Group project: Prediction of Transplant Survival Rates

Equitably for Allogeneic HCT patients

Group O'liang

Group members

Chananon Wongkondee	6542023426
Peerapat Tintermsup	6542091026
Wayu Limsuwan	6542111026
Wisarut Peerachaidecho	6542113226
Sathapanan Sanarak	6542116126

Present

Asst. Prof. Suronapee Phoomvuthisarn, Ph.D.

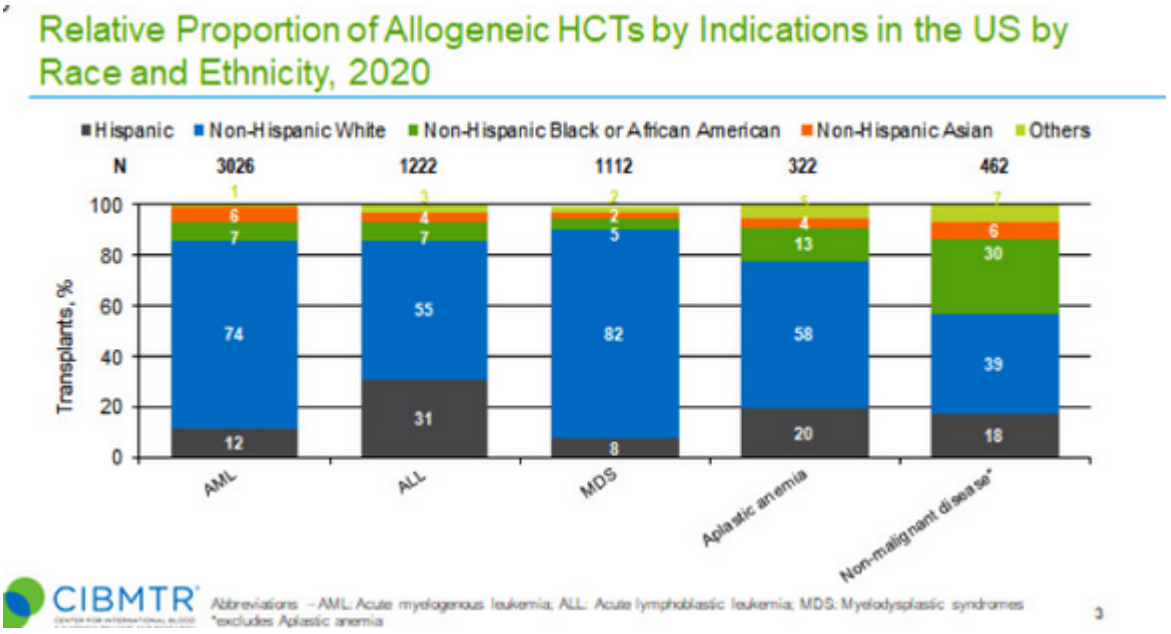
This report is part of the course
2603498 Data Science Practicum
Department of Statistics
Faculty of Commerce and Accountancy
Chulalongkorn University
2nd semester/2024

FRAMEWORK

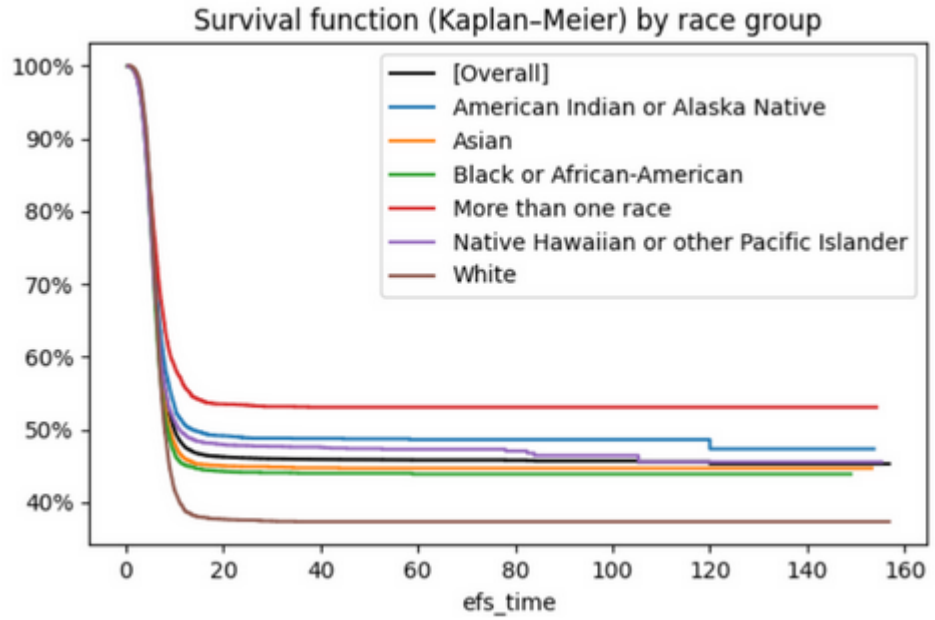
- Business Understanding
- Data Understanding
- Data Preparation
- Modeling
- Evaluation
- Appendix

Business Understanding

Hematopoietic Stem Cell Transplantation (HSCT) is a critical treatment for patients with blood disorders, but event-free survival (EFS) varies significantly across patient and donor characteristics. Medical institutions and research centers need to optimize treatment strategies and donor selection to improve patient survival outcomes.



How can we develop a predictive model that accurately ranks survival outcomes for allogeneic Hematopoietic Cell Transplantation (HCT) patients while minimizing racial disparities in prediction accuracy to support equitable healthcare delivery?



From the data set This chart uses Kaplan-Meier curves to show how the probability of event-free survival differs across various race groups. It visually suggests that some racial groups may experience different survival outcomes over time following the transplant.

How would a Data Science solution help?

1. Improved Donor Matching:

- A predictive model can analyze HLA compatibility more accurately by considering a wider range of genetic markers and demographic factors. This can increase match rates for minority patients.

2. Reducing Bias in Predictions:

- By using fairness-aware metrics like the Stratified Concordance Index, the model can ensure that predictions are equally accurate across different racial groups, supporting fairer clinical decision-making.

3. Efficient Resource Allocation:

- Hospitals can allocate resources, such as transplant slots and post-transplant care, more effectively by predicting which patients are most likely to benefit and which patients are considered to be risky, helping reduce disparities in access.

DATA UNDERSTANDING

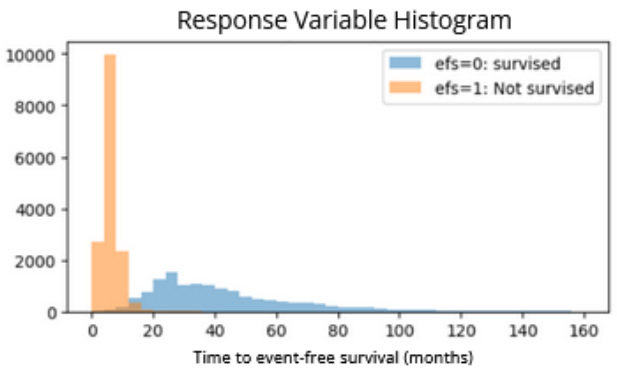
Input Variable

variable	description	type
dri_score	Refined disease risk index	Categorical
psych_disturb	Psychiatric disturbance	Categorical
cyto_score	Cytogenetic score	Categorical
diabetes	Diabetes	Categorical
hla_match_c_high	Recipient / 1st donor allele level (high resolution) matching at HLA-C	Numerical
hla_high_res_8	Recipient / 1st donor allele-level (high resolution) matching at HLA-A,-B,-C,-DRB1	Numerical
tbi_status	TBI	Categorical
arrhythmia	Arrhythmia	Categorical
hla_low_res_6	Recipient / 1st donor antigen-level (low resolution) matching at HLA-A,-B,-DRB1	Numerical
graft_type	Graft type	Categorical
vent_hist	History of mechanical ventilation	Categorical
renal_issue	Renal, moderate / severe	Categorical
pulm_severe	Pulmonary, severe	Categorical
prim_disease_hct	Primary disease for HCT	Categorical
hla_high_res_6	Recipient / 1st donor allele-level (high resolution) matching at HLA-A,-B,-DRB1	Numerical
cmv_status	Donor/recipient CMV serostatus	Categorical
hla_high_res_10	Recipient / 1st donor allele-level (high resolution) matching at HLA-A,-B,-C,-DRB1,-DQB1	Numerical
hla_match_dqb1_high	Recipient / 1st donor allele level (high resolution) matching at HLA-DQB1	Numerical
tce_imm_match	T-cell epitope immunogenicity/diversity match	Categorical
hla_nmdp_6	Recipient / 1st donor matching at HLA-A(lo),-B(lo),-DRB1(hi)	Numerical
hla_match_c_low	Recipient / 1st donor antigen level (low resolution) matching at HLA-C	Numerical
rituximab	Rituximab given in conditioning	Categorical
hla_match_drb1_low	Recipient / 1st donor antigen level (low resolution) matching at HLA-DRB1	Numerical
hla_match_dqb1_low	Recipient / 1st donor antigen level (low resolution) matching at HLA-DQB1	Numerical
prod_type	Product type	Categorical
cyto_score_detail	Cytogenetics for DRI (AML/MDS)	Categorical
conditioning_intensity	Computed planned conditioning intensity	Categorical
ethnicity	Ethnicity	Categorical
year_hct	Year of HCT	Numerical
obesity	Obesity	Categorical
mrd_hct	MRD at time of HCT (AML/ALL)	Categorical
in_vivo_tcd	In-vivo T-cell depletion (ATG/alemtuzumab)	Categorical
tce_match	T-cell epitope matching	Categorical

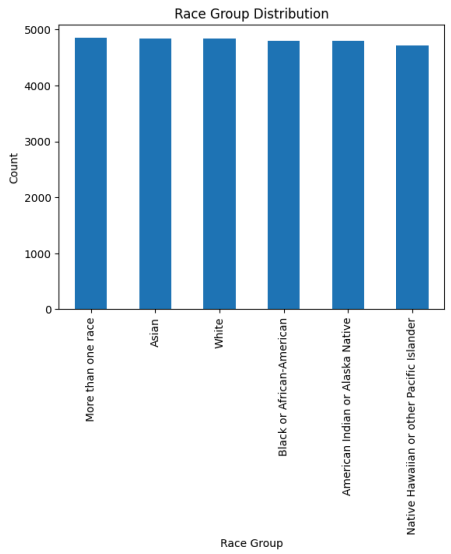
hla_match_a_high	Recipient / 1st donor allele level (high resolution) matching at HLA-A	Numerical
hepatic_severe	Hepatic, moderate / severe	Categorical
donor_age	Donor age	Numerical
prior_tumor	Solid tumor, prior	Categorical
hla_match_b_low	Recipient / 1st donor antigen level (low resolution) matching at HLA-B	Numerical
peptic_ulcer	Peptic ulcer	Categorical
age_at_hct	Age at HCT	Numerical
hla_match_a_low	Recipient / 1st donor antigen level (low resolution) matching at HLA-A	Numerical
gvhd_proph	Planned GVHD prophylaxis	Categorical
rheum_issue	Rheumatologic	Categorical
sex_match	Donor/recipient sex match	Categorical
hla_match_b_high	Recipient / 1st donor allele level (high resolution) matching at HLA-B	Numerical
race_group	Race	Categorical
comorbidity_score	Sorror comorbidity score	Numerical
karnofsky_score	KPS at HCT	Numerical
hepatic_mild	Hepatic, mild	Categorical
tce_div_match	T-cell epitope matching	Categorical
donor_related	Related vs. unrelated donor	Categorical
melphalan_dose	Melphalan dose (mg/m^2)	Categorical
hla_low_res_8	Recipient / 1st donor antigen-level (low resolution) matching at HLA-A,-B,-C,-DRB1	Numerical
cardiac	Cardiac	Categorical
hla_match_drb1_high	Recipient / 1st donor allele level (high resolution) matching at HLA-DRB1	Numerical
pulm_moderate	Pulmonary, moderate	Categorical
hla_low_res_10		Numerical

Response Variable

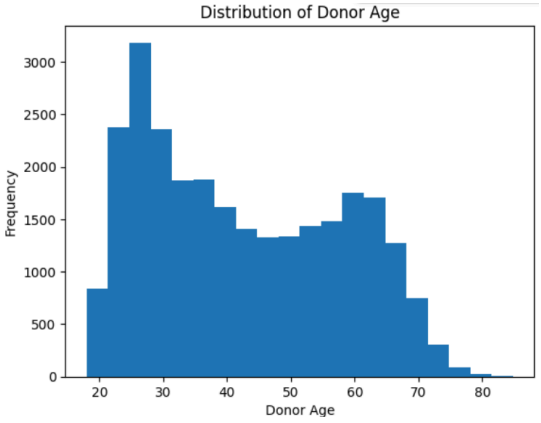
- Event-free Survival
    - (1 = not survived, 0 = survived)
- Time to event (months)



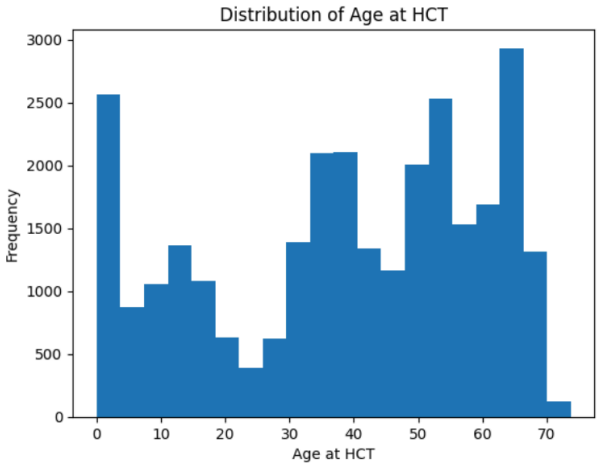
EDA



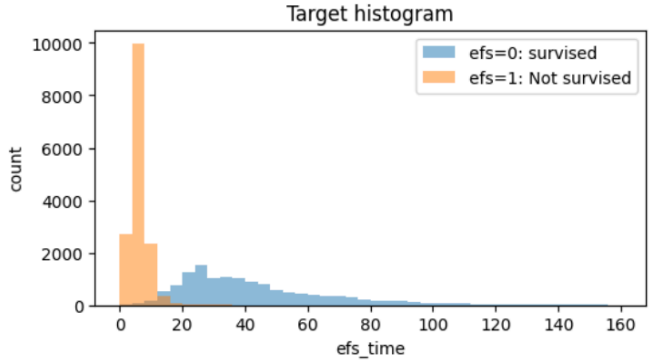
This chart shows a generally balanced distribution of patients across most racial groups (around 4800 each), with slightly fewer (around 4700) in the 'Native Hawaiian or other Pacific Islander' group.



Donor Age: Distribution skewed right, peaks in early 20s and late 50s/early 60s.



Age at HCT: Multimodal, high frequency in very young, peaks around 30s, 50s, and late 60s.

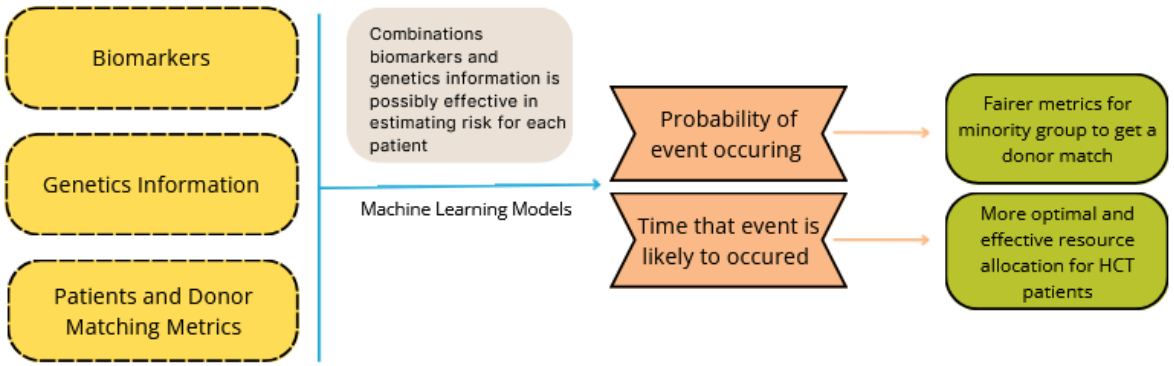


Events (Not survived) usually happen early after the transplant. If someone survives without these events, they tend to live longer.


Sex and Ethnicity on Survival Rate
AVG(Efs)
0.000
1.000
Sex Match
Race Group
F-F
F-M
M-F
M-M
American Indian or Alaska Native
0.4670
0.5600
0.5432
0.4672
Asian
0.5092
0.6218
0.5559
0.5086
Black or African-American
0.5063
0.6202
0.5802
0.5100
More than one race
0.4275
0.5089
0.5027
0.4167
Native Hawaiian or other Pacific Islander
0.4825
0.5702
0.5597
0.4788
White
0.5933
0.6879
0.6582
0.5815


Age at Transplant and Donor Age Effect on Survival
Donor Age (bin)
Age At Hct (bin)
10
20
30
40
50
60
70
Avg. Efs
0.2121
1.0000
70
1.0000 1
0.8387 31
0.6552 29
0.4000 15
0.6667 15
0.8000 25
1.0000 4
60
0.6909 55
0.6614 1,335
0.6315 1,015
0.6206 709
0.6445 979
0.6644 1,198
0.7600 200
50
0.6184 76
0.6051 1,413
0.6090 862
0.6190 777
0.5891 976
0.6643 989
0.6029 136
40
0.5682 44
0.6303 925
0.5783 709
0.5774 601
0.6197 681
0.6415 569
0.5455 88
30
0.6066 61
0.5839 1,264
0.5576 1,180
0.5343 932
0.5544 799
0.5850 612
0.5049 103
20
0.6000 20
0.5399 376
0.5377 305
0.4587 218
0.5419 227
0.5879 165
0.5946 37
10
0.2121 33
0.4180 988
0.4379 628
0.4368 435
0.4770 369
0.4846 324
0.4419 43

Summary


Biomarkers
Genetics Information
Patients and Donor Matching Metrics
Combinations biomarkers and genetics information is possibly effective in estimating risk for each patient
Machine Learning Models
Probability of event occuring
Time that event is likely to occured
Fairer metrics for minority group to get a donor match
More optimal and effective resource allocation for HCT patients

DATA PREPARATION

DATA PREPROCESSING

1. Mapping Categorical Values

- Creates a dictionary to map old group values to new ones.
- Merges data with GVHD prophylaxis mapping.
- ex. {'diabetes': {'nan': 'Missing', 'Not done': 'Missing'}}

2. Handling Missing Data

- Drops columns with more than 10% missing values.
- Removes irrelevant columns like 'ID' and 'prod_type'.

3. Splitting Data for Train, Validation and Test Set

- Using 60/20/20 method

SPLIT DATASET	Training	Validation	Test
Percent	60%	20%	20%
Row	17280	5760	5760
Column	45	45	45

.

4. Handling Missing Values

- Imputing numerical missing values (excluding efs & efs_time).
- Creates _missing indicator columns.
- Replaces missing values with the median of training data.

5. Normalization

- Normalize numerical values by scales numerical features to [-1, 1] range.
- Prevents division by zero for constant columns.

6. Encoding Categorical Variables

- One-hot Encoding
- Converts categorical columns into dummy variables (get_dummies).
- Drops the first category to prevent multicollinearity.

7. Race Group Weighting

- Computes inverse frequency weights for race_group.
- Ensures balanced representation in modeling.

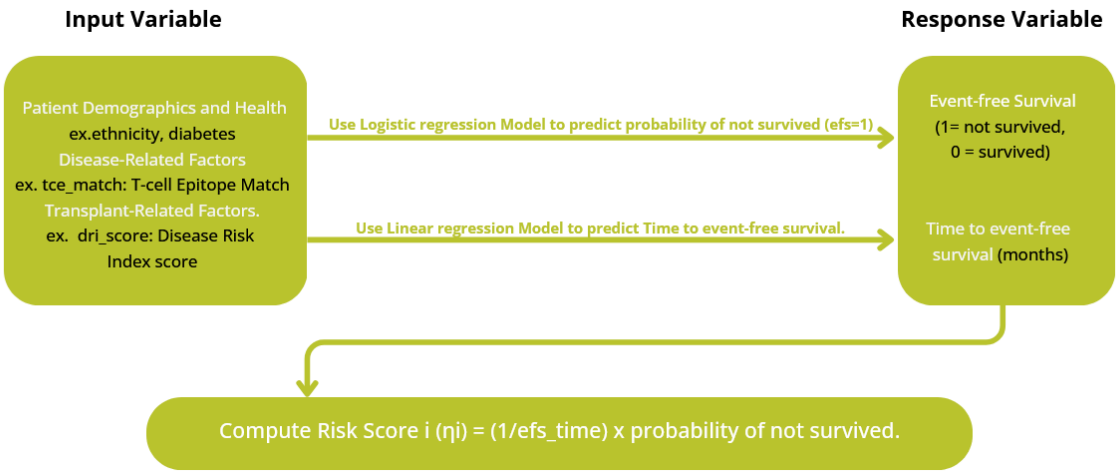
Modeling

Baseline Model

- Model Training: Fitting a Cox Proportional Hazards Model
- Target Variables: efs_time (duration), efs (event occurrence).
- Dropped Columns: Drop some missing indicator variables to reduce redundancy.
- Weights: weights_col to adjust for class imbalance.
- Robust Standard Errors: Used to improve model stability.

Model	**Stratified C-Index**	**C-index Asian**	**C-index Native Hawaii**	**C-index More than 1**	**C-index Black**	**C-index Indian+ Alaska**	**C-index White**
Cox Proportional Hazards Model	0.6437	0.6537	0.6725	0.6703	0.6497	0.6863	0.6335

Basic Model



- Linear Regression to predict efs_time
    1. Perform RandomizedSearchCV (5 folds, 30 iterations, alpha interval is loguniform(0.001,100), l1 ratio interval is np.linspace(0.01, 0.99, 20)) on the training set for lasso, ridge, elastic net to select the best parameter based on negative_mean_absolute_error.
    2. Evaluation on the test set.

	Model	Best Alpha	MAE	MSE	Best L1 Ratio	Loss function
0	Lasso	0.006025	1.834612	8.124038	NaN	$\sum(y_i - \hat{y}_i)^2 + \alpha \sum \lvert w_i \rvert$
1	Ridge	70.721141	1.835839	8.143009	NaN	$\sum(y_i - \hat{y}_i)^2 + \alpha \sum w_i^2$
2	ElasticNet	0.007359	1.835503	8.151891	0.835263	$\sum(y_i - \hat{y}_i)^2 + \alpha \left[(1 - l_1) \sum w_i^2 + l_1 \sum \lvert w_i \rvert\right]$

- Logistic Regression to predict efs
    1. Perform RandomizedSearchCV (5 folds, 30 iterations, C interval is loguniform(0.001, 100), l1 ratio interval is np.linspace(0.01, 0.99, 20)) on the training set for lasso, ridge, elastic net to select the best parameter based on f1 score.
    2. Evaluation on test set

	Model	Best C	Accuracy	F1 Score	Best L1 Ratio	Loss function
0	L1 Logistic Regression	0.190700	0.679688	0.719306	NaN	$-\sum [y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] + C\sum \lvert w_i \rvert$
1	L2 Logistic Regression	0.001952	0.671701	0.720968	NaN	$-\sum [y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] + C\sum w_i^2$
2	ElasticNet Logistic Regression	0.001009	0.664410	0.720139	0.01	$-\sum [y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)] + C\left[(1-l_1)\sum w_i^2 + l_1 \sum \lvert w_i \rvert\right]$

- Combine linear regression and logistic regression to get individual Risk score
    1. Calculate Stratified C-Index (Race) of all pair combination of linear regression and logistic regression
    2. Select best pair combination on validation set (table 1.1)
    3. Evaluate best combination on test set using C-index (table 1.2), C-index stratified (table 1.3)

1.1

	Linear Model	Logistic Model	Stratified C-Index (Race)
0	Lasso	L1 Logistic Regression	0.634669
1	Lasso	L2 Logistic Regression	0.632490
2	Lasso	ElasticNet Logistic Regression	0.629365
3	Ridge	L1 Logistic Regression	0.634359
4	Ridge	L2 Logistic Regression	0.631970
5	Ridge	ElasticNet Logistic Regression	0.628583
6	ElasticNet	L1 Logistic Regression	0.634969
7	ElasticNet	L2 Logistic Regression	0.632543
8	ElasticNet	ElasticNet Logistic Regression	0.629543

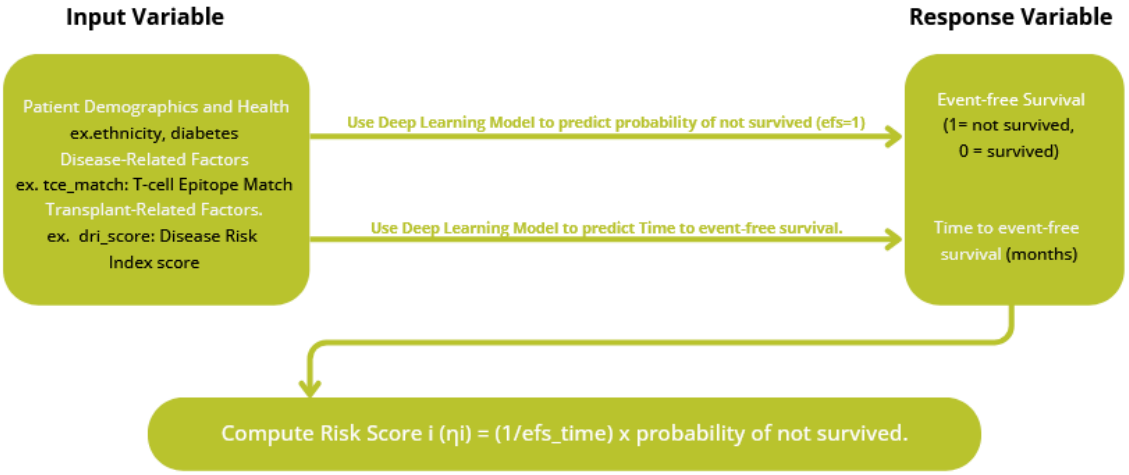
1.2

Linear Model	Logistic Model	C-Index	Stratified C-index
ElasticNet	L1 Logistic Regression	0.658	0.639911

1.3

	Race Group	C-Index
0	Asian	0.669224
1	Black or African-American	0.639673
2	More than one race	0.677235
3	Native Hawaiian or other Pacific Island	0.641569
4	White	0.639197
5	American Indian or Alaska Native	0.669345

Deep Learning Model



Classification Model (efs=1)

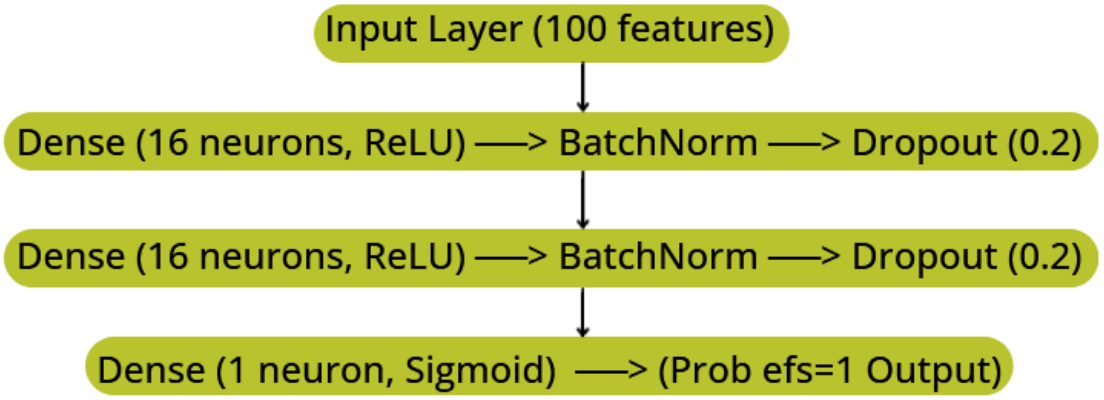
- use the same dataset from the baseline model.

Below this, there's a table showing the data split used for this classification model:

	Training	Validation	Test
Percent	60%	20%	20%
Row	17280	5760	5760
Column	100	100	100

- Architecture:
    - Input Layer: 100 features
    - Hidden Layers: 2x Dense (16 units, ReLU)
    - Batch Normalization & Dropout (0.2)
    - Output: Sigmoid Activation
- Training Details:
    - Loss: Binary Crossentropy
    - Optimizer: Adam
    - Metric: F1-Score

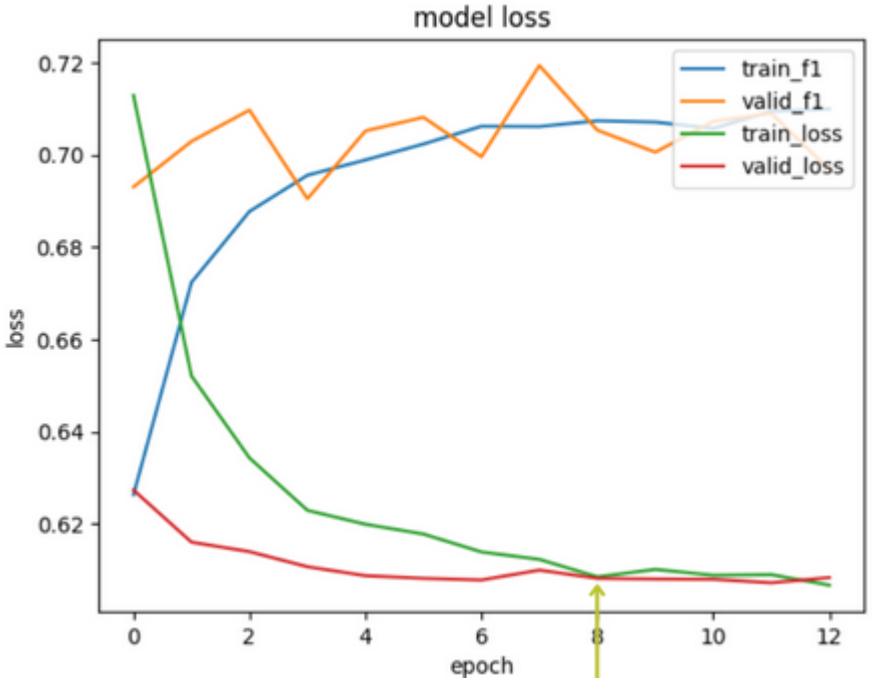
- Early Stopping: patience=5, monitor=val_f1_score
- Best Model Saved: best_model.h5



Below this, there's information about the training progress and final test set performance:

Epoch 8/50: train_loss: 0.7111 / valid_loss: 0.6101

Test set F1 Score: 0.7211842483472262



Regression Model (efs_time)

1. Filters Rows Where efs = 1:

- The regression model only predicts survival time (efs_time) for cases where the event occurred.
- A dashed orange arrow points from this description to the orange bars in the histogram on the right, which represent cases where 'efs = 1' (not survived). This visually reinforces that the regression model focuses on this subset of the data.

2. Drops Columns:

- efs_time: Target variable (separate for regression).
- efs: Not needed for regression since we're only keeping efs = 1 cases.
- weights_col: Used separately for sample weighting, likely not useful as a direct predictor in the regression model.
- Several _missing columns: Indicators for missing values, likely not useful for this specific regression task on the filtered data.

3. The filtered dataset is stored in:

- X_train_nn_efs_time: Training features
- X_valid_nn_efs_time: Validation features
- X_test_nn_efs_time: Test features

4. Extracts Sample Weights (weights_col) for Regression:

- weights_col_efs_time: Weights corresponding to cases where efs = 1.

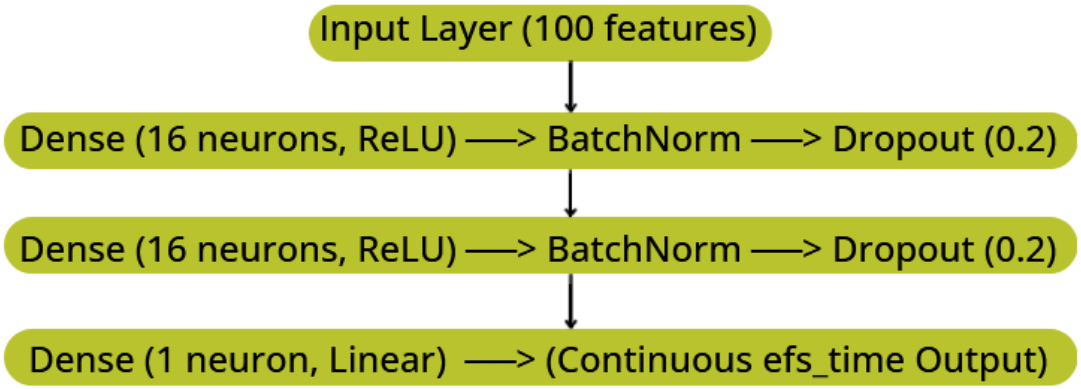
5. Scales the Target Variable (efs_time) Using MinMaxScaler:

Architecture:

- Input Layer: 100 features
- Hidden Layers: 2x Dense (16 units, ReLU)
- Batch Normalization & Dropout (0.2)
- Output: Linear Activation

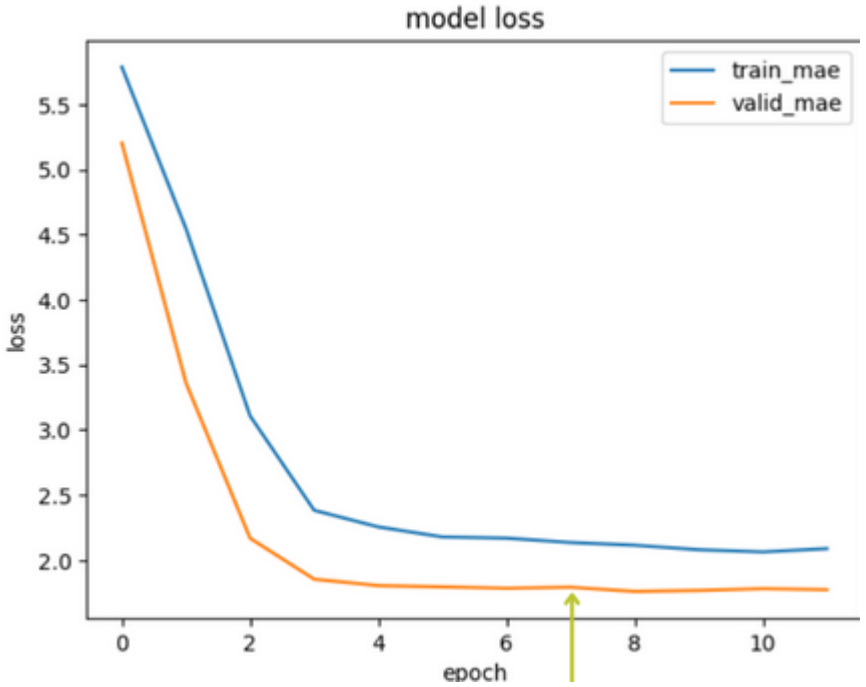
Training Details:

- Loss: Mean Squared Error (MSE)
- Optimizer: Adam
- Metric: Mean Absolute Error (MAE)
- Early Stopping: patience=5, monitor=val_loss
- Best Model Saved: best_model.h5



Below this, there's information about the training progress:

Epoch 7/50: train_loss (MSE): 9.9280 / val_mae: 2.1274



And then compute Risk Score i (ri) from the output of 2 model = (1 / efs_time) x probability of not survived.

Below this, there is a table showing the C-index values for this combined risk score model:

Model	Stratified C-Index	C-index Asian	C-index Native Hawaii	C-index More than 1	C-index Black	C-index Indian+ Alaska	C-index White
Train	0.6465	0.6767	0.6515	0.6584	0.6458	0.6610	0.6479
Validation	0.6295	0.6500	0.6245	0.6526	0.6253	0.6611	0.6458
Test	0.6435	0.6501	0.6661	0.6607	0.6445	0.6771	0.6399

Model Tuning using keras-tuner

- Hyperparameter Tuning: Optimize model performance using Bayesian Optimization (kt.BayesianOptimization).
- Searches optimal units, activation function, dropout rate, learning rate, number of layers.
- Validation Strategy: 60% Train | 20% Validation | 20% Test
- Early Stopping & Model Checkpoint used to prevent overfitting.

Below this, there is a table showing the best hyperparameters found for both the classification and regression models:

Best Hyperparameters	units	activation	dropout	lr	num layers
Classification Model (efs=1)	56	Leaky ReLU	0.2	0.0075	1

Regression Model (efs_time)	32	ReLU	0.0	0.0043	5

Model	**Stratified C-Index**	**C-index Asian**	**C-index Native Hawaii**	**C-index More than 1**	**C-index Black**	**C-index Indian+ Alaska**	**C-index White**
Deep Learning After Hyperparameter Tuning	0.6450	0.6490	0.6772	0.6643	0.6496	0.6769	0.6395

The final model improved Stratified C-Index by 0.233% compared to the model before tuning.

Evaluation

$$\text{Stratified Concordance Index} = \overline{\text{C-index}} - \sqrt{\frac{1}{g}\sum_{k=1}^{g}(\text{C-index}_k - \overline{\text{C-index}})^2}$$

$$C\text{-index} = \frac{\sum_{i,j} 1_{T_j<T_i} \cdot 1_{\eta_j>\eta_i} \cdot \delta_j}{\sum_{i,j} 1_{T_j<T_i} \cdot \delta_j}$$

with:

- η_i, the risk score of a unit i
- $1_{T_j<T_i} = 1, if\ T_j < T_i$ else 0
- $1_{\eta_j>\eta_i} = 1, if \eta_j > \eta_i$ else 0
- $\delta_j = 1, if\ efs_j = 1$ else 0

$$\overline{\text{C-index}} = \frac{1}{g}\sum_{k=1}^{g} \text{C-index}_k$$

with:

- g = Number of Total Race Group (More than one race, Asian, White, Black or African-American, American Indian or Alaska Native, Native Hawaiian or other Pacific Islander)

Why use Stratified C-index? : Because it specifically measures how consistently the model predicts survival risk *across different racial groups*, helping to identify and reduce potential unfairness or bias in the predictions.

Measures of Success:

The primary measure of success appears to be the Stratified C-Index. This metric combines the overall predictive ability (discrimination) of the model with a measure of its consistency across different racial groups.

- Higher Stratified C-Index: A higher value indicates better overall prediction of survival outcomes while also maintaining more similar predictive performance across the various racial categories. This suggests a more equitable model.

Additionally, the C-index for each individual racial group is important to monitor:

- Consistent C-indices across groups: Ideally, the C-index values for Asian, Native Hawaiian, More than 1 race, Black, Indian+ Alaska, and White patients should be relatively close to each other. Significant disparities would indicate that the model performs better for some racial groups than others, which would be a failure in achieving equity.

Model	**Stratified C-Index**	**C-index Asian**	**C-index Native Hawaii**	**C-index More than 1**	**C-index Black**	**C-index Indian+ Alaska**	**C-index White**
Baseline Model (Cox Proportional Hazards Model)	0.6437	0.6537	0.6725	0.6703	0.6497	0.6863	0.6335
Regression Model (ElasticNet + L1 Logistic Regression)	0.6399	0.6692	0.6397	0.6772	0.6416	0.63912	0.6693

Deep Learning	0.6450	0.6490	0.6772	0.6643	0.6496	0.6769	0.6395

- **Baseline Model:** Shows a Stratified C-index of 0.6437, with C-indices for individual racial groups ranging from 0.6335 (White) to 0.6863 (Indian+ Alaska).
- **Regression Model (ElasticNet + L1 Logistic Regression):** Has a slightly lower Stratified C-index of 0.6399. The range of C-indices across racial groups is wider, from 0.6397 (Native Hawaiian) to 0.6693 (White).
- **Final Model (Deep Learning):** Achieves the highest Stratified C-index of 0.6450. The C-indices for individual racial groups are relatively consistent, ranging from 0.6395 (White) to 0.6772 (Native Hawaiian).

In short, the table indicates that the Final Deep Learning Model demonstrates the best overall discriminatory ability (highest Stratified C-index) and also shows relatively consistent performance across the different racial groups compared to the other models. The Regression Model has a slightly lower overall performance and more variability across racial groups, while the Baseline Model falls in between.