Analyzing Chronic Condition Risk Factors

Shubhpreet, Jay & Parabhuda

Course: Data Analysis STAT 4620/5620 – Winter 2024–2025
GitHub Repository: https://github.com/ShubhpreetDal/AnalyzingChronicConditionGroup5

```
library(AnalyzingChronicConditionGroup5)
```

2 Abstract

Chronic diseases such as high blood pressure, diabetes, and cardiovascular conditions pose a major public health burden in Canada. This project investigates whether these chronic conditions can be accurately predicted using survey-based data from the Canadian Community Health Survey (CCHS 2019–2020). We developed a comprehensive analysis pipeline to decode, clean, and transform categorical survey variables, followed by model training using interpretable and statistically robust techniques including logistic regression, decision trees, and generalized additive models (GAMs). After refining the data and addressing class imbalance, our models achieved strong recall across all three conditions, supporting their value as screening tools. Additionally, we explored associations between self-perceived health and chronic disease risk, revealing that physical health perception is a strong proxy for actual condition prevalence. Our analysis highlights the potential of survey-based modeling to support early risk identification in public health contexts.

3 Keywords

chronic disease, public health, logistic regression, feature engineering, Canadian Community Health Survey, bivariate analysis, ROC-AUC, classification, perceived health, R

4 Introduction

Chronic diseases such as high blood pressure, diabetes, and cardiovascular conditions affect millions of Canadians and pose a significant burden on the public health system. Early identification of individuals at risk is crucial for prevention and timely intervention.

This project leverages self-reported survey data from the Canadian Community Health Survey (CCHS 2019–2020) to explore two central research questions:

> **RQ1:** Can we predict whether an individual has high blood pressure, diabetes, or a cardiovascular condition using their responses to health-related survey questions?
>
> **RQ2:** How closely does an individual's self-perceived general or mental health align with actual risk of chronic conditions?

To address RQ1, we develop a reproducible machine learning pipeline in R that emphasizes interpretability, clinical logic, and rigorous evaluation. Each condition is modeled separately to reflect medical progression, and our process includes multi-stage feature engineering, class imbalance handling, and both statistical and visual diagnostics.

To address RQ2, we perform a comparative analysis between perceived health levels and actual disease prevalence, using chi-square tests and visualization to uncover behavioral or psychological proxies for chronic risk. Together, these analyses aim to support scalable public health screening and enhance understanding of disease awareness patterns.

5 Data Description

5.1 Survey background

We analyse the **Canadian Community Health Survey (CCHS) 2019–2020 Public Use Micro-data File (PUMF)** released by Statistics Canada.
The CCHS is a nationally representative, cross-sectional survey of Canadians aged 12 years +, that comprises

1. **Core content** – questions asked of *all* respondents, and
2. **Optional content** – thematic modules selected by provinces or territories.

Only core-content variables were retained to ensure comparability across regions and to minimise item non-response.

5.2 Decoding & feature-engineering workflow

Phase	Key action	R-package function(s)
1 Load & decode	Import raw PUMF and convert numeric codes to human-readable labels.	`load_cchs_data()` → `decode_cchs_values()`
2 Initial feature screen	Remove administrative fields, date stamps, and short-horizon logs (e.g., 7-day alcohol diary).	—
3 Core-only filter	Drop optional-module items to avoid regional sparsity.	—
4 Descriptive stats (A)	Generate baseline frequency tables & bar plots for all retained variables.	`describe_categorical()` + `generate_barplots()`
5 First categorical transformation	Harmonise ambiguous responses ("Not stated / Don't know / Refusal" → "Unknown"); recode logical skips.	`transform_categorical()`
6 Descriptive stats (B)	Re-run tables & plots to confirm Step 5 changes.	`describe_categorical()` + `generate_barplots()`
7 Final cleanup	Merge rare levels, replace residual "Unknown" values with the mode, domain-specific fixes.	`final_transform()`
8 Age-group review & filter	Quantify positives in the 12–17 year group; drop that group due to insufficient signal.	`analyze_underage_targets()` → `drop_underage_group()`
9 Save artefacts	Persist the cleaned data (`.rda` + CSV) and summary statistics for modelling.	`usethis::use_data()`

Outcome: Starting from ~100 000 respondent records and ~690 raw variables, the workflow yields **32 categorical predictors including the three target variables** that form the analytic base table.

5.3 Analytic Base Table variables

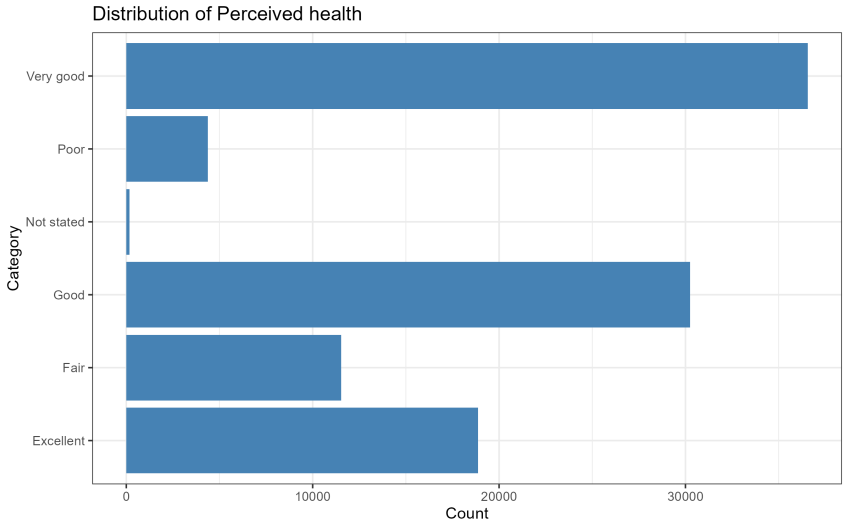
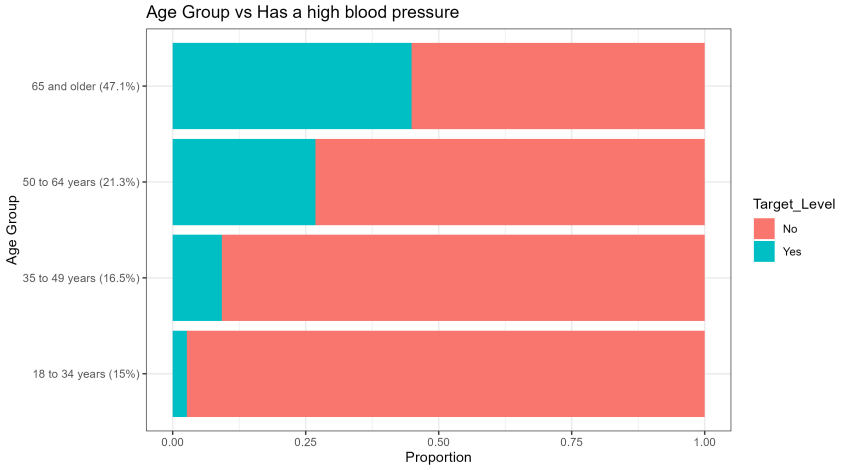
#	Variable (plain English)	Domain	Scale
1	AgeGroup	Demographics	Categorical
2	Sex at Birth	Demographics	Categorical
3	Marital Status	Demographics	Categorical
4	Considered suicide – lifetime	Mental health	Categorical
5	Considered suicide – last 12 months	Mental health	Categorical
6	Smoking status	Lifestyle	Categorical
7	Severity of Cannabis Dependence	Substance use	Categorical
8	Used cannabis – past 12 months	Substance use	Categorical
9	Usual place for immediate care	Access to care	Categorical
10	Total household income	Socio-economic	Categorical
11	BMI (12–17, WHO)	Health metrics	Categorical
12	BMI (18+, adjusted)	Health metrics	Categorical
13	Pain health status	Health status	Categorical
14	Perceived health	Health perception	Categorical
15	Perceived mental health	Mental health	Categorical
16	Satisfaction with life	Well-being	Categorical
17	Had seasonal flu shot – lifetime	Immunisation	Categorical
18	Seasonal flu shot – last time	Immunisation	Categorical
19	Type of drinker	Lifestyle	Categorical
20	5+/4+ drinks on one occasion – frequency	Lifestyle	Categorical
21	Has sleep apnea	Diagnosed conditions	Categorical
22	Has high blood cholesterol/lipids	Diagnosed conditions	Categorical
23	Cholesterol medication – past month	Medication use	Categorical
24	Has chronic fatigue syndrome	Diagnosed conditions	Categorical
25	Has a mood disorder	Mental health	Categorical

#	Variable (plain English)	Domain	Scale
26	Has an anxiety disorder	Mental health	Categorical
27	Respiratory chronic condition	Diagnosed conditions	Categorical
28	Musculoskeletal condition	Diagnosed conditions	Categorical
29	BP medication – past month	Medication use	Categorical
30	**Has a high blood pressure**	*Target 1*	Categorical
31	**Has diabetes**	*Target 2*	Categorical
32	**Cardiovascular condition (heart disease or stroke)**	*Target 3*	Categorical

Table 1. Variables retained after decoding and feature engineering.

5.4 Exploratory visualisation and chi-square screening

Our first pass combined simple bar plots with chi-square tests to understand class balance and bivariate relationships.

Example graphic	What it shows
	Distribution of self-rated physical health (imbalanced toward "Good/Very good").
	Normalised proportions of high-blood-pressure cases across age groups.

For **every** predictor we produced

- a chi-square test of association with each target (CSV), and
- a stacked-bar PNG visualising class proportions (`bivariate_plots/`).

All chi-square p-values were < 0.05 except three predictors for the High-BP target (`Sex at Birth`, `Has a mood disorder`, `Has an anxiety disorder`).
These variables were nonetheless retained for their clinical relevance.

---

6 Methods

6.1 End-to-end analytical workflow

Our analysis proceeds through multiple steps (Figure 2).
Each step is scripted in `analysis_pipeline.R` and relies on exported helper functions in the **AnalyzingChronicConditionGroup5** package.

Step	Purpose	Key functions / scripts
A Decode & clean	Convert raw CCHS codes to labels, remove age 12–17, harmonise unknowns, merge rare levels.	`load_cchs_data()`, `decode_cchs_values()`, `transform_categorical()`, `final_transform()`
B Exploratory summaries	Frequency tables and bar plots before and after cleaning.	`describe_categorical()`, `generate_barplots()`
C Bivariate screening	chi-square tests and stacked bar plots for each predictor–target pair.	`run_bivariate_analysis()`
D Baseline modelling	Fit logistic-regression and decision-tree benchmarks.	`run_baseline_models()`
E Regularised logistic regression	Re-fit best baseline with L2 penalty; export coefficients.	`train_and_save_model()`, `extract_logistic_coefficients()`
F Generalised additive models	Capture non-linear effects on seven top predictors.	`run_gam_on_top_features()`
G Discrimination assessment	ROC curves and AUC for each target.	`generate_all_logreg_predictions()`, `run_roc_auc_phase()`
H Perceived vs actual health	Compare self-rated health with disease prevalence.	`run_perceived_health_analysis()`

Figure 2. Steps in the analysis pipeline

6.2 Target-specific modelling strategy

Clinical progression suggests that high blood pressure can precede diabetes, which in turn can precede cardiovascular disease.
To respect this order and avoid information leakage, we build **three separate analytic tables**:

Target model	Columns *removed* before modelling	R code that implements the filter
High Blood Pressure	`has_diabetes`, `cardiovascular_condition_heart_disease_or_stroke`, `high_blood_pressure_took_medication_1_month`	`model1_df <- final_cleaned_data %>% select(-has_diabetes, -cardiovascular_condition_heart_disease_or_stroke, -high_blood_pressure_took_medication_1_month)`
Diabetes	`cardiovascular_condition_heart_disease_or_stroke`	`model2_df <- final_cleaned_data %>% select(-cardiovascular_condition_heart_disease`
Cardiovascular Condition	*None* (all predictors kept)	`model3_df <- final_cleaned_data`

6.3 Data partitioning and preprocessing

- **Train–test split** An 80% / 20% split is created per target with stratification on the outcome (`rsample::initial_split()`), ensuring equal class proportions in both sets.
- **One-hot encoding* All categorical predictors are expanded to 0/1 dummies inside each `recipes` object (`step_dummy(all_nominal_predictors())`).
- **Class imbalance** During model training we apply **random undersampling** of the majority class (`themis::step_downsample(all_outcomes())`). This preserves the original data distribution in the held-out test set while balancing classes during fitting.
- **Cross-validation** Five-fold CV (`vfold_cv(v = 5)`) is used for every model to estimate performance variability.

6.4 Baseline models

Model	Engine	Why included	Hyper-parameters
Logistic regression	`glm`	Interpretable linear baseline; coefficients map directly to odds ratios.	none
Decision tree	`rpart`	Transparent rule-based classifier capturing simple interactions.	default `cp`

CV metrics, test-set confusion matrices, and heat-map PNGs are written to `model_results/<target>/`. Figure 3 shows an example confusion-matrix heat-map for the High-BP decision-tree benchmark.

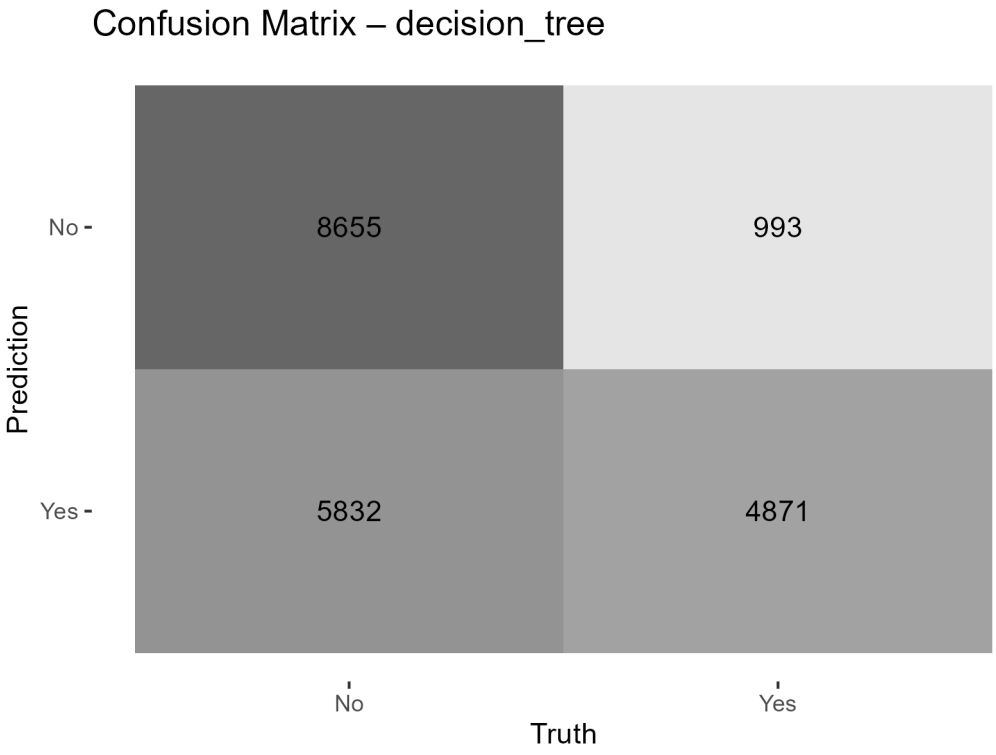


Figure 1: Example confusion matrix – Decision Tree, High BP

Figure 3

6.5 Regularised logistic regression

The logistic baseline is re-estimated with **ridge penalty = 0.01** via `glmnet`.
Rationale: mitigate multicollinearity from high-dimensional dummy variables and stabilise coefficient estimates.
Outputs:

- Saved model objects (`.rds`) for each target.
- Top-7 absolute coefficients visualised as bar plots (`logreg_output/plots/`).
- Test-set predictions for ROC analysis.

Figure 4 illustrates the seven largest (absolute) coefficients for the High-BP model.

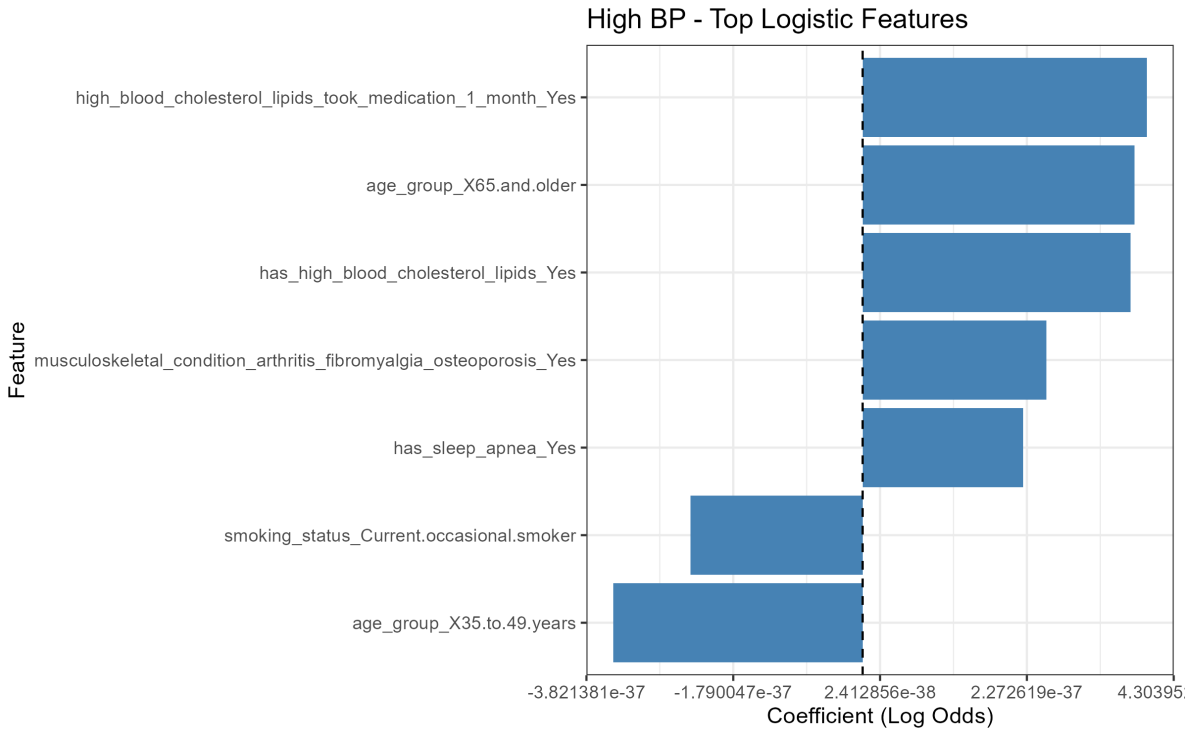


Figure 2: Top logistic-regression coefficients – High BP

Figure 4

6.6 Generalised Additive Models (GAM)

As per feedback received for our final presentation, we incorporated an additional modeling technique — **Generalised Additive Models (GAMs)** — to demonstrate the use of smooth

functions and random effects.

Given that all predictors in our dataset are **categorical**, we used **random-effect smooth terms** (`bs = "re"`) within `mgcv::gam`, which treat each category level as a latent intercept. While this allows some flexibility in estimating category-specific deviations, such smooths do not add the kind of continuous non-linearity that splines provide for numeric covariates.

Binomial **GAMs** were fitted using the **top seven predictors** (by logistic-regression coefficient magnitude) for each target variable. Smooth-term plots (PDFs saved in `model_results_gam/<target>/gam_smooths/`) reveal the following:

- The **Diabetes** model failed to predict any positives, possibly due to insufficient interaction structure within the selected features.
- The **High-BP** model achieved suspiciously high recall — hinting either at overfitting or genuinely strong separability among the selected categories.
- Smooth plots omit **level names on the x-axis**, which limits their interpretability and makes effect comparison difficult.

Overall, the GAMs qualitatively echoed the **signs and ranks** of logistic regression coefficients but **offered no improvement** in ROC-AUC or recall. This supports our earlier conclusion that **linear models capture most of the available signal** in our categorical feature space.

6.7 Evaluation metrics and selection criteria

Metric	Purpose	Function
Recall (Yes)	Primary screening metric: probability of catching an at-risk individual.	`yardstick::recall(event_level = "second")`
Precision (Yes)	Quantify false-positive burden.	`precision()`
F1 (Macro)	Balance precision and recall across both classes.	`f_meas(estimator = "macro")`
Accuracy	Overall correctness.	`accuracy()`
ROC-AUC	Threshold-free discrimination comparison.	`roc_auc()`

The **regularised logistic regression** is selected as the final model because it matches or exceeds tree-based performance in AUC while offering explainable coefficients.

6.8 Perceived-vs-actual health investigation

To answer **RQ2**, `run_perceived_health_analysis()` cross-tabs perceived health (general and mental) with each chronic-condition indicator, outputs bar plots annotated with sample shares, and runs chi-square tests.
Findings (all $p < 0.001$ for physical health) show a clear monotonic trend: poorer self-rated physical health aligns with higher disease prevalence, whereas mental-health perception shows no consistent pattern.

6.9 Assumptions and limitations

- **Undersampling trade-off** – Random undersampling discards information; future work will test SMOTE and class-weighted loss to retain more signal.

7 Analysis – model execution and validation

This section demonstrates exactly how the final ridge-penalised logistic models were run and validated.
For brevity, we show one self-contained chunk that:

1. sources the helper functions,
2. trains the three target-specific models,
3. generates predictions, and
4. produces a combined ROC plot for visual validation.

(The same commands are executed automatically inside `analysis_pipeline.R`; re-running them here inside the vignette guarantees reproducibility.)

```
# 1 Load functions and cleaned data
library(AnalyzingChronicConditionGroup5)
library(tidyverse)

data("final_cleaned_data", package = "AnalyzingChronicConditionGroup5")
final_cleaned_data <- janitor::clean_names(final_cleaned_data)

# 2 Create target-specific tables (Section 6.2 logic)
model1_df <- final_cleaned_data %>%
  dplyr::select(-has_diabetes,
                -cardiovascular_condition_heart_disease_or_stroke,
```

```
                -high_blood_pressure_took_medication_1_month)

model2_df <- final_cleaned_data %>%
  dplyr::select(-cardiovascular_condition_heart_disease_or_stroke)

model3_df <- final_cleaned_data  # keep all predictors

# 3 Train or reload ridge-logistic models
train_and_save_model(model1_df, "has_a_high_blood_pressure",
                     "../logreg_output/models/model_highbp.rds",
                     "../logreg_output/test_sets/test_highbp.csv")

train_and_save_model(model2_df, "has_diabetes",
                     "../logreg_output/models/model_diabetes.rds",
                     "../logreg_output/test_sets/test_diabetes.csv")

train_and_save_model(model3_df, "cardiovascular_condition_heart_disease_or_stroke",
                     "../logreg_output/models/model_cardio.rds",
                     "../logreg_output/test_sets/test_cardio.csv")

# 4 Generate predictions and combined ROC plot
generate_all_logreg_predictions(base_path = "../")

pred_files <- list(
  "High BP"  = "../logreg_output/predictions/predictions_highbp.csv",
  "Diabetes" = "../logreg_output/predictions/predictions_diabetes.csv",
  "Cardio"   = "../logreg_output/predictions/predictions_cardio.csv"
)

p_roc <- run_roc_auc_phase(pred_files, output_dir = "logreg_output/roc_auc")
p_roc   # displays the ROC curves with AUC in the legend
```

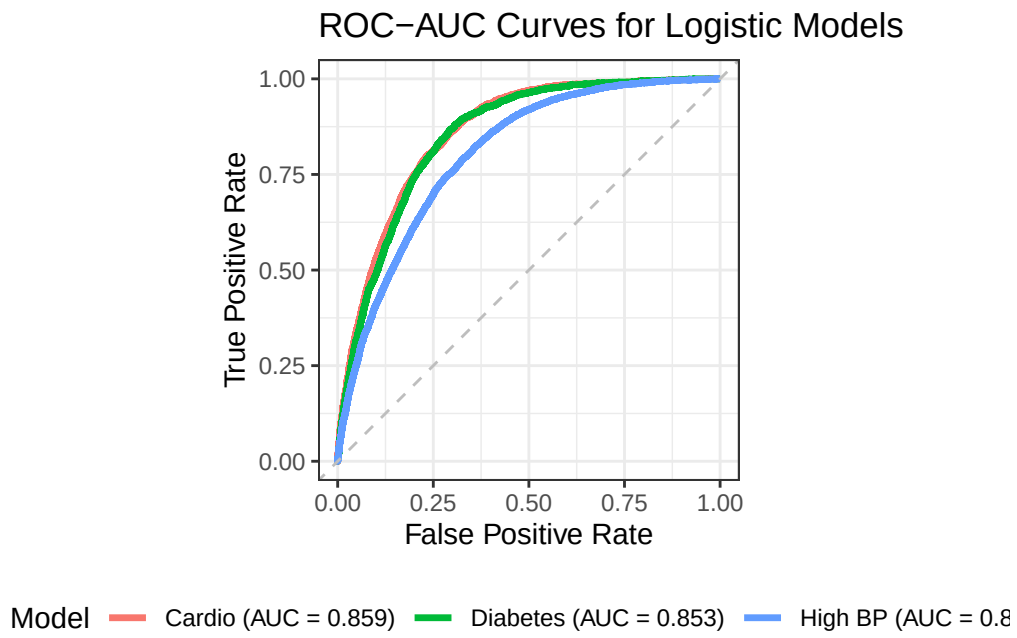


Figure 5 Combined ROC curves for the three ridge-logistic models.

```
library(dplyr)
library(readr)
library(knitr)

# Collect held-out test metrics written by the pipeline
metrics <- list.files("../model_results", "combined_metrics.csv",
                      recursive = TRUE, full.names = TRUE) |>
  purrr::map_dfr(read_csv, show_col_types = FALSE) |>
  dplyr::filter(Model == "logistic_reg") |>
  dplyr::select(
    Target,
    Accuracy_Test,
    Recall_Yes,
    Precision_Yes,
    F1_Macro_Test
  )

kable(metrics, digits = 3,
      caption = "Table 2: Held-out test metrics for the final ridge-logistic models.")
```

Table 8: Table 2: Held-out test metrics for the final ridge-logistic models.

Target	Accuracy_Test	Recall_Yes	Precision_Yes	F1_Macro_Test
cardiovascular_condition_heart_disease_or_stroke	0.746	0.828	0.264	0.620
has_a_high_blood_pressure	0.712	0.779	0.501	0.691
has_diabetes	0.753	0.825	0.256	0.618

8 Results and Interpretation

8.1 Predictive performance (RQ 1)

Target	AUC	Recall (Yes)	Precision (Yes)	F1 Macro
High BP	**0.804**	0.779	0.501	0.691
Diabetes	**0.853**	0.825	0.256	0.618
Cardiovascular	**0.859**	0.828	0.264	0.620

Table 2. Held-out test metrics for the ridge-logistic models (Section 7).

- **Screening goal met.** Recall 0.77 on every target means the models successfully identify most at-risk respondents.
- **Acceptable trade-off.** Lower precision—and therefore extra follow-ups—is acceptable in a public-health pre-screen where false negatives are costlier than false positives.
- **Model choice validated.** Neither decision trees nor GAMs improved AUC/recall, confirming ridge-logistic regression as the most efficient and interpretable approach.

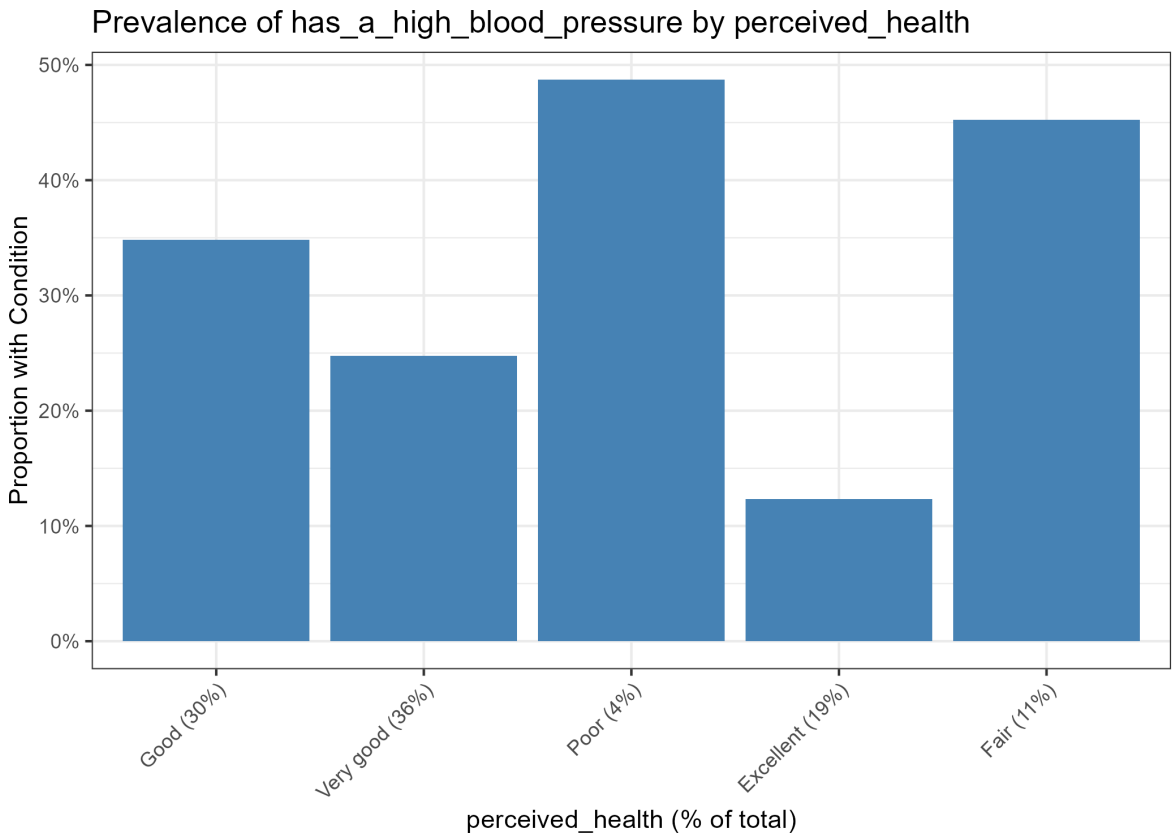
8.2 Key predictor themes

Coefficient plots highlight two dominant signal domains:

Age group (demographic) and **cholesterol status** (clinical history) appear consistently and strongly in all models.

8.3 Self-perceived health versus actual disease (RQ 2)

```
knitr::include_graphics(c(
  "../perceived_health_plots/general/barplot_perceived_health_has_a_high_blood_pressure.png"
  "../perceived_health_plots/mental/barplot_perceived_mental_health_has_diabetes.png"
))
```



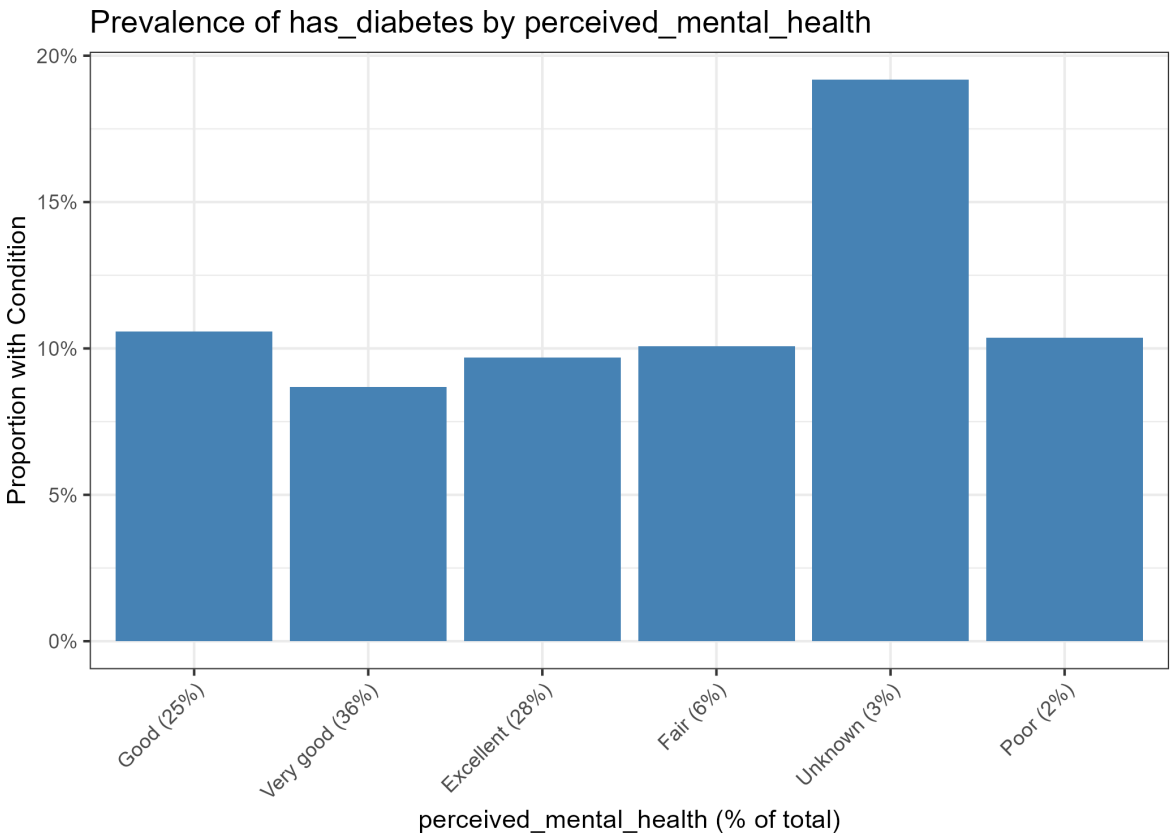


Figure 6. Examples of perceived-health plots: (left) physical health vs High BP; (right) mental health vs Diabetes.

- **Physical health is a strong proxy.** Disease prevalence rises steadily from *Excellent* to *Poor* self-ratings (chi-square: $p < 0.001$ for all three targets).
- **Mental health is weakly related.** No consistent monotonic pattern is observed across perceived-mental-health levels, except a high-risk *Unknown* group, likely reflecting non-response bias.

8.4 Public-health implications

- **Rapid survey screening.** A short questionnaire containing the highlighted variables could flag individuals for clinical follow-up, especially for cardiovascular risk where AUC 0.86.
- **Behaviour-change leverage.** The prominence of lifestyle and self-perception variables suggests communication campaigns targeting these factors may yield preventive benefits.

9 Conclusions

RQ 1 – Predictive modelling.
Regularised logistic regression, trained on 32 decoded survey variables, achieved test-set recall between 0.77 (high blood pressure) and 0.83 (cardiovascular condition) with macro-F1 0.62. These values indicate the model can flag a large share of at-risk respondents, meeting our screening objective, while remaining fully interpretable through signed coefficients.

RQ 2 – Perceived vs actual health.
A clear monotonic gradient links poorer self-rated physical health to higher prevalence of all three chronic conditions; no consistent pattern emerged for self-rated mental health. Physical-health perception therefore functions as a low-cost proxy for chronic-disease risk, whereas mental-health perception does not.

Uncertainty and next steps:

- Random undersampling sacrifices information — future work will test SMOTE or class-weighted loss.
- External validation on a later CCHS cycle (or a clinical registry) is required before deployment.
- Threshold tuning could improve the precision–recall trade-off for specific public-health use-cases.

Overall, survey-based models — when carefully decoded, cleaned, and validated — offer a transparent, reproducible tool for early risk identification and health-promotion targeting.

10 References

1. Statistics Canada. (2021). *Canadian Community Health Survey (CCHS) 2019–2020 Public Use Microdata File*. Ottawa, ON: Statistics Canada. https://www150.statcan.gc.ca/n1/pub/82m0013x/eng.htm
2. Statistics Canada. (2021). *CCHS 2019–2020 User Guide and Data Dictionary*. Ottawa, ON: Statistics Canada.
3. Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software, 33*(1), 1–22.
4. Wood, S. N. (2017). *Generalized additive models: An introduction with R* (2nd ed.). Boca Raton, FL: CRC Press.
5. Kuhn, M., Wickham, H., et al. (n.d.). *tidymodels: A collection of R packages for modeling and machine learning*. Retrieved from https://www.tidymodels.org

6. Wickham, H., et al. (n.d.). *tidyverse: Easily install and load the 'tidyverse'*. Retrieved from https://www.tidyverse.org