Cohort analysis: Hypertension in children with T1D

In this report, we analyzed the Hypertension in children with T1D cohort to understand the elevated BP subjects. Specifically, we focused on :

- undercoving the determing factors, among potential clinical/demographic/family/immune/genetic variables, for elevated BP in children with T1D.
- developing a machine learning model to predict elevated BP in children with T1D.
- evaluating the model performance using various metrics for various models.

In conclusion, we build a ML model to predict elevated BP in children with T1D. The model is able to predict elevated BP with an impressive AUC of 0.96. Note that the performance is inflated due to the fact that older children in our cohort are almost unlikely to be with elevated BP group. Further details can be found at the end of this report.

Variables of interest

Response Variable

elevated_BP:

- Case group (Y=1): if the subject has elevated BP, i.e. patients who have any BP >90th % at both dates that BP was measured.
- Control group (Y=0): otherwise.

Predictors: Genetic Risk Score (GRS)

T1D genetic risk score 2 (GRS2) was calculated using the SNPs from GRS2 paper (Sharp et al. 2019).

- **Features generated:**
  - `IID` , `GRS` (standardized score)

Predictors: Ancestry & Family History

- Ethnicity (self-reported ethnic origin of 4 grandparents)
- Family history of T1D (yes/no)
- Family history of T1D, T2D or other forms of diabetes in 1st or 2nd degree relatives (yes/no)

- **Features generated:**
  - `cluster_pred_ancestry` (predominant continent where grandparents were born)
  - `cluster_shannon` (ancestry diversity): high value reflect higher diversity.
  - `Cluster_family_diabetes` ((yes/no))
  - `cluster_family_12_degree_bin` (Yes/No for 1st/2nd degree relatives)

Predictors: Clinical Features

- Age (months)
- BMI (SDS)
- Total daily insulin dose (units/kilogram/day)
- Personal history of other autoimmune disease (yes/no, type of autoimmune disease)
- Markers of diabetes control over the last 3 months: Hemoglobin A1C (%)

- **Features generated:**
  - `cluster_BMI`
  - `cluster_autoimmune_disease`
  - `Cluster_age`
  - `Cluster_insulin`
  - `Cluster_A1c`

Predictors: Antibody Data

- Glutamic acid decarboxylase 65 (GAD65) autoantibody titers (nmol/L)
- Insulinoma-associated antigen 2 (IA-2) autoantibody titers (nmol/L)
- Zinc transporter 8 (ZnT8) autoantibody titers (nmol/L)

- **Features generated:**
  - `96GAD`
  - `IA2`
  - `ZnT8`

Data Processing

- In total, we obtained 312 subjects, and 18 features of interest.
- We processed the highly correlated features (rho>0.8) and normalized the predictors.
- We split the data into training and testing sets (50/50), i.e. we randomly selected 50% of the data for training and 50% for testing.
- We used the training set to train the model and the testing set to evaluate the model performance.

Modeling Srategy

We applied and evaluated the following classifiers on a held-out test set using two evaluataion metrics: **ROC** curve and **Precision-Recall** curve.. The ROC AUC measures the model's ability to distinguish between the two classes, while the PR AUC focuses on the model's performance on the positive class (elevated BP).

We considered the follwing models:

1. Logistic Regression (standardized)

Standardized logistic regression fits a linear model linking predictor variables to the log-odds of the binary outcome. This model serves as our baseline: it’s fast to train, yields easily interpretable odds-ratios, and sets a performance floor against which more complex learners can be judged.

1. LASSO Logistic (`glmnet`)

LASSO logistic regression introduces an L_1 penalty on the magnitude of coefficients, shrinking many to exactly zero. This regularization both guards against over-fitting (especially when the number of predictors is large relative to sample size) and performs automatic feature selection by excluding weakly predictive variables. We fit via glmnet::cv.glmnet() using AUC-optimized cross-validation to select the penalty strength (λ). The resulting sparse model highlights only the strongest predictors, simplifying interpretation and often improving generalization.

1. Random Forest (caret)

Random forests build an ensemble of decision trees, each grown on a bootstrap-sample of the training set and considering a random subset of predictors at each split. By averaging across many decorrelated trees, the model captures non-linear relationships and high-order interactions while reducing variance. We used caret::train(method = "rf") with repeated cross-validation to tune the number of variables tried at each split (mtry). Random forests are robust to outliers, handle mixed-type features, and provide built-in measures of variable importance.

1. XGBoost (mlr)

XGBoost implements gradient boosting of decision trees, sequentially fitting each new tree to the residual errors of its predecessors. This method excels at capturing complex interactions and skewed distributions by optimizing a regularized objective (including L_1/L_2 penalties) under a fast, distributed framework. We wrapped it in mlr, tuning tree depth, learning rate, and sampling parameters via randomized search on PR-AUC. XGBoost often outperforms other algorithms when appropriately tuned, at the cost of more hyperparameter complexity. However, it turns out our data is not large enough to benefit from the XGBoost algorithm and the perofmance is dominated by the age factor. Therefore, we do not recommend using XGBoost for this dataset.

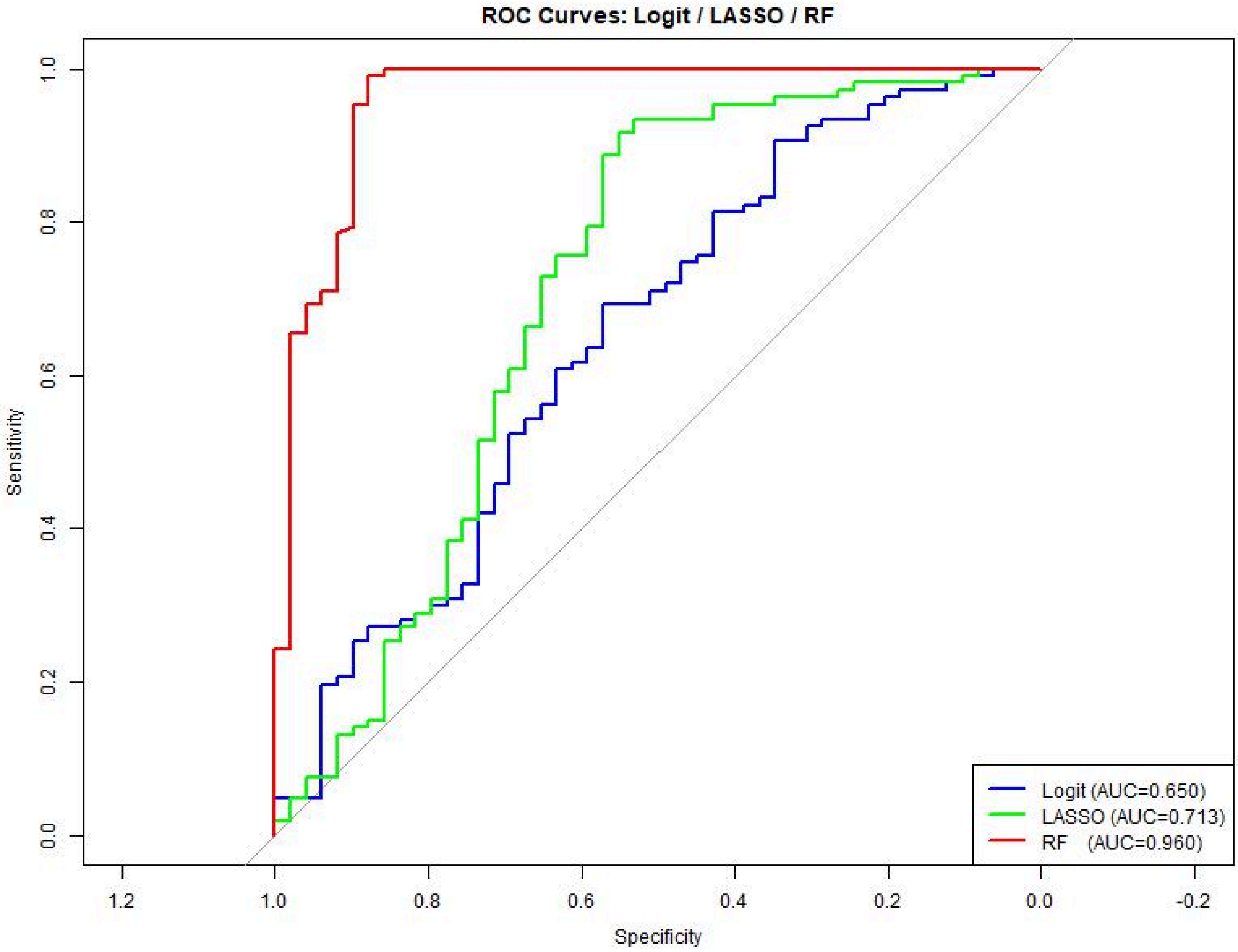
1. Linear SVM (caret)

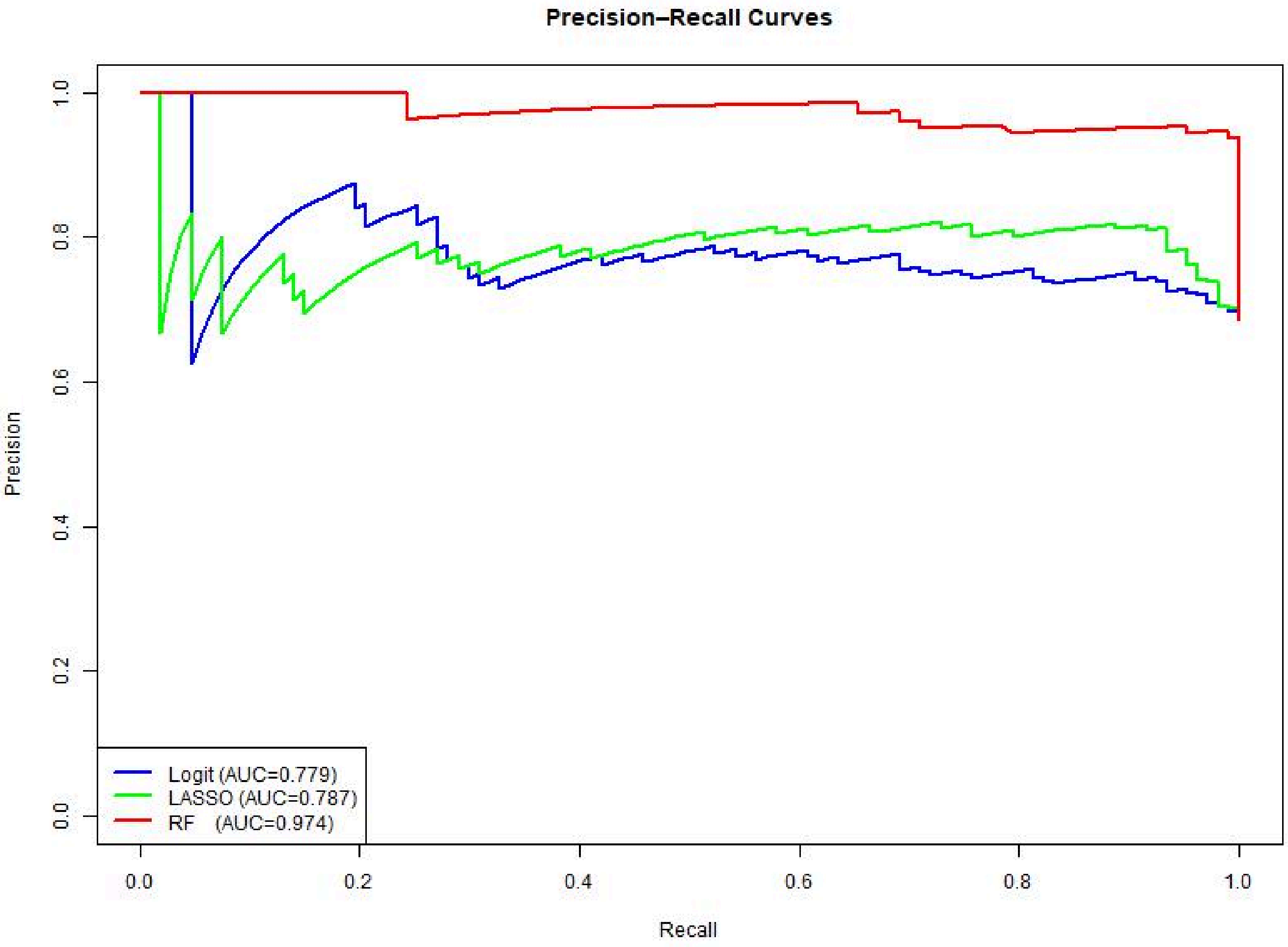
A linear Support Vector Machine seeks the hyperplane that maximizes the margin between classes in feature space. It's effectively a large-margin classifier that's robust to high-dimensional data and can be less sensitive to outliers than logistic regression. We trained with caret::train(method = "svmLinear"), tuning the cost parameter (C) via cross-validation to balance margin width against misclassification error. Although its decision boundary is linear, SVMs can yield strong performance in cases where classes are well-separated in a high-dimensional feature representation. However, due to limited samle size of this project, the performance of SVM is not as good as the Random Forest model on our dataset.

Conclusion

1. **Model Performance:**
   - **Random Forest** achieved the highest discrimination (ROC AUC ≈ 0.96, PR AUC ≈ 0.97) among all methods (Figure 1 and 2), markedly outperforming the simpler Logistic (ROC AUC ≈ 0.65, PR AUC ≈

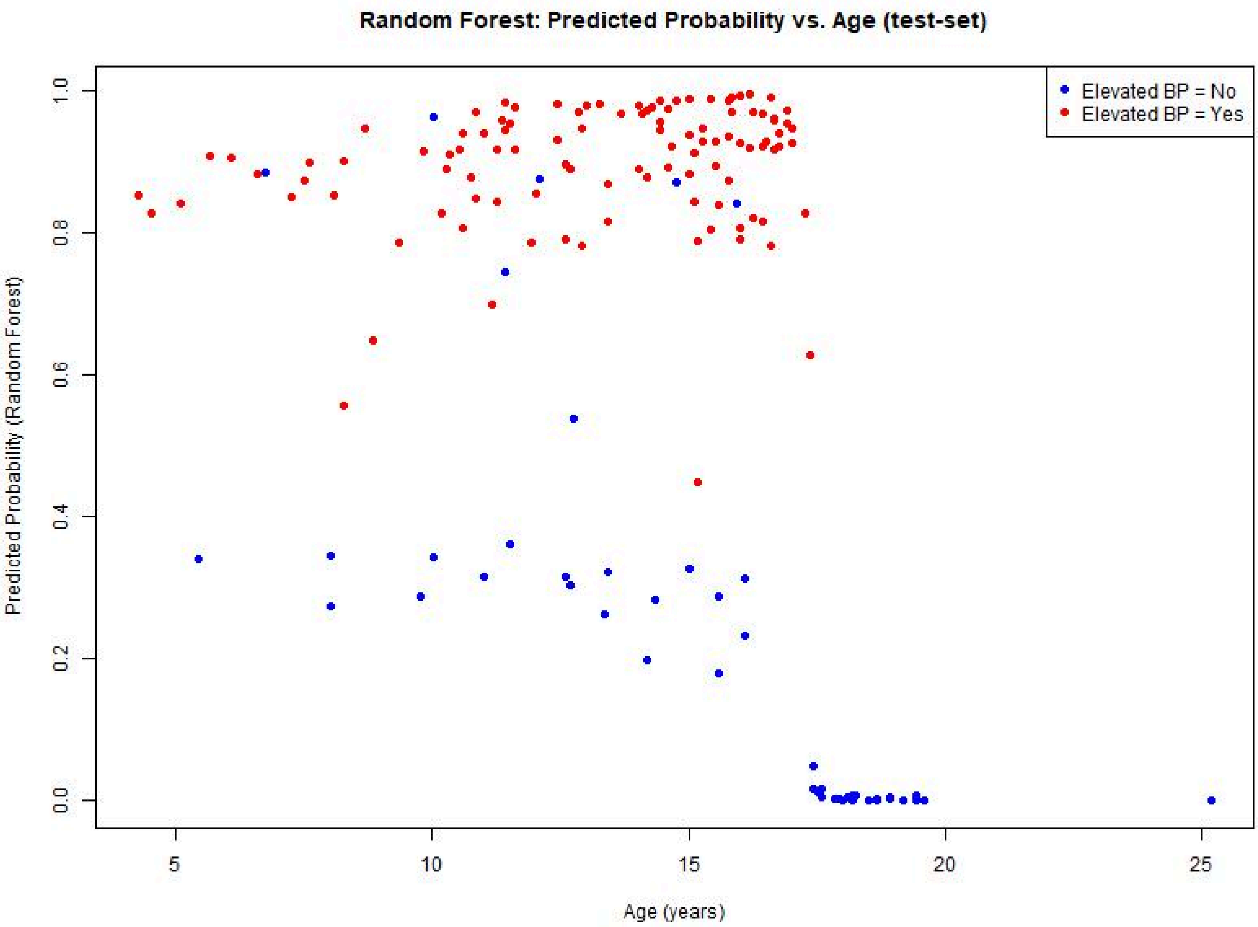
0.78) and succint LASSO (ROC AUC ≈ 0.71, PR AUC ≈ 0.79) models. The SVM and xgboost models performed similarly to the LASSO model and therefore not included in the figure below.

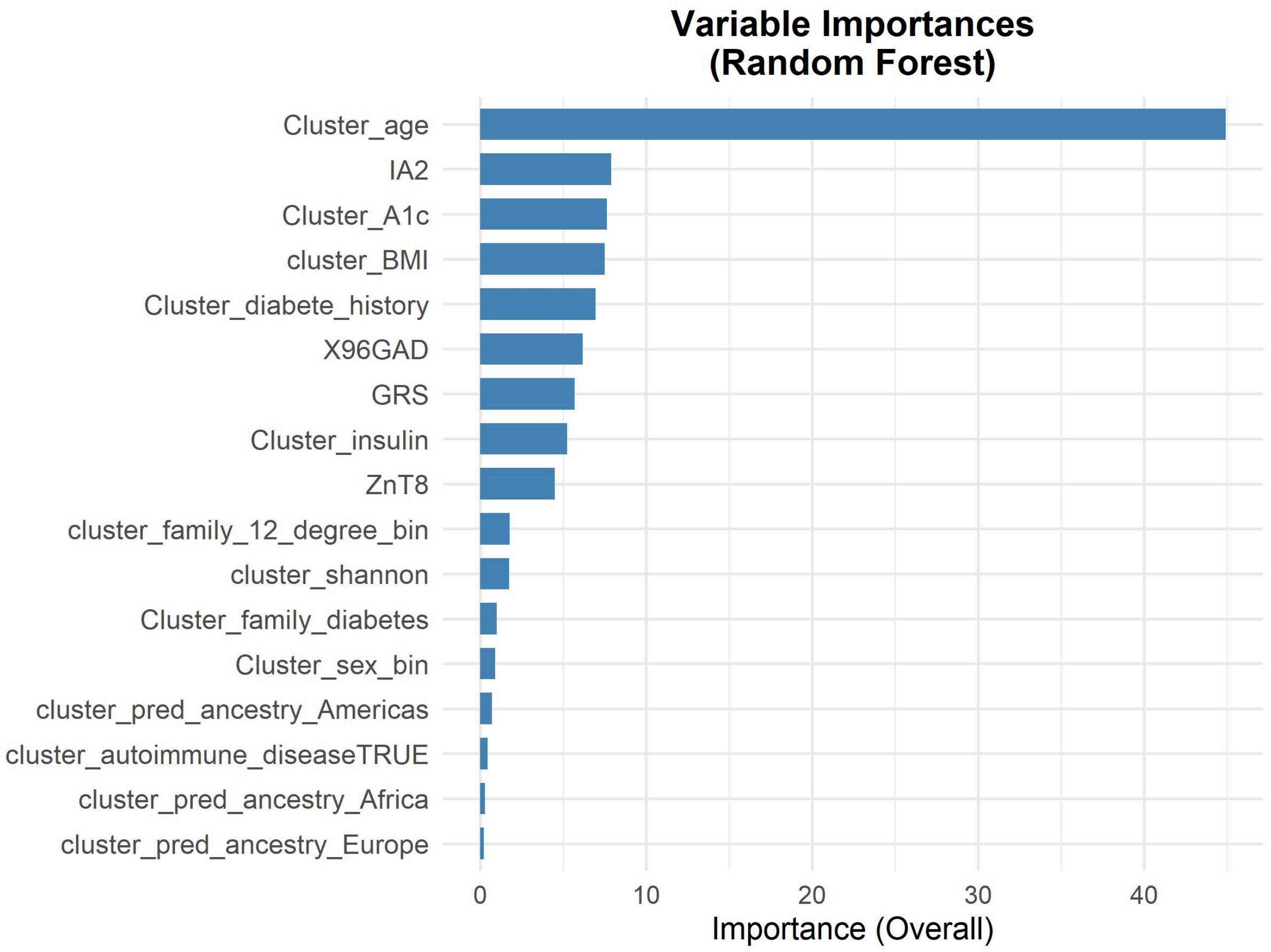




1. **Key Predictors:**

- **Age** was the most significant predictor in all models, since we observed that tall older-aged children are unlikely to have elevated BP (Figure 3). This explains why the ROC performance is so good.
- Across methods, **age**, **autoantibody titers (IA2, GAD, ZnT8)**, **insulin dose**, and **A1c** emerged as important features (Figure 4).
- Family history, ancestry, and genetic risk score contributed but were less influential in tree-based models.


Random Forest: Predicted Probability vs. Age (test-set)
Predicted Probability (Random Forest)
0.0
0.2
0.4
0.6
0.8
1.0
5
10
15
20
25
Age (years)
● Elevated BP = No
● Elevated BP = Yes



1. **Next Steps:**
   - **Updates:** The model need to be updated with new data Imputed basal C-peptide (nmol/l), -Insulin pump therapy (yes/no) and duration (months),...etc, which have not been incorporated yet.
   - **Validation:** The current model performance is evaluated only on one randomly selected training/test split. We need to test the final model with bootstrapp or permutation to assess generalizability and gauge uncertainty in performance metrics. i.e. we need to build confidence interval for the ROC and PR AUC metrics.