Predicting Hemoglobin Concentration from Nail Bed Images

Olivia Fan, Tingnan Hu

15 December, 2022

1. Introduction

Anemia is as a life-threatening disease that affects 2 billion, or approximately 1 in 3 people world wide (Mannino 2018). In 2019, this long lasting disease accounted for 50.3 million cumulative YLD (years lived with disabilities) (Safiri 2021). Patients suffering from chronic anemia require frequent monitoring of indicators such as the Hgb concentration in blood to track the progression of their disease. Despite high prevalence, however, the current diagnostic process requires blood tests which causes discomfort and trauma in patients, in addition to incurring high monetary cost, especially in regions of lower socio-economic development (Safiri 2021). Nosratnejad et al. finds cost-effectiveness of anemia screening, total intervention costs for anemia treatment during a person's lifetime can amount to $3575, while treating anemia can cost between $18 and $500 per month depending on the type of anemia and necessary treatment (Nosratnejad 2014). Therefore, we aim to create machine learning models that enable non-invasive inexpensive diagnosis using patients' nail bed images to predict their Hgb concentration.

Previous work (Mannino 2018) considers three predictors of interest to predict Hgb concentration: the fingernail beds, the conjunctiva and the palmar creases. In this study, we aim to reevaluate the predictors identified, as well as explore new predictors significant to predicting hemoglobin concentration. In this study, we investigate machine learning techniques for non-invasive anemia diagnosis by re-evaluating the predictive ability of previously selected predictors and proposing new predictors, via comparing our results with previous work to examine the correlation between the features and the hemoglobin concentration levels. We extend the previous research by developing a multilinear LASSO regression model, using features extracted from the nail bed images as predictors and cross validation for hyperparameter tuning.

2. Methodology: LASSO Multilinear Model

Data

We examine 72 nail bed images collected from patients enrolled in Dr. Nirmish Shah's clinic at Duke University Hospital. Each patient has four images corresponding to different fingers. In collaboration with an ongoing Bass Connections team, we processed the images such that the nail bed is captured in a bounding box while the background of the image is discarded. Color information of each pixel is extracted from the bounded nail bed images as features.

We know that each pixel can be represented by Red, Green, and Blue (RGB) values, but the RGB color space contains both color and light information, which is different for each image since photos are taken at different times and settings. To eliminate the inconsistency caused by variation in lighting and background, we used two additional color spaces – Hue, Saturation, Value (HSV) and Lightness, A, B (LAB) – to separate the color information from the lighting information. We computed the mean of each value/channel across the bounded nail bed image for each of these three color spaces (HSV, LAB, RGB) and used them as our model input. Our response variable is blood hemoglobin concentration associated with each nail bed in g/dL. Our predictor variables include:

Variable Name	Description
Mean value of Hue	Average value of Hue component (the color component / base pigment) of the Hue-Saturation-Value color space
Mean value of Saturation	Average value of Saturation component (amount of color / depth of the pigment / dominance of hue) of the Hue-Saturation-Value color space
Mean value of Value	Average value of Value component (brightness of the color) of the Hue-Saturation-Value color space
Mean value of Lightness	Average value of Lightness component from black to white on a scale of 0 - 100 of the LAB color space
Mean Value of A	Average value of representation of greenness to redness on a scale of -128 to +127 of LAB color space
Mean Value of B	Average value of representation of blueness to yellowness on a scale of -128 to +127 of LAB color space
Mean Value of R	Average value of redness of Red-Green-Blue color space
Mean Value of G	Average value of greenness of Red-Green-Blue color space
Mean Value of B	Average value of blueness of Red-Green-Blue color space

Table 1: Variable name and descriptions

Due to confidentiality agreement with Dr. Shah' clinic, we are unable to provide the complete raw image data. We display a glimpse of the pre-processed data below.

```
## Rows: 72
## Columns: 15
## $ Image_URL     <chr> "https://storage.labelbox.com/cklyhytg2ulao0774uvyw0poy%~
## $ xmin          <dbl> 248, 340, 469, 644, 347, 511, 640, 744, 201, 322, 458, 6~
## $ xmax          <dbl> 292, 396, 527, 696, 385, 553, 682, 774, 248, 376, 509, 6~
## $ ymin          <dbl> 537, 465, 414, 426, 398, 413, 472, 532, 767, 796, 700, 4~
## $ ymax          <dbl> 582, 523, 474, 482, 461, 472, 522, 570, 803, 838, 726, 5~
## $ Mean_H        <dbl> 8.1922, 13.0864, 12.3233, 13.1862, 11.3486, 11.9142, 10.~
## $ Mean_S        <dbl> 0.2842, 0.2684, 0.2863, 0.2962, 0.2670, 0.2686, 0.2695, ~
## $ Mean_V        <dbl> 0.7824, 0.8003, 0.7863, 0.7895, 0.7871, 0.7820, 0.7820, ~
## $ Mean_L        <dbl> 68.7689, 68.3792, 68.0705, 68.5229, 68.4438, 68.2958, 67~
## $ Mean_A        <dbl> 15.4397, 13.9356, 15.3458, 14.7066, 14.6354, 14.0090, 14~
## $ Mean_B        <dbl> 10.6895, 12.0016, 11.8946, 14.3816, 11.9706, 11.9289, 11~
## $ Mean_Prop_R   <dbl> 0.3999, 0.3980, 0.3943, 0.3994, 0.3972, 0.3997, 0.3977, ~
## $ Mean_Prop_G   <dbl> 0.3077, 0.3125, 0.3147, 0.3140, 0.3113, 0.3115, 0.3107, ~
## $ Mean_Prop_B   <dbl> 0.2925, 0.2895, 0.2910, 0.2866, 0.2915, 0.2888, 0.2916, ~
## $ concentration <dbl> 9.5, 9.5, 9.5, 9.5, 9.5, 9.5, 9.5, 9.5, 9.2, 9.2, 9.2, 9~
```

Exploratory Data Analysis

Concentration of Hemoglobin in g/dL
Min. : 9.20
1st Qu.: 9.50
Median : 9.90
Mean :10.43
3rd Qu.:10.20
Max. :13.30

Table 2: Quantiles of concentration of Hemoglobin (g/dL)

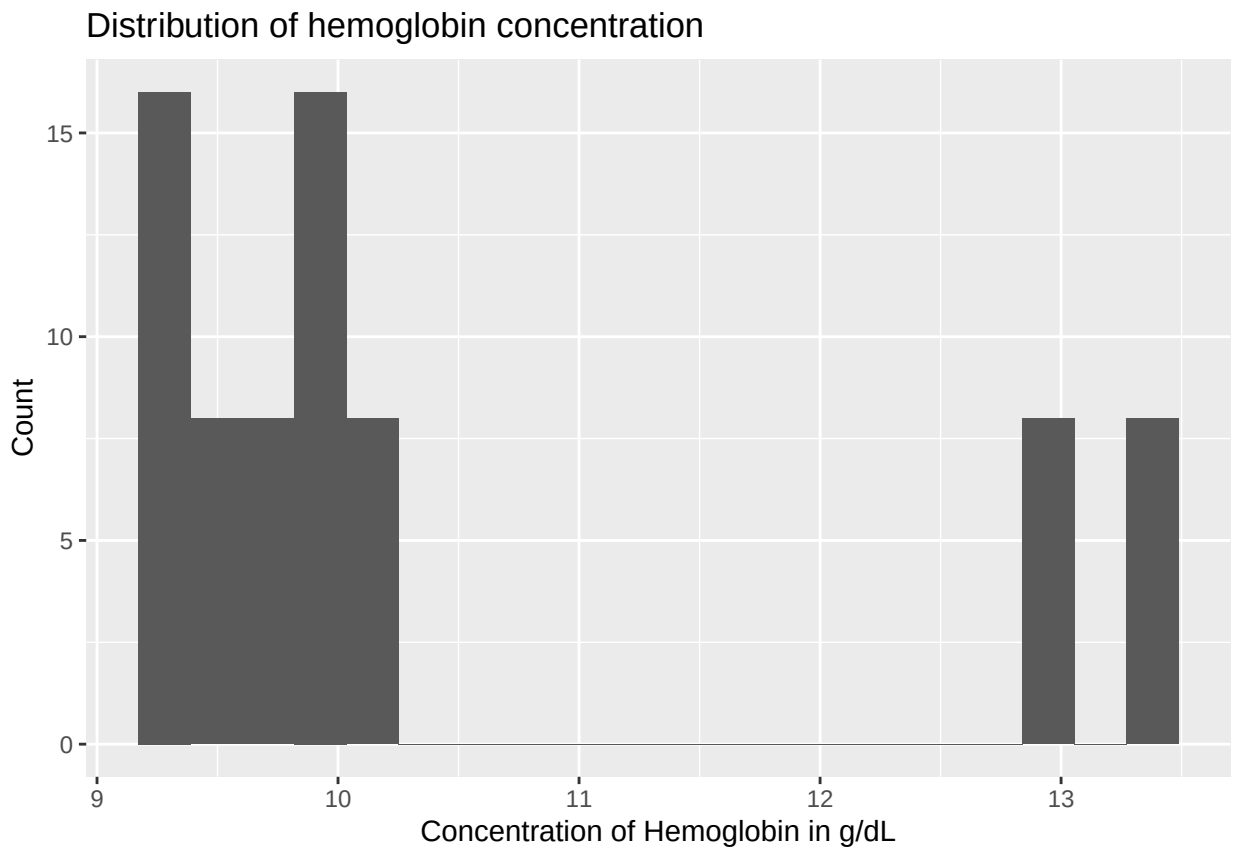

Figure 1: Distribution of response variable

Figure 1 and the corresponding summary statistics show that the Hemoglobin concentration in g/dL ranges from 9.20 g/dL to 13.30 g/dL, with a mean of 10.43 g/dL and median of 9.90 g/dL, which better captures the center of the distribution since it is right-skewed.

Cross Validation for Hyperparameter Tuning

In order to perform variable selection to gauge insight into the predictors which have significantly associations with hemoglobin concentrations, we use the LASSO which uses L_1 norm penalty, shrinking the coefficient estimates of insignificant predictors towards zero by minimizing $\{\Sigma(y_i - \hat{y}_i)^2 + \lambda\Sigma_i|\beta_i|\}$ (Tibshirani 1996). For this reason, we standardize the data prior to finding LASSO estimates. We chose the LASSO model over the best subset selection model because the LASSO model is more robust and less sensitive to changes in the dataset, and over ridge regressions because we would like to filter out predictors by setting β_i to exactly zero through variable selection (Tibshirani 1996).

The LASSO minimizes $\{\Sigma(y_i - \hat{y}_i)^2 + \lambda\Sigma_i|\beta_i|\}$, the residual sum of squares plus a shrinkage penalty of lambda multiplied by the sum of absolute values of the coefficients in which λ is a hyperparameter that we tune via cross validation.

$$y = \Sigma x_i\beta_i + \beta_0 + \lambda\Sigma|\beta_i|$$

We use cross validation to tune the hyperparamter λ and visualize the shrinkage of the coefficients.

We observe that the best λ which results in the smallest MSE is 0.0584

```
## [1] 0.05840522
```

	coefficient
(Intercept)	15.4670
Mean_H	0.0045
Mean_S	11.6495
Mean_L	0.0124
Mean_Prop_G	-30.2895

As a result, we derive a sparse model which only involves a subset of the features extracted from the nailbed images.

Predictions

```
lasso.pred <- predict(lasso.mod, s = bestlam, newx = x[test,])
mean((lasso.pred - y.test)^2)
```

```
## [1] 14.01923
```

why* you do a training vs test model and how you partitioned the data Performing predictions on the test data set, we derive a test MSE of 14.019.

Discussion of Model Output

We have derived the following linear model:

$$\hat{Hgb} = 15.4670 + 0.0045 \times Mean\_H + 11.6495 \times Mean\_S + 0.0124 \times Mean\_L - 30.2895 \times Mean\_Prop\_G$$

We notice that while variables mean value of hue, mean value of saturation, and mean value of lightness are positively associated with the response variable `Hgb` concentration, Average value of greenness of Red-Green-

Blue color space is negatively associated with the response variable. This means as mean value of hue, mean value of saturation, or mean value of lightness increases, the hemoglobin concentration level also tends to increase. On the other hand, as the mean value of greenness of Red-Green-Blue color space increases, the hemoglobin concentration level tends to decrease.

We find that out of all the predictors, `Mean\_Prop\_G` tends to associates most significantly with the response variable with the coefficient with the largest absolute magnitude. For each 1 unit increase in the mean value of greenness of RGB color space, Hgb concentration is expected to decrease by 30.2895 on average, keeping all else constant.

3. Conclusion

In this study, we explored how multilinear regression can be used to predict hemoglobin concentration level based on features derived from patients' nail beds images. The LASSO model selects 4 predictors (mean value of Hue, mean value of Saturation, mean value of Lightness, and mean value of representation for greenness) among the 9 predictors. (Mannino 2018) finds that representation of blueness in RGB color space is not a significant predictor for blood hemoglobin levels. This notion is consistent with our result that mean value of representation of blueness is not included in the final model. Among the 4 predictors chosen by LASSO model, mean value of Saturation has the strongest positive relationship with the response, and mean value of representation for greenness has a negative relationship with the response. In the future, we would like to explore more on these two predictors.

Limitations of our work includes that the data from the clinic lacks labels, so we are unable to use classification methods such as random forest or SVM. Further, we only have 18 (72/4) effectively independent observations for Hg concentration, which might lead to poor model performance. Our future work includes collecting more data, fitting Bayesian regressions on the data, exploring more machine learning methods such as random forest or support vector machines, and using other shrinkage methods such as ridge regression. We would also like to compare the utility of adding features such as those extracted by a convolutional neural network (CNN). To begin a comparison, we ask: - How much variability is explained by the principal components of CNN feature set? It would be interesting to compare these principal components to the principal components of the features outlined in section 2.1 but will required additional data processing. See appendix for a detailed description of our preliminary exploration.

Appendix

Prior to having obtained data used in main analysis, we explored the TBND_V2 (Transient Biometrics Nails Dataset) on Kaggle, which contains unlabeled nail bed images (Barbosa, Theoharis, and Abdallah 2019).

With the unlabelled data, we try unsupervised clustering methods such as Kmeans clustering to gain some insights on classifying nail bed images. We use VGG16, a convolutional neural network (CNN) to exxtract features from the input images, turning each image into a feature vector with 4096 entries. We remove the final (prediction) layer from the neural network manually, and the new output layer is a fully-connected layer with 4,096 individual nodes. We do this by specifying the "outputs" argument when initialising the model. We therefore get input of our model by using the neural net VGG16 as a feature extractor for the image data.

We then perform a principal component analysis (PCA) on the feature vectors to reduce the dimension of the feature space. For each of the 93 image samples, we now have a corresponding 1 by 4096 feature vector. This means that our model needs to process a 93 by 4096 matrix. To reduce the computational and complexity cost of processing high-dimensional data, we perform a principal component analysis (PCA) on the matrix for dimension reduction. We set the parameter to 50 to obtain the top 50 principal components of the feature vector. The principal components are by default sorted in descending order. This means that the first principal component will be able to explain the most variability in the feature vector. It's a linear combination of the feature variables and its direction captures the most variability. Thus, PCA helps us to reduce the dimension of the features from 4096 to 50 while preserving as much information in the original data as possible.

PCA Results

Based on the documentation, the explained_variance_ratio_ function returns the percentage of variability explained by each of the selected components. Running this function gives us the amount of variability that is explained by all the PCs (0.1015 is explained by the first PC, 0.0894 by the second, and 0.0781 by the third etc.) The table below shows the top 10 principal components:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
0.10152	0.08947	0.07806	0.06449	0.05609	0.04782	0.04122	0.03169	0.03162	0.02549

We report a bar chart to represent the variability explained by different principal components, as well as the cumulative step plot to represent the variability explained by the first most important components.

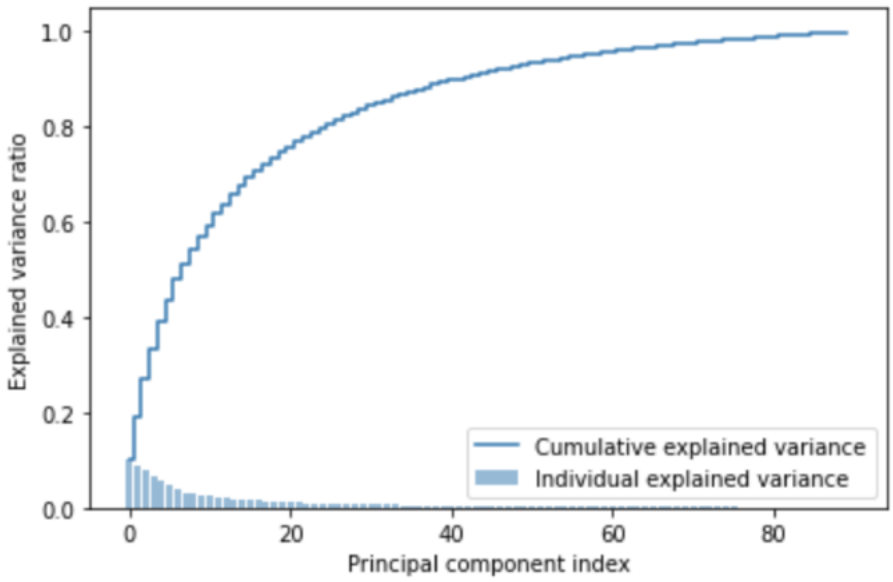

Figure 2: PCA cumulative step plot (10 Principle Conponents from CNN)

The neural net in the model functions as a feature extractor for the image data, which is the input of our model. To be more specific, we used VGG16, a convolutional neural network (CNN) in our model. It extracts features from the input images, turning each image into feature vectors (4096 by 1). We removed the final (prediction) layer from the neural network manually, and the new output layer is a fully-connected layer with 4,096 individual nodes. We do this by specifying the "outputs" argument when initialising the model.

References

Barbosa, Igor Barros, Theoharis Theoharis, and Ali E. Abdallah. 2019. "TBND_V2." Kaggle. https://doi.org/10.34740/KAGGLE/DS/309682.

Mannino, Robert G. 2018. "A NONINVASIVE, IMAGE-BASED SMARTPHONE APP FOR DIAGNOSING ANEMIA," 12.

Nosratnejad, Barfar, S. 2014. "Cost-Effectiveness of Anemia Screening in Vulnerable Groups: A Systematic Review." https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4124557/.

Safiri, Saeid. 2021. "Burden of Anemia and Its Underlying Causes in 204 Countries and Territories, 1990–2019: Results from the Global Burden of Disease Study 2019," 1. https://jhoonline.biomedcentral.com/articles/10.1186/s13045-021-01202-2.

Tibshirani, Robert. 1996. "Regression Shrinkage and Selection via the Lasso." *Journal of the Royal Statistical Society. Series B (Methodological)* 58 (1): 267–88. http://www.jstor.org/stable/2346178.