ALP, AST, and ALT. While this might be alarming in other types of data, the presence of outliers in these measures is not surprising. In fact, it is expected.

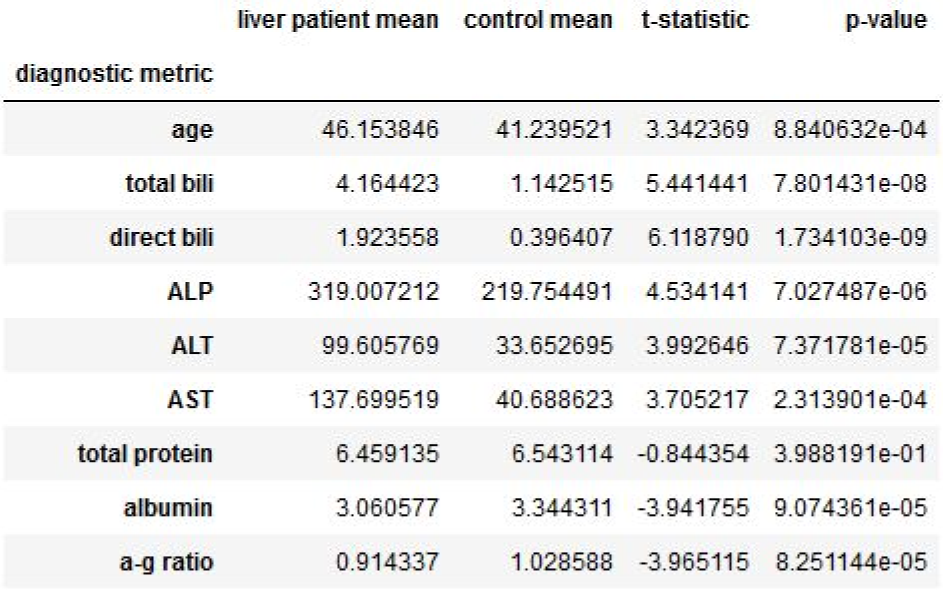
The molecules being measured by these tests are normally kept at low concentrations in the blood. When the liver becomes damaged, these measures can rise several fold, because there is a failure of the body's ability to keep them low. In addition, the scale of the increase in certain markers is used by doctors to predict what kind of liver disease is present. For example, an ALT result 2-3x higher than the normal maximum might indicate alcoholic or fatty liver, while a measure 20x higher is more specific to viral hepatitis or poisoning.

A small number of outliers are present in total protein, albumin, and a-g ratio. It is not clear what justification could be made for removing them; there is no way to judge whether or not the measurements are legitimate or errors. Also, the outliers are not dramatically larger or smaller in these fields, as are the outliers for bilirubin and the liver enzymes.

For these reasons, I have chosen not to remove any outliers from the data set.

Data Exploration

The diagnostic measures appear to have independent predictive value
An investigation of the covariance structure of the predictive measures in the data set indicates that most of the measures are not highly correlated to each other (Figure 1), but almost all are related to liver disease diagnosis in a statistically-significant way (Table 1).



	liver patient mean	control mean	t-statistic	p-value
diagnostic metric				
age	46.153846	41.239521	3.342369	8.840632e-04
total bili	4.164423	1.142515	5.441441	7.801431e-08
direct bili	1.923558	0.396407	6.118790	1.734103e-09
ALP	319.007212	219.754491	4.534141	7.027487e-06
ALT	99.605769	33.652695	3.992646	7.371781e-05
AST	137.699519	40.688623	3.705217	2.313901e-04
total protein	6.459135	6.543114	-0.844354	3.988191e-01
albumin	3.060577	3.344311	-3.941755	9.074361e-05
a-g ratio	0.914337	1.028588	-3.965115	8.251144e-05

Table 1: T-test comparisons of the numeric diagnostic metrics between controls and liver patients

Age, total and direct bilirubin, ALP, ALT, and AST are all higher in liver patients than in controls. Albumin and a-g ratio are lower in liver patients. Only total protein is not significantly different between controls and liver patients, indicating that this measure may have little utility in training the learning model.

Total and direct bilirubin are highly correlated, which is not surprising given direct bilirubin makes up a portion of total bilirubin. Similarly, albumin and a-g ratio are positively correlated, since albumin is the numerator in the albumin-globulin ratio.

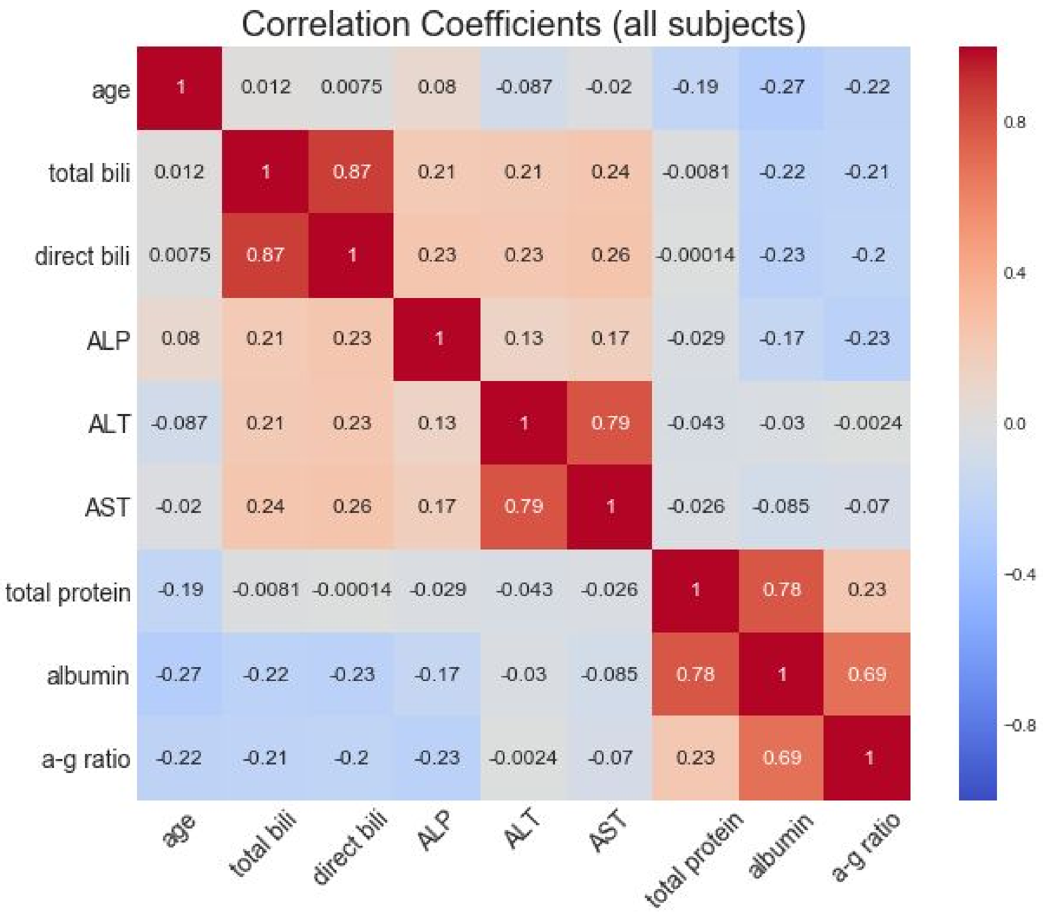


Figure 1: Correlations among the diagnostic measures (all subjects)

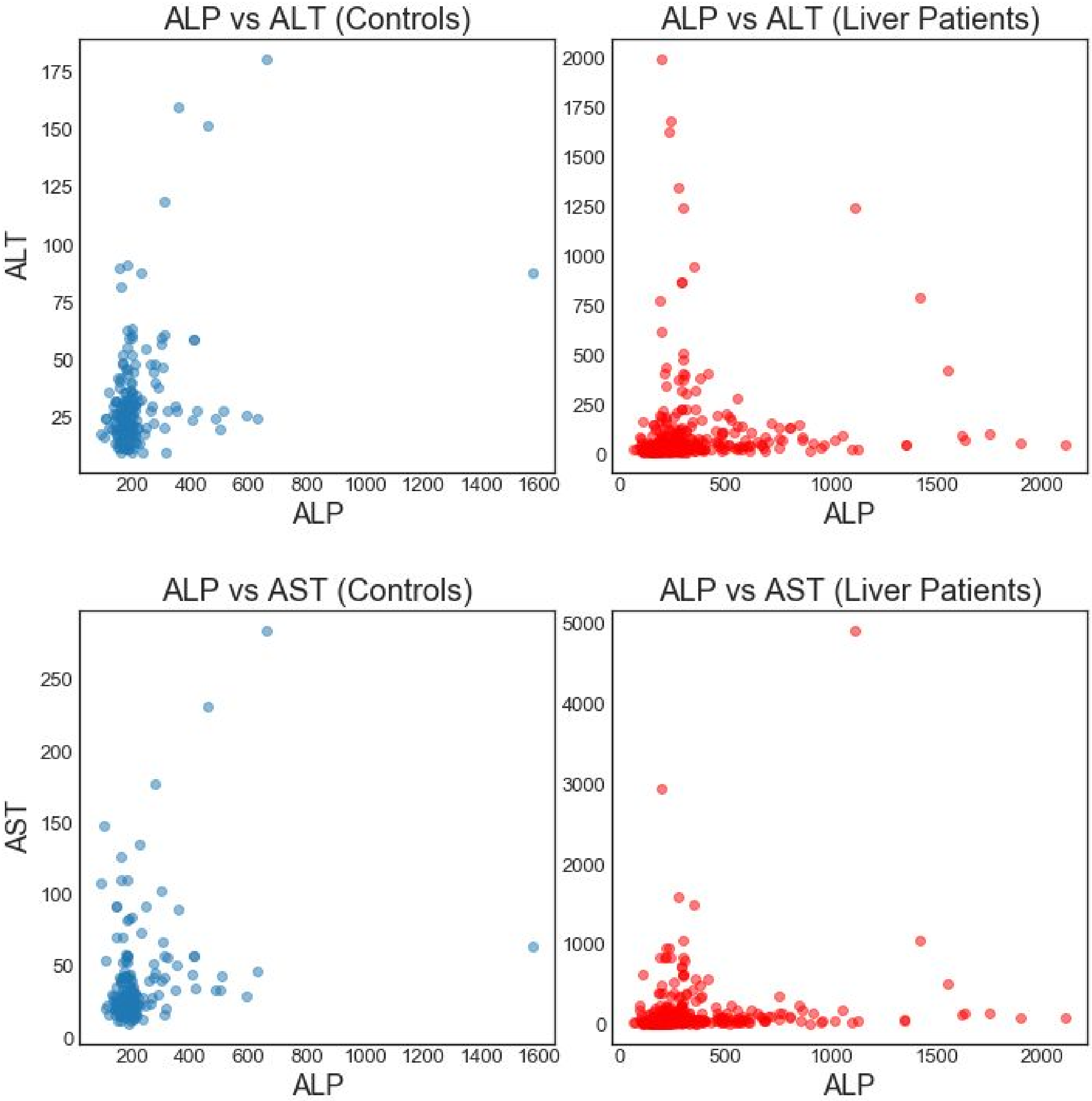


Figure 2: Distribution of ALP vs ALT and AST

AST and ALT are also highly correlated, indicating that they have overlapping (somewhat redundant) value in diagnosing liver disease.

ALP has only weak correlation with the other two liver enzymes (AST and ALT). Furthermore, very high levels of ALP in the blood are not often seen with very high levels of ALT or AST in liver patients (Figure 2), suggesting that ALP and ALT/AST may be elevated in different forms of liver disease.

From the above examination, there appear to be 5 meaningful clusters of correlated measures in the dataset: age, the bilirubin measures (total and direct), ALP, the transaminases (ALT and AST), and the albumin measures (albumin and a-g ratio). However, there are only weak correlations between these clusters, likely indicating a good amount of independent predictive power in this set of measures. This is a good thing, and should allow the machine learning algorithm to predict liver disease better than any of the individual measures in isolation.

The blood tests have low sensitivity for liver disease

One notable observation from close inspection of the diagnostic measure distributions (appendix 1) is that many liver disease patients have blood tests that are normal or close to normal. This indicates that these blood tests have relatively low sensitivity for liver disease; a normal result for these blood tests cannot rule out a liver disease diagnosis.

On the other hand, it appears that some of the blood tests have good specificity, or in other words a blood test result above a certain value virtually guarantees the presence of liver disease. These two conclusions, taken together, suggest that a machine learning algorithm trained on this data will be capable of high specificity but not high sensitivity.

The high specificity and low sensitivity somewhat reduce the value of this machine learning model; if the model cannot rule out liver disease, then it cannot help patients and doctors avoid unnecessary liver biopsies. However, a model with high specificity and low sensitivity still has value, because its use as a 'rule-in' test for liver disease could save some patients from undergoing unnecessary tests for other diseases in their doctor's differential diagnosis.

Sex is significantly associated with liver disease in this dataset

A proportions z-test indicates that there is a significantly greater proportion of men among the liver patients than among the controls (0.78 vs. 0.70; $p < 0.05$). This may suggest that male sex is associated with a higher risk of liver disease. However, this could also be a result of sampling bias, if for some reason male liver patients were easier to recruit than female liver patients, or female controls were easier to recruit than male controls. Large and carefully-collected data sets are needed to draw strong conclusions on this kind of question.

Limitations of the dataset

Before exploring how learning models perform, it is important to recognize the limitations of this dataset. This dataset is not optimal for the purpose of machine learning, for multiple reasons. The first and most important factor is the small size of the dataset. With fewer than 600 observations across both classes, stochastic effects become quite notable, and the performance of the machine learning models varies with the specific test-train split.

The dataset also has a substantial degree of class imbalance, with most observations coming from the group of patients with liver disease. This is in stark contrast to the reality in the larger population, where most people do not have liver disease. Class imbalance can cause learning models to overpredict the class that has disproportionately high representation in the data. Class imbalance also makes tuning based on accuracy a poor choice, since accuracy may increase simply by predicting the over-represented class more often.

Generally, the best remedies to class imbalance are sampling methods, such as taking a subsample of the over-represented class in order to create a class balance that is more relevant for the anticipated usage of the model. So in the case of liver disease prediction, it would be best to sample the liver disease patients to create a dataset where the proportion of liver disease patients matches the proportion of people who end up having liver disease after having liver-related blood tests taken. However, using this approach here would lead to a dataset that is far too small to be useful.

Finally, the features themselves are not powerful predictors of liver disease. As discussed previously, the blood tests used here appear to have poor sensitivity for liver disease detection, and many liver disease patients have normal or near-normal blood test results. If it ends up being the case that the clusters of controls and liver disease patients do not adequately separate, the ability of any learning model to predict liver disease based on these features will be limited.

Machine Learning Modeling

Overview of Rationale and Approach

The predictors in this problem are continuous, quantitative variables (with the exception of sex). As discussed previously, there is some degree of covariance among the features in this dataset. In order to mitigate the effect of this covariance on the success of the learning models (especially logistic regression), it would be best to use a set of uncorrelated predictor variables.

Principal Component Analysis (PCA) is a common and effective tool for producing derived features (components) that are uncorrelated. PCA also has the benefit of allowing for dimensionality reduction while keeping most of the information in the dataset, which may improve the performance of the learning models.

However, correct implementation of PCA requires that the data be unskewed, centered, and equally scaled between variables. The overall strategy here will be:

1. Split the data into training and test datasets
2. Deskew the quantitative variables using various transformations
3. Standardize the features
4. Use PCA to produce uncorrelated features, possibly with dimensionality reduction
5. Train various learning models, with cross-validated hyperparameter tuning
6. Get an idea for the performance of the models on new data by applying the trained models to the test dataset

Splitting into training and test datasets

Prior to any transformation on the data, the dataset was split into training and test sets. All decisions regarding which transformations to use for deskewing data, how many principal components to use, and what hyperparameters to use for each model, were made using the training data only. This is to best simulate the use of these models for new, unseen data.

Deskewing and Standardization

Sex was removed from the dataset before standardization and PCA, since it is a categorical variable. It was reintroduced into the dataset after PCA/dimensionality reduction so that it could be used as a feature by the learning models

Age and albumin are unskewed, so no transformation was applied to these features. A-g ratio and total protein are mildly skewed, and the remainder of the quantitative variables are heavily skewed. For each skewed variable, multiple deskewing transformations were investigated to see which returned the best result. Total bilirubin was best treated with the reciprocal transformation, while the rest of the skewed features were best treated with the Box-Cox transformation. For features transformed by Box-Cox, the test data were transformed with the lambda values determined to be the best in the training data.

After deskewing, each feature was standardized to a mean of 0 and standard deviation of 1, using the StandardScaler class within sci-kit learn. The test data was transformed using the mean and standard deviation of the training data.

Principal Component Analysis

PCA was applied to the deskewed, standardized training data. Two distinct elbow curves appear in the explained variance plot, one at 4 components and the other at 7-8 components. The first 4 components account for 82% of the variance in the data, while the first 7 components explain 99% of the variance. Since there were 9 quantitative variables to start with, these results suggest that there is at least 2 variables' worth of redundancy in the data due to covariance.

The test data were transformed using the components determined by the training data.

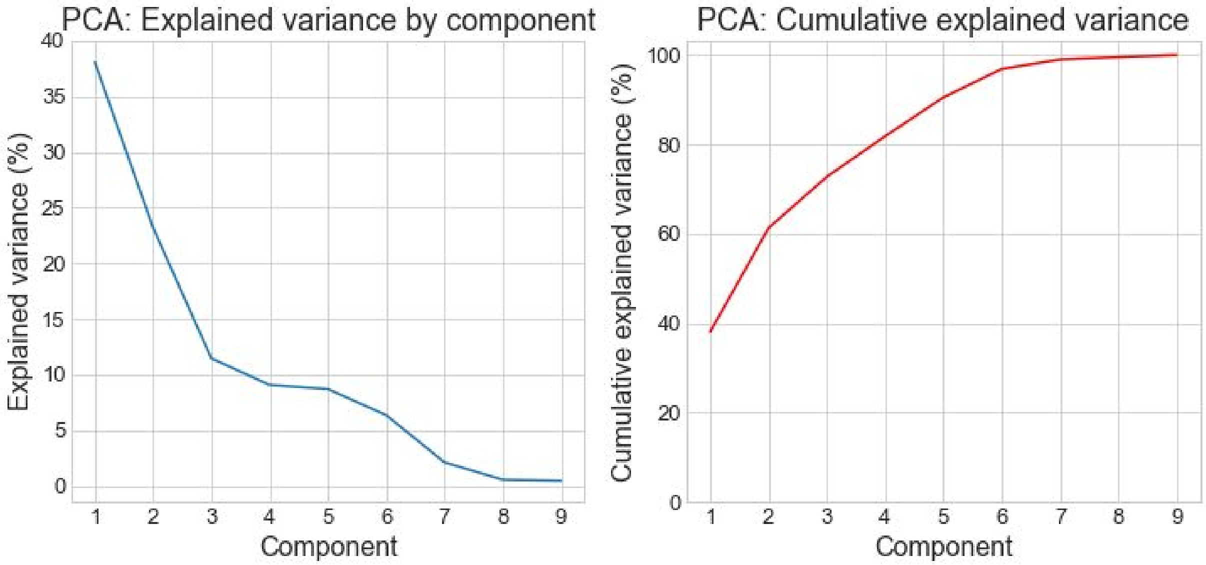
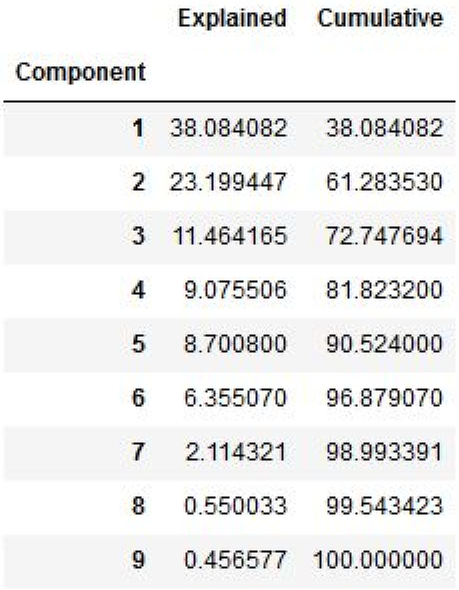


Figure 3: PCA explained variance plots



	Explained	Cumulative
Component		
1	38.084082	38.084082
2	23.199447	61.283530
3	11.464165	72.747694
4	9.075506	81.823200
5	8.700800	90.524000
6	6.355070	96.879070
7	2.114321	98.993391
8	0.550033	99.543423
9	0.456577	100.000000

Table 2: PCA explained variance (percent)

Choice of performance metric

As discussed previously, accuracy is a poor choice of performance metric for a dataset with class imbalance. Since the features have poor sensitivity but decent specificity, precision was chosen as the main evaluation metric. This matches an intuitive understanding of how the model would be used - if the model would primarily be used as a rule-in test for liver disease, then a doctor would want to be as confident as possible that a patient who is predicted to have liver disease actually has liver disease.

Logistic Regression

An evaluation of the performance of logistic regression for various number of components revealed that the best performance occurred when 8 components were used. Precision and accuracy improved with more components, while area under the ROC curve (AUC) stayed relatively the same above 5 components.

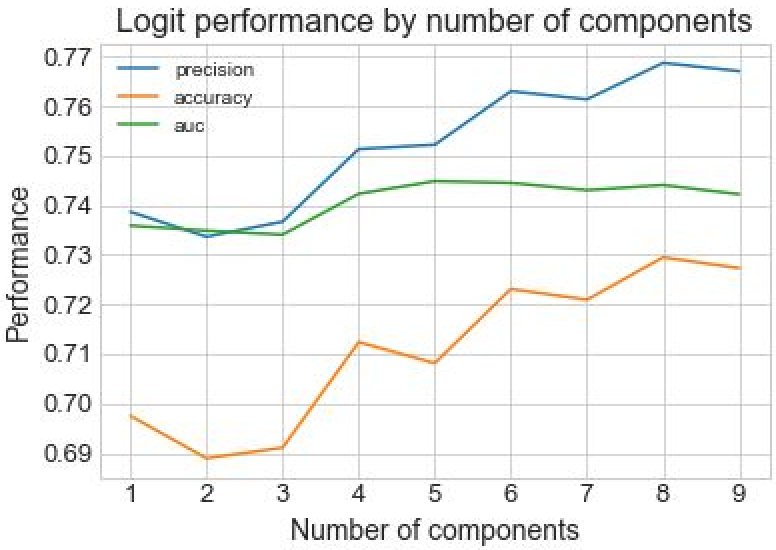


Figure 4: Performance of the logistic regression model with various numbers of PCA components

Tuning of the C hyperparameter using 5-fold cross-validation showed that a value less than 10^{-3} led to the highest precision. This suggests that logistic regression performed best when strong regularization was used to prevent overfitting.

Evaluating the performance of the logit model on the test data shows precision very close to the values achieved in cross-validation (0.807 vs 0.809), indicating that the model succeeded in creating generalizable predictions and did not overfit. An inspection of the confusion matrix shows a decent error profile, with more controls being predicted negative than positive, and many more liver patients being predicted positive than negative.

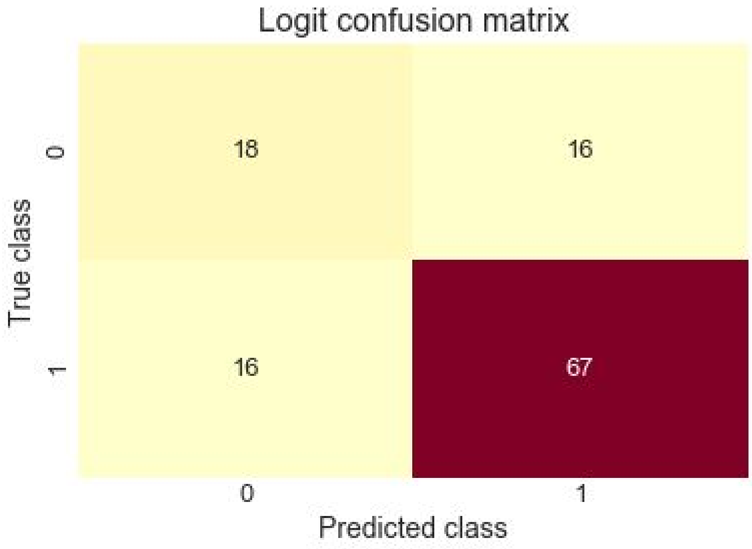


Figure 5: Confusion matrix of the logistic regression model on the test data

Random Forest

Random forest is more resistant to skew, scaling, and excess features, so it was trained on both the original and the transformed data to see which led to better performance. After hyperparameter tuning, the random forest trained on the transformed data had a slightly higher cross-validated precision than the model trained on the original data (0.778 vs 0.771). Similarly to logistic regression, random forest performed best using 8 components of the transformed data.

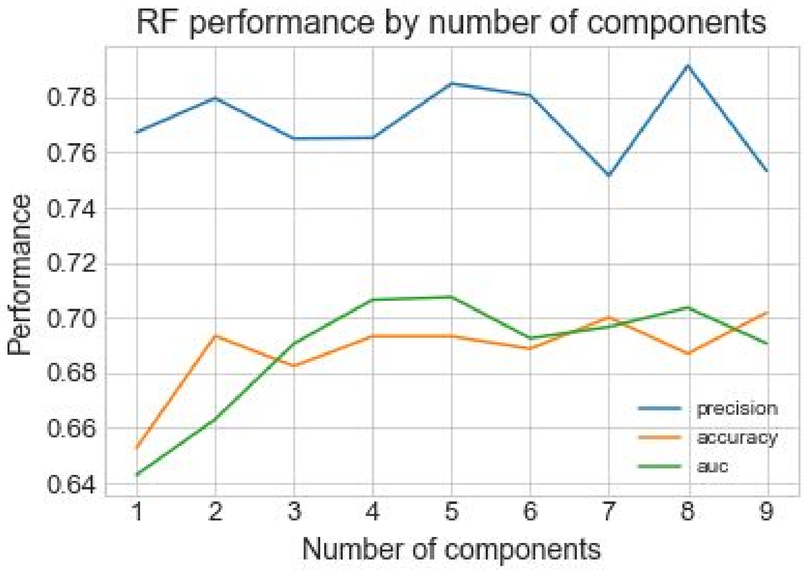


Figure 6: Performance of the random forest model with various numbers of PCA components

The random forest model trained on the transformed data performed as well as logistic regression with regard to accuracy (0.726) but had lower precision (0.787). Furthermore, an inspection of the confusion matrix reveals it is classifying most subjects as having liver disease, even those within the control group. This is not as useful a model as logistic regression.

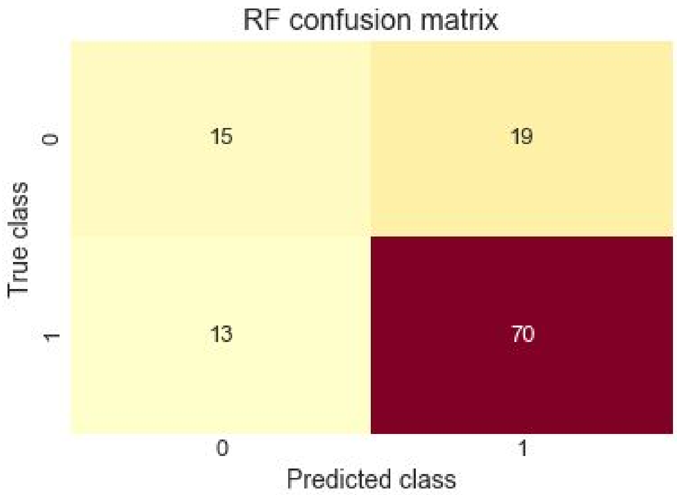


Figure 7: Confusion matrix of the random forest model on the test data

Support Vector Machine

SVM is sensitive to skew, scaling, and excess features, so the transformed data was used for training. SVM performed best using only the first three PCA components.

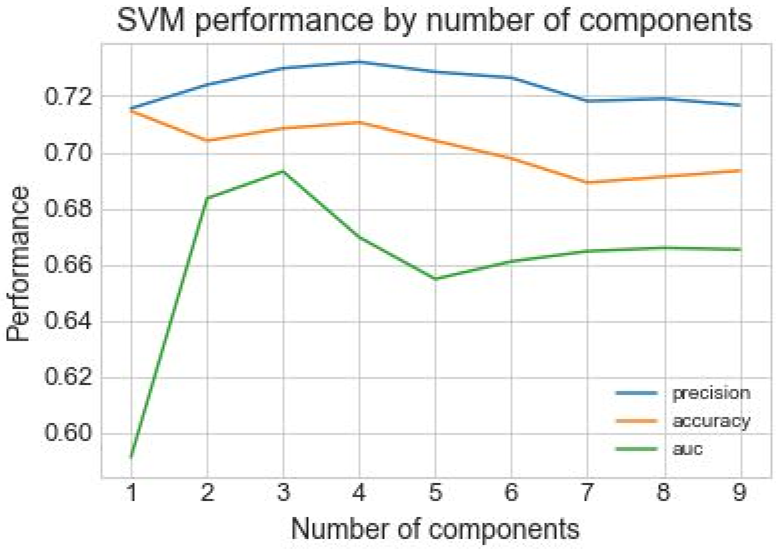


Figure 8: Performance of the support vector machine model with various numbers of PCA components

SVM exhibited worse performance on the test dataset than the other two models, having both lower accuracy (0.667) and lower precision (0.762).

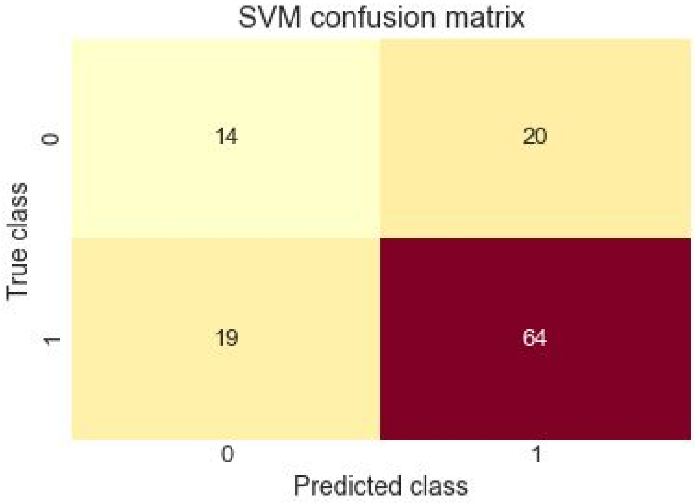


Figure 9: Confusion matrix of the support vector machine model on the test data

Ensemble Classifiers

Determining predicted class labels by combining the predictions of multiple classification algorithms can sometimes produce better performance than any of the individual algorithms. Two different approaches were explored here. The first was to predict the disease state of a subject by taking a majority vote among logistic regression, random forest, and SVM. Since SVM performed worse than the other two models, prediction of disease state was also attempted based on averaging the predicted probabilities of logistic regression and random forest.

The majority vote classifier exhibited an accuracy that was superior to both logistic regression and random forest (0.735) and a precision only slightly lower than that of logistic regression (0.802). The averaging classifier did not fare quite as well, with lower accuracy (0.726) and lower precision (0.780).

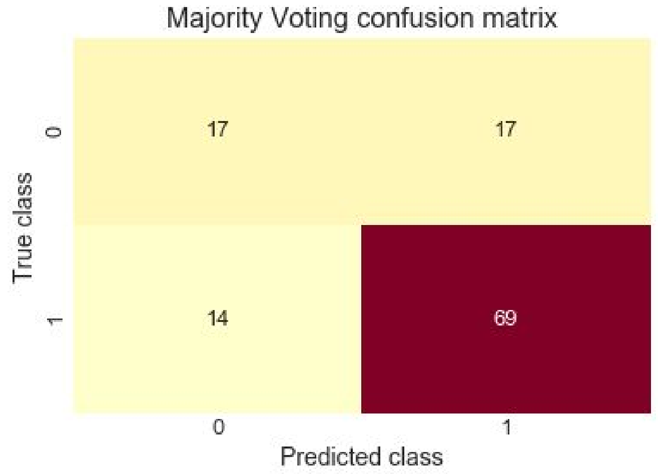


Figure 10: Confusion matrix of the majority vote among logistic regression, random forest, and SVM on the test data

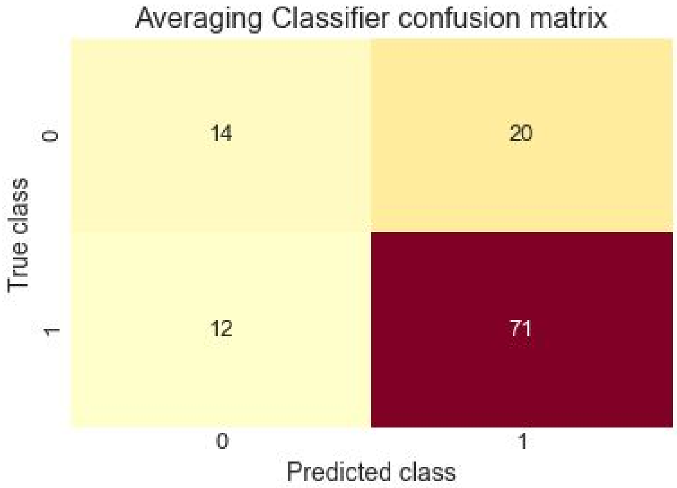
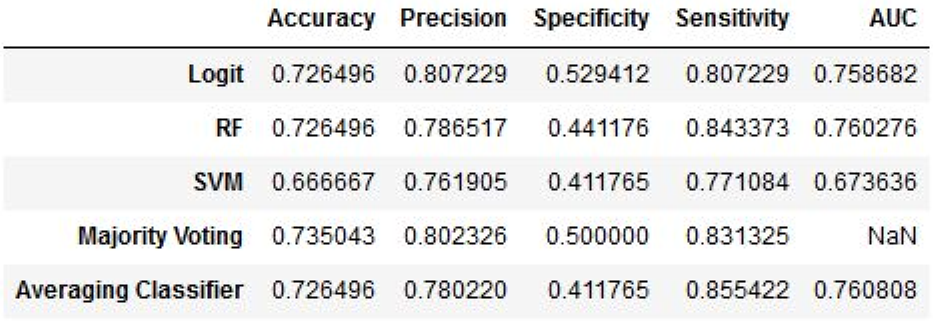


Figure 11: Confusion matrix of the average predicted probability of the logistic regression and random forest models on the test data

Comparison of models

SVM performed the worst out of all the models, having no performance advantage over logistic regression or random forest. Logistic regression performed the best of the individual classification algorithms, achieving the same accuracy as random forest but with higher precision and a better error profile.

The choice between the logit model and the majority vote classifier is a harder one, as logistic regression has a slight precision and specificity advantage while the majority voting classifier has a slight accuracy and sensitivity advantage. The difference between these models in real-world use would likely be small, so both would be good choices. Logistic regression may be favored for practical reasons, however, due to its relative simplicity.



	Accuracy	Precision	Specificity	Sensitivity	AUC
Logit	0.726496	0.807229	0.529412	0.807229	0.758682
RF	0.726496	0.786517	0.441176	0.843373	0.760276
SVM	0.666667	0.761905	0.411765	0.771084	0.673636
Majority Voting	0.735043	0.802326	0.500000	0.831325	NaN
Averaging Classifier	0.726496	0.780220	0.411765	0.855422	0.760808

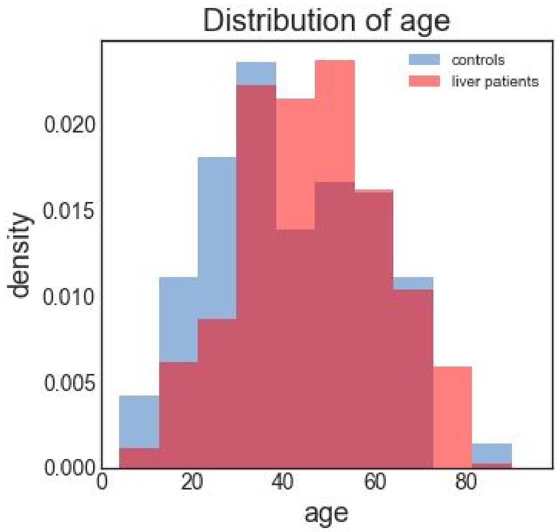
Table 3: Performance metrics for all models

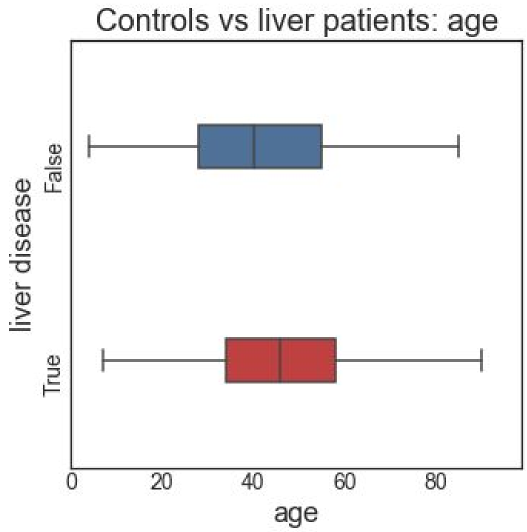
Future Directions

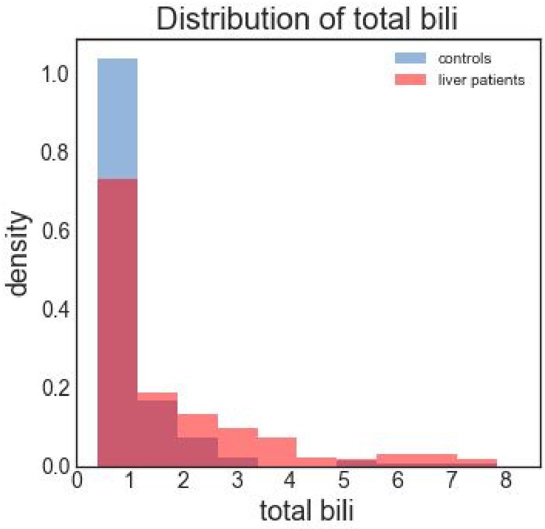
One additional analysis that could yield interesting results for this dataset is clustering. Clustering analysis could help illustrate to what degree the classes overlap, and what proportion of liver patients have blood tests indistinguishable from those of the control group. These results could inform why learning models do not perform particularly well on this data. Clustering could also be performed within the liver patient group, and may reveal subclasses of liver patients reflecting different types of liver disease.

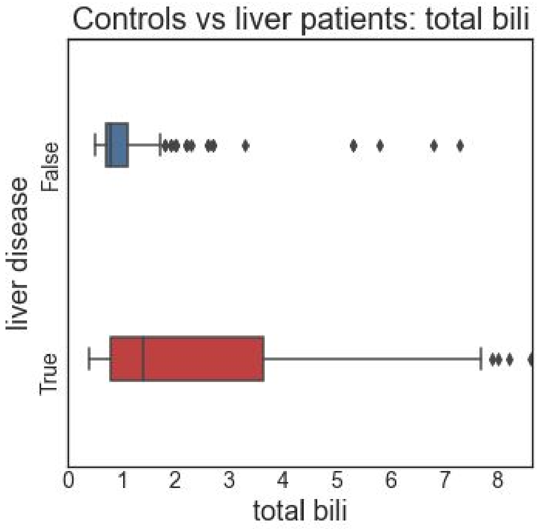
This problem may also benefit from changing the probability at which the models make a liver disease prediction, in order to further enhance the precision of the logistic regression, random forest, and average predicted probability model. However, evaluating the performance of different threshold probabilities in a robust manner generally requires two test sets; the first is used after hyperparameter tuning to determine which threshold is best, and the second is used to evaluate the performance of that threshold. This dataset is simply too small to engage in this kind of analysis.

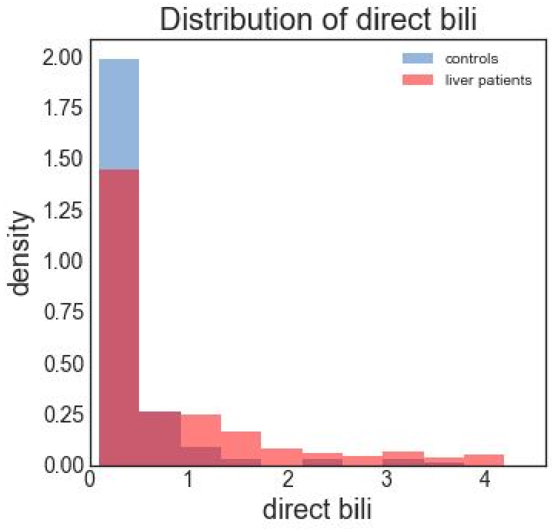
Appendix: Distributions of the diagnostic measures

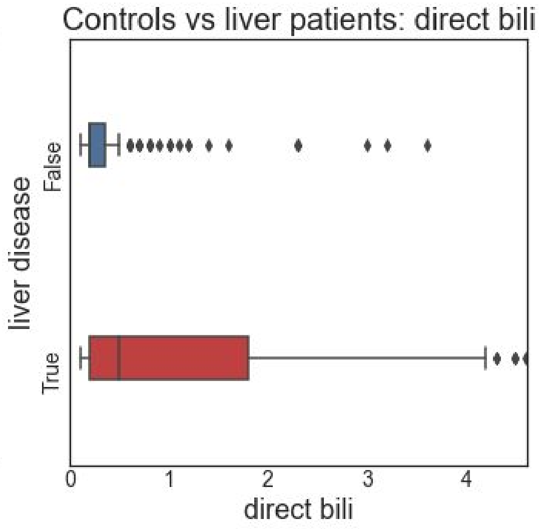

Distribution of age

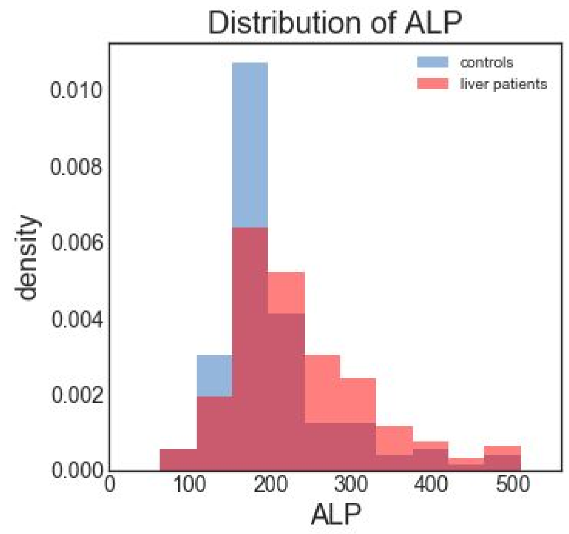

Controls vs liver patients: age


Distribution of total bili

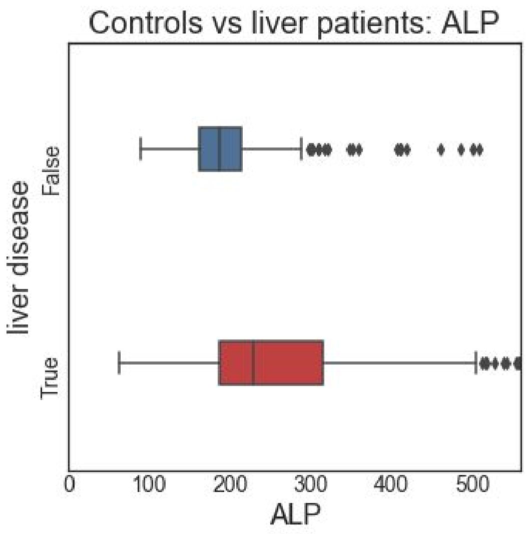

Controls vs liver patients: total bili


Distribution of direct bili

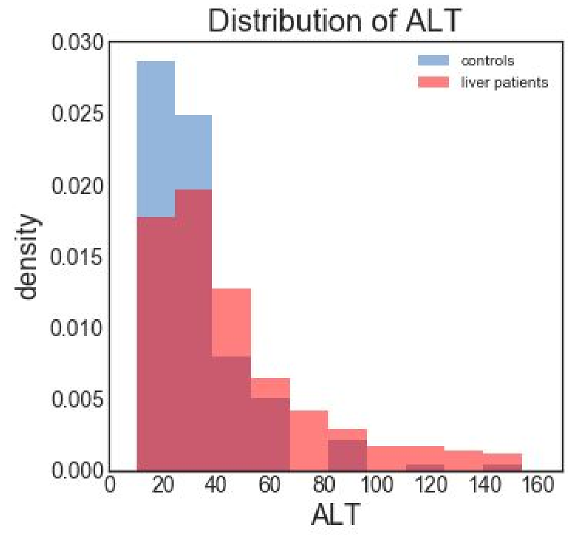

Controls vs liver patients: direct bili



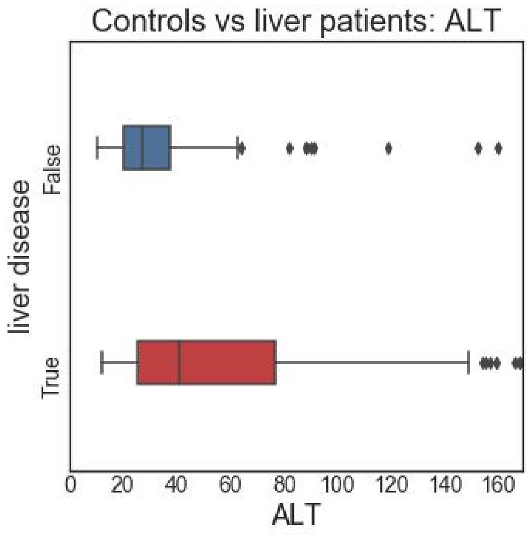
Distribution of ALP



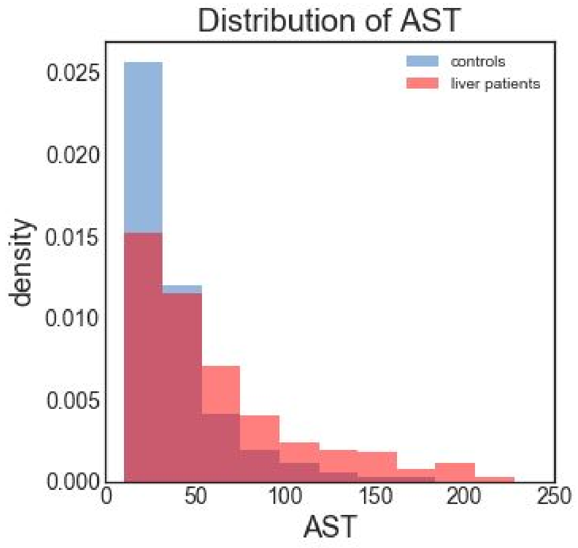
Controls vs liver patients: ALP



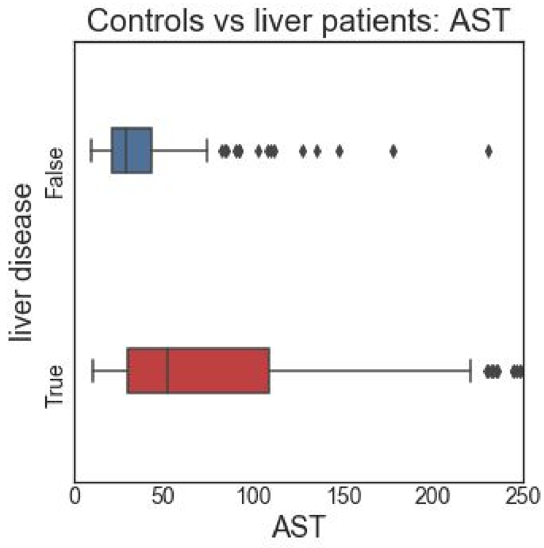
Distribution of ALT



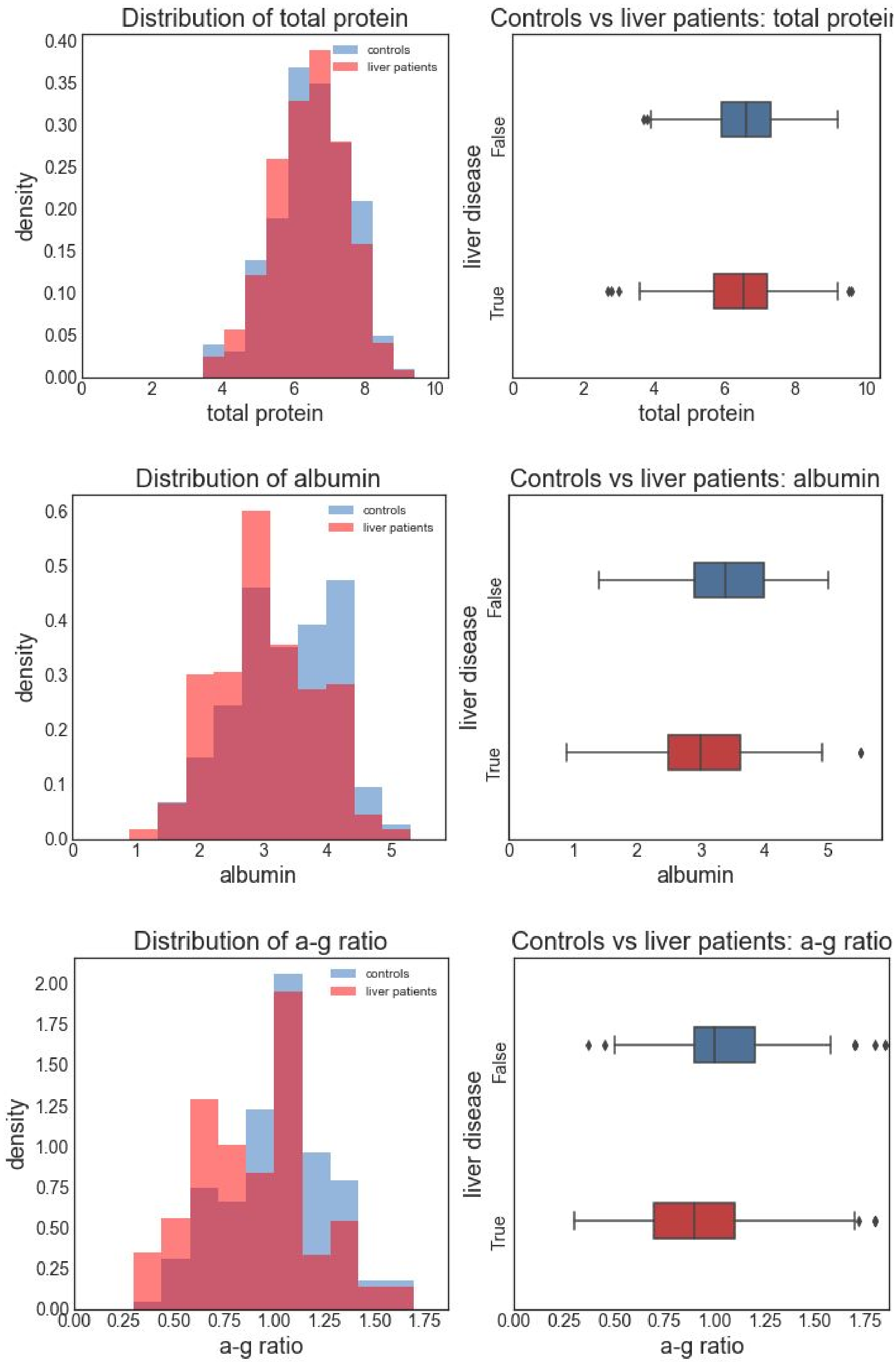
Controls vs liver patients: ALT



Distribution of AST



Controls vs liver patients: AST


Distribution of total protein
controls
liver patients
density
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
2
4
6
8
10
total protein
Controls vs liver patients: total protei
liver disease
False
True
0
2
4
6
8
10
total protein
Distribution of albumin
controls
liver patients
density
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
1
2
3
4
5
albumin
Controls vs liver patients: albumin
liver disease
False
True
0
1
2
3
4
5
albumin
Distribution of a-g ratio
controls
liver patients
density
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
a-g ratio
Controls vs liver patients: a-g ratio
liver disease
False
True
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
a-g ratio