The Role of Social Support in Stroke Recovery: A Causal Inference Analysis Using Observational Rehabilitation Data

George Tzimas   Samiya Mohsin

May 2025

1   Introduction

Stroke is one of the leading causes of disability worldwide. In the United States, one in six deaths from cardiovascular disease are caused by stoke incidents [7]. There are various forms of clinical intervention that can help with stroke recovery, such as antiplatelet and anticoagulation therapy, mechanical thrombolysis, among others [12]. While clinical interventions have had a documented history of effective recovery from stroke, social determinants of health such as social support have also been shown to have a positive effect on recovery and rehabilitation outcomes [1].

In this study, we investigate the effect of perceived social support on post-stroke recovery. Using data form a large, observational dataset of U.S. stroke rehabilitation patients [11], we apply causal inference methods to estimate the effect of social support on functional outcomes post-stroke. We then analyze heterogeneity in treatment effects (HTE) through the use of causal forest modeling to assess whether some subgroups benefit more from social support than others. Aligned with the CDC's Health Impact in 5 Years initiative [6], we aim to identify patient populations that may benefit the most from early social support interventions.

1

2 Literature Review

Previous research has shown that social support can positively influence stroke recovery outcomes. Tsouna-Hadjis et al. (2000) found that higher perceived social support was associated with improved functional status and lower levels of depression in stroke patients over a one-year follow-up period [13]. Similarly, Glass et al. (1993) explored the impact of social networks and family involvement on stroke rehabilitation outcomes. Their study showed that patients with closer family involvement and larger social networks made faster and more extensive functional gains during recovery and that those that were isolated may be at a greater risk for poor outcomes [9]. A more recent systematic review of the relationship between social support and stroke by Elloker and Rhoda (2018) further highlights the importance of family and community-based support networks. Their study found that having high levels of social support had a positive influence on participation, social and leisure activities, as well as returning to work post-stroke [8]. However, it is important to keep in mind that social support is not only important in terms of the benefits it offers in regards to stroke recovery, but also for the negative consequence that can arise in its absence. Another systematic review by by Northcott et al. (2016), which included 70 studies and over 4,800 participants, found that the social networks of stroke survivors tend to typically become smaller, especially when it comes to non-kin relationships. This drop in social support was significantly associated with increased levels of depression and a lower quality of life [10]. Being able to provide early and targeted social support interventions could help alleviate a lot of the psychological distress associated with post-stroke recovery.

3 Methods

3.1 Data Overview

We used data from the *Stroke Recovery in Underserved Populations 2005-2006* study, which included 1,219 stroke rehabilitation patients from multiple U.S. facilities. The dataset contains a total of 226 columns that capture demographic characteristics, patient comorbidities, and other recovery-related outcomes measurements.

3.2 Outcome Definition

The primary outcome for our analysis is change in total Functional Independence Measure (FIM) score, which was calculated as the difference between the FIM score at 3 months (and 12 months, for sensitivity analysis) and at the time of admission.

3.3 Exposure Definition

The treatment variable was a binary indicator derived from the Duke–UNC Functional Social Support Questionnaire (FSSQ) scores, which measure perceived functional social support across emotional and instrumental domains [5]. Participants were classified based on a predefined score threshold of 50, with those equal and above that score being categorized as having "high" social support and those below with "low" social support.

3.4 Predictor Measures

A total of 22 baseline covariates were selected and used to adjust for potential confounding in all regression and causal forest models:

- **Demographics:** Age (binned into categories), gender, race/ethnicity, marital status, and level of education.

- **Comorbidity Indicators:** Presence of diabetes, hypertension, heart disease, obesity, cancer, mental health disorders, respiratory issues, arthritis, kidney disease, other circulatory issues, and the total comorbidity score.

- **Stroke Severity:** Length of stay is used as a proxy measure of stroke severity.

- **Baseline Functional Status:** Total functional independence measured at admission time only.

A detailed comparison of all predictors stratified by exposure arm is available in Table A.1.

3.5 Modeling

3.5.1 Propensity score matching

To estimate the causal effect of high social support on functional recovery, we used an Inverse Probability Weighting (IPW) approach. We did this by fitting a logistic regression model to predict the probability of receiving high social support (DUKE_F $>= 50$) as a function of the predictor variables.

$$\hat{e}(X_i) = P(T_i = 1 | X_i) \tag{1}$$

The propensity scores (weights) were then used to derive the inverse probability weights that balance the distribution of observed covariates between the two treatment arms.

$$w_i = \begin{cases} \frac{1}{\hat{e}(X_i)} & \text{if } T_i = 1 \quad \text{(high support)} \\ \frac{1}{1-\hat{e}(X_i)} & \text{if } T_i = 0 \quad \text{(low support)} \end{cases} \tag{2}$$

3.5.2 Survey design and Logistic regression

Next, we created a survey-weighted design object to incorporate the estimated propensity score weights into our analysis and ensure stable modeling statistical estimates. We then fit a generalized linear model function using the survey design object to regress the outcome of interest (change in FIM score) on the treatment indicator (high vs. low social support) and all relevant covariates. This modeling step allows us to estimate the average treatment effect (ATE) after adjusting for any residual confounding through the propensity score weighing.

3.5.3 Causal Forest

We also implemented a Causal Forest to estimate potential presence of heterogeneity in treatment effect (HTE) [2]. Causal Forest is an extension of the traditional Random Forest [4] modeling approach, with the biggest difference being that it is designed to estimate the Conditional Average Treatment Effect (CATE) for each of the participants by fitting weighted differences in outcomes between treated and control individuals in a non-parametric manner. This allows for the estimation of personalized treatment effects by estimating the difference between the observed outcome under one arm and the counterfactual outcome which was not observed.
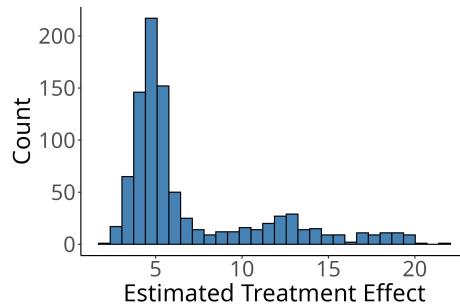
$$\tau(X_i) = \mathbb{E}[Y_i \mid T_i = 1, X_i] - \mathbb{E}[Y_i \mid T_i = 0, X_i], \qquad (3)$$

Additionally, we examined variable importance metrics from the forest to identify which baseline characteristics most strongly predicted variation in treatment effect.
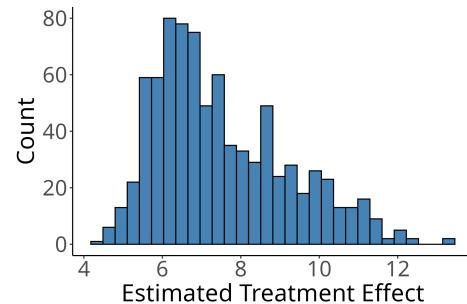
4 Results

The IPW-adjusted regression model estimated that patients with high social support experienced a 6.2-point greater improvement in total FIM score at 3 months (p <0.001), adjusting for all other covariates. This effect remained significant at 12 months with a 7.6-point greater improvement in total FIM score. Model coefficients for both models can be found in Table B.1.

Causal Forest analysis confirmed the presence of substantial HTE in terms of FIM score at three months. Calibration testing using a linear model regressing observed outcomes on the forest-predicted treatment effects showed a statistically significant differential effect term (p <0.05), supporting the presence of individualized treatment variation (Table C.1). However, the evidence at the one-year follow-up was not statistically significant (p =0.831), suggesting weaker evidence for heterogeneity. This difference in HTE outcomes between the three-month and one-year follow-up is likely due to loss of follow-up from the patients (99 less patients at the one-year time point).

(a) Three-month follow-up   (b) One-year follow-up.

Figure 1: Distribution of CATE estimates.

Focusing on the three-month time point, we conducted variable importance to identify what variables drive heterogeneity in outcomes. The top

driver for heterogeneity in FIM score gain from the start of the study up to the three-month time point was the baseline FIM score at admission time(Figure 2).
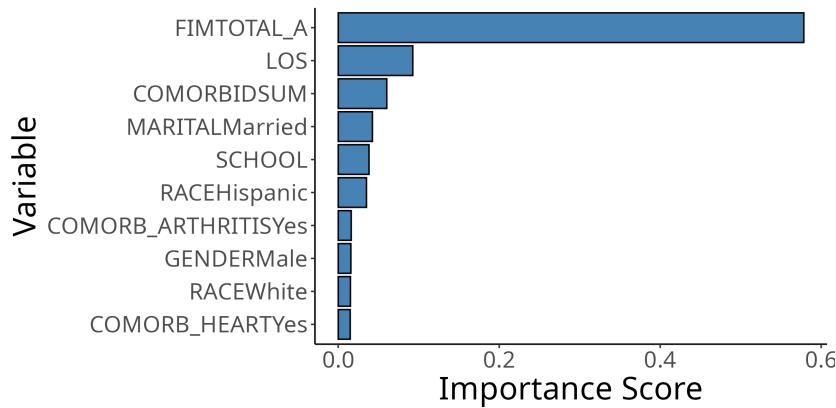
Figure 2: Top variables that drive treatment effect heterogeneity for FIM score gain at a three-month follow-up.

To further explore heterogeneity, we stratified participants by quartiles of baseline FIM score and found that the lowest-functioning quartile had a mean CATE of 13.6 points, substantially higher than the top quartile, which averaged 4.5 points (Table 1). To formally test whether these differences in CATE across baseline FIM quartiles was statistically significant, we used a Kruskal-Wallis rank sum test. The result showed a highly significant difference in CATEs across quartiles ($p < 0.05$), confirming that patients with lower baseline function experienced significantly greater gains from high social support (Table 2).

Table 1: Mean estimated CATE and standard deviation by baseline FIM quartile.

fim_quartile	mean_cate	sd_cate	n
Q1 (lowest)	13.484606	3.2270675	234
Q2	5.239652	1.3854878	247
Q3	4.755871	0.7334782	216
Q4 (highest)	4.523827	0.7499607	231
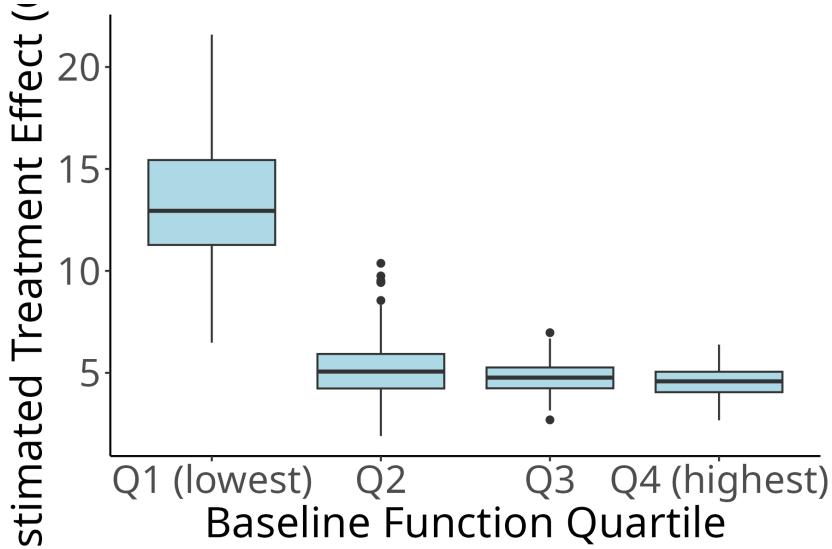
Figure 3: Estimated treatment effects stratified by baseline FIM quartiles.

Table 2: Kruskal–Wallis test comparing CATE distributions across baseline FIM quartiles.

Test	Statistic	Degrees of Freedom	p-value
565.1668	<1e-04	3	Kruskal-Wallis rank sum test

The results show that patients who began rehabilitation with lower functional capacity (lower baseline FIM score) gained the most from increased social support. This suggests that social support interventions may offer the greatest marginal utility for individuals at highest risk of prolonged dependence. Individuals that are move functionally impaired may face greater physical, emotional, and logistical barriers when it comes to recovering, and thus may benefit disproportionately from social support. In contrast, patients who are already functionally independent at baseline are likely to have less room for measurable improvement, resulting in smaller observed treatment effects.

5   Discussion

These findings support the hypothesis that social support plays a crucial role in stroke recovery, especially for patients with greater functional impairment at baseline. The presence of strong HTE suggests that interventions should be tailored towards patients entering rehab with lower independence, as they may derive the greatest benefit from enhanced social support pathways. Based on these results, we propose a risk-stratified intervention aligned with the CDC's HI-5 framework which would involve screening all patients on admission for social support and functional status, and targeting resources such as peer support, caregiver education, and social work referrals to those with the lowest FIM scores.

5.1   Proposed Intervention

Based on these findings, we propose intervention strategy that targets stroke survivors with low functional independence during admission time. The primary target of such interventions would be stroke patients with low baseline social support, with secondary targets being clinical staff at hospitals and care centers. This strategy is based on the CDC's "Three Buckets of Prevention" framework [3], which combines traditional care, innovative outreach, and population-level initiatives, and can be tailored towards the promotion of equitable stroke recovery outcomes:

1. **Bucket 1:** Traditional clinical preventive intervention

   - **Admission Screening:** When a patient is admitted to a hospital/care center, they will be screened using the Duke-UNC Functional Social Support Questionnaire and baseline FIM scoring to identify individuals with low functional status and limited support.

   - **Clinical Workflow Integration:** Those identified as high-risk patients will be flagged in the EMR and prioritized for social and peer support referrals.

2. **Bucket 2:** Preventive interventions that go beyond the clinic

   - **Caregiver and Peer Support Training:** Partner with community organizations to train peer mentors and caregivers that can

continue providing support to individuals through services such as home visits or virtual sessions after patients are discharged.

- **Digital Recovery App** Build a web/mobile application for patients with low functional independence that integrates recovery tracking, resource linking, and motivational support content.

- **Home-based Outreach:** For patients that are at the highest risk levels (e.g., lowest FIM quartile), create a targeted program that offers weekly/biweekly in-home or telehealth support check-ins by community health workers or other specialists.

3. **Bucket 3:** Community-wide interventions

- **Raising Awareness:** Create public health campaigns that explain the role of social support in stroke recovery for the general population, as well as tailored materials for caregivers, stroke survivors, and community leaders.

- **Public Health Partnerships:** Collaborate with local health departments and community leaders to incorporate social support tracking into ongoing community health needs assessments.

This comprehensive approach would allow us to offer scalable, equity-oriented care solutions to patients that can support their recovery progress from stroke incidents. It would ensure that the people with the most to gain from social support are prioritized, and that the health system can sustain and learn from the intervention over time.

6   Conclusion

Using both traditional survey-weighted regression and causal inference methods on real-world rehabilitation data, we demonstrate that high social support significantly improves stroke recovery, with the greatest benefit observed among the most functionally impaired patients. IPW-adjusted GLM models provided robust average treatment effect estimates, while causal forest modeling allowed for individualized benefit predictions, revealing substantial heterogeneity in treatment effects. These findings support the use of stratified care models for stroke rehabilitation, where social support interventions can be targeted to those most that are most likely to benefit. Early intervention,

informed by predictive analytics, can offer a promising path toward recovery. Collectively, this multi-method approach reinforces the public health case for scalable, data-driven programs that reduce disparities and improve outcomes.

A   Appendix A: Summary Table of Predictor Variables by Social Support Category

Table A.1: Participant characteristics stratified by high social support (low and high).

	Level	Low	High	p-value
Sample Size (n)		246	682	
Sex (%)	Female	119 (48.4)	355 (52.1)	0.360
	Male	127 (51.6)	327 (47.9)	
Race/Ethnicity (%)	Black	45 (18.3)	124 (18.2)	0.499
	Hispanic	19 (7.7)	35 (5.1)	
	Other	5 (2.0)	17 (2.5)	
	White	177 (72.0)	506 (74.2)	
Age Group (%)	≤54	55 (22.4)	115 (16.9)	0.092
	55–64	53 (21.5)	128 (18.8)	
	65–74	49 (19.9)	176 (25.8)	
	75–84	72 (29.3)	196 (28.7)	
	85+	17 (6.9)	67 (9.8)	
Years of Education (mean (SD))		12.36 (3.12)	12.36 (2.98)	1.000
Marital Status (%)	Divorced	39 (15.9)	149 (21.8)	0.231
	Married	136 (55.3)	356 (52.2)	
	Never married	49 (19.9)	127 (18.6)	
	Widowed/Separated	22 (8.9)	50 (7.3)	
Baseline FIM Score (mean (SD))		55.09 (18.31)	57.90 (19.55)	0.049
High Social Support (%)	0	246 (100.0)	0 (0.0)	<0.001
	1	0 (0.0)	682 (100.0)	
Total FIM Gain (mean (SD))		43.51 (20.61)	49.02 (18.72)	<0.001
Motor FIM Gain (mean (SD))		34.61 (16.68)	37.53 (15.20)	0.012

	Level	Low	High	p-value
Cognitive FIM Gain (mean (SD))		8.91 (7.69)	11.49 (7.77)	<0.001
Length of Stay (mean (SD))		21.32 (11.58)	19.72 (11.20)	0.056
Diabetes (%)	No	151 (61.4)	452 (66.3)	0.193
	Yes	95 (38.6)	230 (33.7)	
Hypertension (%)	No	69 (28.0)	175 (25.7)	0.519
	Yes	177 (72.0)	507 (74.3)	
Heart Disease (%)	No	139 (56.5)	356 (52.2)	0.278
	Yes	107 (43.5)	326 (47.8)	
Obesity (%)	No	224 (91.1)	637 (93.4)	0.283
	Yes	22 (8.9)	45 (6.6)	
Cancer (%)	No	237 (96.3)	653 (95.7)	0.830
	Yes	9 (3.7)	29 (4.3)	
Mental Health Condition (%)	No	116 (47.2)	356 (52.2)	0.200
	Yes	130 (52.8)	326 (47.8)	
Respiratory Disease (%)	No	199 (80.9)	562 (82.4)	0.666
	Yes	47 (19.1)	120 (17.6)	
Arthritis (%)	No	209 (85.0)	598 (87.7)	0.328
	Yes	37 (15.0)	84 (12.3)	
Kidney Disease (%)	No	228 (92.7)	646 (94.7)	0.312
	Yes	18 (7.3)	36 (5.3)	
Other Circulatory Disease (%)	No	188 (76.4)	509 (74.6)	0.638
	Yes	58 (23.6)	173 (25.4)	
Fractures (%)	No	244 (99.2)	676 (99.1)	1.000
	Yes	2 (0.8)	6 (0.9)	
Comorbidity Count (mean (SD))		2.85 (1.32)	2.76 (1.25)	0.320

B Appendix B: IPW-Adjusted Regression Model Estimates

Table B.1: Survey-weighted linear regression estimating the effect of high support on FIM total score change at 3 months and 1 year.

Characteristic	3-Month Outcome Beta	3-Month Outcome SE	12-Month Outcome Beta	12-Month Outcome SE
(Intercept)	63***	6.32	59***	6.31
High Support				
0	—	—	—	—
1	6.2***	1.34	7.6***	1.51
AGE_CAT				
¡=54	—	—	—	—
55-64	-2.9	2.02	-0.14	1.84
65-74	-6.7**	2.23	-3.7	2.04
75-84	-9.6***	2.25	-8.5***	2.39
85+	-7.5*	3.42	-6.8*	3.00
GENDER				
Female	—	—	—	—
Male	1.9	1.46	1.3	1.57
RACE				
Black	—	—	—	—
Hispanic	-4.3	3.13	-8.5	4.33
Other	-5.0	6.73	5.1	3.80
White	3.4	1.78	3.0	1.99
SCHOOL	0.22	0.236	0.55*	0.230
MARITAL				
Divorced	—	—	—	—
Married	-4.2*	2.03	0.30	2.49
Never married	-0.18	2.30	3.5	2.50
Widowed/Separated	-1.2	2.75	3.8	2.55
FIMTOTAL_A	-0.28***	0.051	-0.36***	0.060
LOS	0.18*	0.090	0.22*	0.090
COMORB_DIABETES				
No	—	—	—	—
Yes	-3.6*	1.46	-3.5*	1.68
COMORB_HTN				
No	—	—	—	—
Yes	-0.94	1.62	-1.0	1.57
COMORB_HEART				
No	—	—	—	—
Yes	-2.4	1.41	-3.1	1.64
COMORB_OBESITY				
No	—	—	—	—
Yes	-1.3	1.98	-0.40	2.04
COMORB_CANCER				
No	—	—	—	—
Yes	-3.4	3.75	-8.7	5.29

COMORB_MENTAL				
No	—	—	—	—
Yes	-1.9	1.34	-1.2	1.57
COMORB_RESPIRATORY				
No	—	—	—	—
Yes	-0.18	1.69	0.82	1.92
COMORB_ARTHRITIS				
No	—	—	—	—
Yes	-3.5	2.10	-2.7	2.20
COMORB_KIDNEY				
No	—	—	—	—
Yes	-5.0*	2.47	-4.4	2.78
COMORB_OTHER_CIRC				
No	—	—	—	—
Yes	-0.63	1.54	0.00	1.96
COMORB_FRACTURES				
No	—	—	—	—
Yes	8.8	8.32	2.2	4.82

Abbreviations: CI = Confidence Interval, SE = Standard Error

[1] *p<0.05; **p<0.01; ***p<0.001

C   Appendix C: Causal Forest

Table C.1: Calibration test results for the Causal Forest model at a three-month follow-up.

Characteristic	Beta	95% CI	p-value
mean.forest.prediction	0.99	0.62, 1.4	<0.001
differential.forest.prediction	1.2	0.37, 1.9	0.002

Abbreviation: CI = Confidence Interval

D   Appendix D: R Code

```
## ----setup,
    include=FALSE-----------------------------------------------------------------
knitr::opts_chunk$set(echo = TRUE)
library(tidyverse)
library(ggplot2)
library(tableone)
library(kableExtra)
library(htmlTable)
library(gtsummary)
library(WeightIt)
```

```r
library(survey)
library(flextable)
library(grf)
library(clipr)


## -----------------------------------------------------------------------------
theme_custom <- function(base_size = 14, axis_title_size = 16,
    axis_text_size = 14) {
  theme(
    axis.title.x = element_text(size = axis_title_size),
    axis.title.y = element_text(size = axis_title_size),
    axis.text.x  = element_text(size = axis_text_size),
    axis.text.y  = element_text(size = axis_text_size),
    axis.text
  )
}


## -----------------------------------------------------------------------------
data <- read_tsv('data/36422-0001-Data.tsv')
cat('Dim:', dim(data), '\n')
head(data)


## -----------------------------------------------------------------------------
df <- data %>%
  rename(
    'GENDER' = FEMALE
  ) %>%
  mutate(
    Facility = factor(
      case_when(
        NEWFAC_NUM == 1 ~ 'New Jersey',
        NEWFAC_NUM == 2 ~ 'New York',
        NEWFAC_NUM == 3 ~ 'Iowa',
        NEWFAC_NUM == 4 ~ 'California',
        NEWFAC_NUM == 5 ~ 'Illinois',
        NEWFAC_NUM == 6 ~ 'Texas',
        NEWFAC_NUM == 7 ~ 'Washington',
        NEWFAC_NUM == 8 ~ 'New York',
        NEWFAC_NUM == 9 ~ 'Texas',
```

```
      NEWFAC_NUM == 10 ~ 'Kentucky',
      NEWFAC_NUM == 11 ~ 'Florida'
    )
  ),

  AGE_CAT = factor(
    case_when(
      AGE_CAT == 1 ~ '<=54',
      AGE_CAT == 2 ~ '55-64',
      AGE_CAT == 3 ~ '65-74',
      AGE_CAT == 4 ~ '75-84',
      AGE_CAT == 5 ~ '85+'
    ), levels=c('<=54', '55-64', '65-74', '75-84', '85+')
  ),

  GENDER = factor(
    case_when(
      GENDER == 0 ~ 'Male',
      GENDER == 1 ~ 'Female'
    )
  ),

  RACE = factor(
    case_when(
      RACE == 1 ~ 'White',
      RACE == 2 ~ 'Black',
      RACE == 3 ~ 'Hispanic',
      RACE == 4 ~ 'Other'
    )
  ),

  MARITAL = factor(
    case_when(
      MARITAL == 1 ~ 'Never married',
      MARITAL == 2 ~ 'Married',
      MARITAL == 3 ~ 'Divorced',
      MARITAL == 4 ~ 'Widowed/Separated'
    )
  ),


  ADMITFROM = factor(
    case_when(
      ADMITFROM == 1 ~ 'Home',
      ADMITFROM == 2 ~ 'Skilled nursing facility',
```

```r
      ADMITFROM == 3 ~ 'Acute',
      ADMITFROM == 4 ~ 'Other medical'
    )
  ),

  LIVESETTING_A = factor(
    case_when(
      LIVESETTING_A == 1 ~ 'Home',
      LIVESETTING_A == 2 ~ 'Other'
    )
  ),

  LIVEWITH_A = factor(
    case_when(
      LIVEWITH_A == 1 ~ 'Alone',
      LIVEWITH_A == 2 ~ 'Family',
      LIVEWITH_A == 3 ~ 'Other'
    )
  ),

  VOCCATEGORY_A = factor(
    case_when(
      VOCCATEGORY_A == 1 ~ 'Employed',
      VOCCATEGORY_A == 2 ~ 'Sheltered',
      VOCCATEGORY_A == 3 ~ 'Student',
      VOCCATEGORY_A == 4 ~ 'Homemaker',
      VOCCATEGORY_A == 5 ~ 'Not working',
      VOCCATEGORY_A == 6 ~ 'Retired (Age)',
      VOCCATEGORY_A == 7 ~ 'Retired (Disability)'
    )
  ),

  VOCEFFORT_A = factor(
    case_when(
      VOCEFFORT_A == 1 ~ 'Full-time',
      VOCEFFORT_A == 2 ~ 'Part-time',
      VOCEFFORT_A == 3 ~ 'Adjusted workload'
    )
  ),

  PAYPRIME = factor(
    case_when(
      PAYPRIME == 1 ~ 'Blue Cross',
      PAYPRIME == 2 ~ 'Medicare',
      PAYPRIME == 3 ~ 'Medicaid',
```

```
      PAYPRIME == 4 ~ 'Commercial insurance',
      PAYPRIME == 5 ~ 'Managed Care',
      PAYPRIME == 6 ~ 'Other',
      PAYPRIME == 7 ~ 'None'
    )
),

PAYSECOND = factor(
  case_when(
    PAYSECOND == 1 ~ 'Blue Cross',
    PAYSECOND == 2 ~ 'Medicare',
    PAYSECOND == 3 ~ 'Medicaid',
    PAYSECOND == 4 ~ 'Commercial insurance',
    PAYSECOND == 5 ~ 'Managed Care',
    PAYSECOND == 6 ~ 'Other',
    PAYSECOND == 7 ~ 'None'
  )
),

DIAG_MAIN = factor(
  case_when(
    DIAG_MAIN == 1 ~ 'Hemorrhagic stroke',
    DIAG_MAIN == 2 ~ 'Ischemic stroke',
    DIAG_MAIN == 3 ~ 'Other stroke',
    DIAG_MAIN == 4 ~ 'Other circulatory disease',
    DIAG_MAIN == 5 ~ 'Other medical condition'
  )
),

DIAG_IG = factor(
  case_when(
    DIAG_IG == 1 ~ 'Left-body stroke',
    DIAG_IG == 2 ~ 'Right-body stroke',
    DIAG_IG == 3 ~ 'Bilateral stroke',
    DIAG_IG == 4 ~ 'No paresis stroke',
    DIAG_IG == 5 ~ 'Other stroke'
  )
),

LIVESETTING_D = factor(
  case_when(
    LIVESETTING_D == 1 ~ 'Home',
    LIVESETTING_D == 2 ~ 'Skilled nursing facility',
    LIVESETTING_D == 3 ~ 'Acute',
    LIVESETTING_D == 4 ~ 'Other'
```

```r
    )
  ),

  LIVEWITH_D = factor(
    case_when(
      LIVEWITH_D == 1 ~ 'Alone',
      LIVEWITH_D == 2 ~ 'Family',
      LIVEWITH_D == 3 ~ 'Friends',
      LIVEWITH_D == 4 ~ 'Attendant',
      LIVEWITH_D == 5 ~ 'Other'
    )
  ),

  DISTANCE_WALKED_A = factor(
    case_when(
      DISTANCE_WALKED_A == 0 ~ 'Activity does not occur',
      DISTANCE_WALKED_A == 1 ~ '<50 ft',
      DISTANCE_WALKED_A == 2 ~'50-149 ft',
      DISTANCE_WALKED_A == 3 ~ '150+ ft'
    ), levels=c('Activity does not occur', '<50 ft', '50-149 ft', '150+
ft'), ordered=T
  ),

  DISTANCE_WALKED_D = factor(
    case_when(
      DISTANCE_WALKED_D == 0 ~ 'Activity does not occur',
      DISTANCE_WALKED_D == 1 ~ '<50 ft',
      DISTANCE_WALKED_D == 2 ~'50-149 ft',
      DISTANCE_WALKED_D == 3 ~ '150+ ft'
    ), levels=c('Activity does not occur', '<50 ft', '50-149 ft', '150+
ft'), ordered=T
  ),

  LIVESETTING_F = factor(
    case_when(
      LIVESETTING_F == 1 ~ 'Home',
      LIVESETTING_F == 2 ~ 'Skilled nursing facility',
      LIVESETTING_F == 3 ~ 'Acute',
      LIVESETTING_F == 4 ~ 'Intermediate Care',
      LIVESETTING_F == 5 ~ 'Skilled Nursing Facility',
      LIVESETTING_F == 8 ~ 'Chronic Hospital',
      LIVESETTING_F == 9 ~ 'Rehabilitation Facility',
      LIVESETTING_F == 10 ~ 'Other',
      LIVESETTING_F == 11 ~ 'Undocument',
      LIVESETTING_F == 12 ~ 'Alternate Level of Care Unit',
```

```
        LIVESETTING_F == 13 ~ 'Subacute Setting'
    )
),

LIVEWITH_F = factor(
    case_when(
        LIVEWITH_F == 1 ~ 'Alone',
        LIVEWITH_F == 2 ~ 'Family',
        LIVEWITH_F == 3 ~ 'Friends',
        LIVEWITH_F == 4 ~ 'Attendant',
        LIVEWITH_F == 5 ~ 'Other'
    )
),

VOCEFFORT_F = factor(
    case_when(
        VOCEFFORT_F == 1 ~ 'Full-time',
        VOCEFFORT_F == 2 ~ 'Part-time',
        VOCEFFORT_F == 3 ~ 'Adjusted workload'
    )
),

VOCCATEGORY_F = factor(
    case_when(
        VOCCATEGORY_F == 1 ~ 'Employed',
        VOCCATEGORY_F == 2 ~ 'Sheltered',
        VOCCATEGORY_F == 3 ~ 'Student',
        VOCCATEGORY_F == 4 ~ 'Homemaker',
        VOCCATEGORY_F == 5 ~ 'Not working',
        VOCCATEGORY_F == 6 ~ 'Retired (Age)',
        VOCCATEGORY_F == 7 ~ 'Retired (Disability)'
    )
),

HLTHMAINTPRIME_F = factor(
    case_when(
        HLTHMAINTPRIME_F == 1 ~ 'Own care',
        HLTHMAINTPRIME_F == 2 ~ 'Unpaid person/family',
        HLTHMAINTPRIME_F == 3 ~ 'Paid person'
    )
),

HLTHMAINTSECOND_F = factor(
    case_when(
        HLTHMAINTSECOND_F == 1 ~ 'Own care',
```

```r
      HLTHMAINTSECOND_F == 2 ~ 'Unpaid person/family',
      HLTHMAINTSECOND_F == 3 ~ 'Paid person'
    )
  ),

  POSTDCCARE_NOW_F = factor(
    case_when(
      POSTDCCARE_NOW_F ==  1 ~ 'None',
      POSTDCCARE_NOW_F == 2 ~ 'Outpatient',
      POSTDCCARE_NOW_F == 3 ~ 'Home based',
      POSTDCCARE_NOW_F == 4 ~ 'Outpatient + Home based',
      POSTDCCARE_NOW_F == 5 ~ 'Inpatient hospital',
      POSTDCCARE_NOW_F == 6 ~ 'Long-term care'
    )
  ),

  POSTDCCARE_ANY_F = factor(
    case_when(
      POSTDCCARE_ANY_F ==  1 ~ 'None',
      POSTDCCARE_ANY_F == 2 ~ 'Outpatient',
      POSTDCCARE_ANY_F == 3 ~ 'Home based',
      POSTDCCARE_ANY_F == 4 ~ 'Outpatient + Home based',
      POSTDCCARE_ANY_F == 5 ~ 'Inpatient hospital',
      POSTDCCARE_ANY_F == 6 ~ 'Long-term care'
    )
  ),

  POSTDC_HOSP = factor(
    case_when(
      POSTDC_HOSP == 1 ~ 'None',
      POSTDC_HOSP == 2 ~ 'Med/surgery',
      POSTDC_HOSP == 3 ~ 'Rehab',
      POSTDC_HOSP == 4 ~ 'Both'
    )
  ),

  SATISOVERALL_F = factor(
    case_when(
      SATISOVERALL_F == 1 ~ 'Very dissatisfied',
      SATISOVERALL_F == 2 ~ 'Somewhat dissatisfied',
      SATISOVERALL_F == 3 ~ 'Somewhat satisfied',
      SATISOVERALL_F == 4 ~ 'Very satisfied'
    ), levels=c('Very dissatisfied', 'Somewhat dissatisfied', 'Somewhat
satisfied', 'Very satisfied'), ordered=T
  ),
```

```r
SATISQOL_F = factor(
  case_when(
    SATISQOL_F == 1 ~ 'Very dissatisfied',
    SATISQOL_F == 2 ~ 'Somewhat dissatisfied',
    SATISQOL_F == 3 ~ 'Somewhat satisfied',
    SATISQOL_F == 4 ~ 'Very satisfied'
  ), levels=c('Very dissatisfied', 'Somewhat dissatisfied', 'Somewhat
satisfied', 'Very satisfied'), ordered=T
),

SATISGOAL_F = factor(
  case_when(
    SATISGOAL_F == 1 ~ 'Very dissatisfied',
    SATISGOAL_F == 2 ~ 'Somewhat dissatisfied',
    SATISGOAL_F == 3 ~ 'Somewhat satisfied',
    SATISGOAL_F == 4 ~ 'Very satisfied'
  ), levels=c('Very dissatisfied', 'Somewhat dissatisfied', 'Somewhat
satisfied', 'Very satisfied'), ordered=T
),

SATISCOMPAR_F = factor(
  case_when(
    SATISCOMPAR_F == 1 ~ 'Very dissatisfied',
    SATISCOMPAR_F == 2 ~ 'Somewhat dissatisfied',
    SATISCOMPAR_F == 3 ~ 'Somewhat satisfied',
    SATISCOMPAR_F == 4 ~ 'Very satisfied'
  ), levels=c('Very dissatisfied', 'Somewhat dissatisfied', 'Somewhat
satisfied', 'Very satisfied'), ordered=T
),

DISTANCE_WALKED_F = factor(
  case_when(
    DISTANCE_WALKED_F == 0 ~ 'Activity does not occur',
    DISTANCE_WALKED_F == 1 ~ '<50 ft',
    DISTANCE_WALKED_F == 2 ~'50-149 ft',
    DISTANCE_WALKED_F == 3 ~ '150+ ft'
  ), levels=c('Activity does not occur', '<50 ft', '50-149 ft', '150+
ft'), ordered=T
),

DISTANCE_WALKED_Y = factor(
  case_when(
    DISTANCE_WALKED_Y == 0 ~ 'Activity does not occur',
    DISTANCE_WALKED_Y == 1 ~ '<50 ft',
```

```r
    DISTANCE_WALKED_Y == 2 ~'50-149 ft',
    DISTANCE_WALKED_Y == 3 ~ '150+ ft'
  ), levels=c('Activity does not occur', '<50 ft', '50-149 ft', '150+
ft'), ordered=T
),

SOCIALSUPP1_D = factor(
  case_when(
    SOCIALSUPP1_D == 1 ~ 'Most of the time',
    SOCIALSUPP1_D == 2 ~ 'Some of the time',
    SOCIALSUPP1_D == 3 ~ 'Hardly ever',
    SOCIALSUPP1_D == 4 ~ 'Don\'t know/refused'
  ), levels=c('Most of the time','Some of the time','Hardly
ever','Don\'t know/refused'), ordered=T
),

SOCIALSUPP2_D = factor(
  case_when(
    SOCIALSUPP2_D == 1 ~ 'Most of the time',
    SOCIALSUPP2_D == 2 ~ 'Some of the time',
    SOCIALSUPP2_D == 3 ~ 'Hardly ever',
    SOCIALSUPP2_D == 4 ~ 'Don\'t know/refused'
  ), levels=c('Most of the time','Some of the time','Hardly
ever','Don\'t know/refused'), ordered=T
),

SOCIALSUPP1_F = factor(
  case_when(
    SOCIALSUPP1_F == 1 ~ 'Most of the time',
    SOCIALSUPP1_F == 2 ~ 'Some of the time',
    SOCIALSUPP1_F == 3 ~ 'Hardly ever',
    SOCIALSUPP1_F == 4 ~ 'Don\'t know/refused'
  ), levels=c('Most of the time','Some of the time','Hardly
ever','Don\'t know/refused'), ordered=T
),

SOCIALSUPP2_F = factor(
  case_when(
    SOCIALSUPP2_F == 1 ~ 'Most of the time',
    SOCIALSUPP2_F == 2 ~ 'Some of the time',
    SOCIALSUPP2_F == 3 ~ 'Hardly ever',
    SOCIALSUPP2_F == 4 ~ 'Don\'t know/refused'
  ), levels=c('Most of the time','Some of the time','Hardly
ever','Don\'t know/refused'), ordered=T
),
```

```r
    STATUS_D_TO_F = factor(
      case_when(
        STATUS_D_TO_F == 1 ~ 'Re-interviewed patient',
        STATUS_D_TO_F == 2 ~ 'Re-interviewed proxy',
        STATUS_D_TO_F == 3 ~ 'Deceased',
        STATUS_D_TO_F == 4 ~ 'Refused',
        STATUS_D_TO_F == 5 ~ 'Unable to locate'
      )
    ),

    STATUS_F_TO_Y = factor(
      case_when(
        STATUS_F_TO_Y == 1 ~ 'Re-interviewed patient',
        STATUS_F_TO_Y == 2 ~ 'Re-interviewed proxy',
        STATUS_F_TO_Y == 3 ~ 'Deceased',
        STATUS_F_TO_Y == 4 ~ 'Refused',
        STATUS_F_TO_Y == 5 ~ 'Unable to locate'
      )
    ),

    STATUS_D_TO_Y = factor(
      case_when(
        STATUS_D_TO_Y == 1 ~ 'Re-interviewed patient',
        STATUS_D_TO_Y == 2 ~ 'Re-interviewed proxy',
        STATUS_D_TO_Y == 3 ~ 'Deceased',
        STATUS_D_TO_Y == 4 ~ 'Refused',
        STATUS_D_TO_Y == 5 ~ 'Unable to locate'
      )
    )

) %>%
mutate(
  across(
    all_of(c('DIAG_TIA', 'MARRIED', 'RACE_WHITE', 'DIAG_TIA',
  'COMORB_ARTHRITIS', 'COMORB_CANCER', 'COMORB_RESPIRATORY',
  'COMORB_DIABETES', 'COMORB_HTN', 'COMORB_HEART', 'COMORB_OTHER_CIRC',
  'COMORB_KIDNEY', 'COMORB_OBESITY', 'COMORB_MENTAL',
  'COMORB_FRACTURES', 'COMPL_ACUTERESPIR', 'COMPL_INFECTIOUS',
  'COMPL_BEDSORES', 'COMPL_INFECT_KIDNEY', 'COMPL_DEHYDRATION',
  'STROKESYM_HEMIP', 'STROKESYM_PARALYSIS', 'STROKESYM_APHASIA',
  'STROKESYM_DYSPHAGIA', 'STROKESYM_NEUROMUSC', 'HOMEHEALTH_D',
  'SIGNS_OF_DEHYDRATION_A', 'SHORT_BREATH_EXERTION_A',
  'SHORT_BREATH_REST_A', 'STANDING_PROBLEM_A',
  'SIGNS_OF_DEHYDRATION_D', 'SHORT_BREATH_EXERTION_D',
```

```
      'SHORT_BREATH_REST_D', 'STANDING_PROBLEM_D', 'OPTIMISTIC_Y_N',
      'INPERSON_INTERVIEW', 'CESDDEPRESSED_D', 'CESDDEPRESSED_F',
      'CESDDEPRESSED_Y')),
            ~ factor(if_else(.x == 1, 'Yes', 'No')),
            .names = "{.col}"
    ),
    across(
      all_of(c('CESDPOSITIVE4_D', 'CESDPOSITIVE8_D', 'CESDPOSITIVE12_D',
      'CESDPOSITIVE16_D', 'CESDPOSITIVE4_F', 'CESDPOSITIVE8_F',
      'CESDPOSITIVE12_F', 'CESDPOSITIVE16_F', 'CESDPOSITIVE4_Y',
      'CESDPOSITIVE8_Y', 'CESDPOSITIVE12_Y', 'CESDPOSITIVE16_Y',
      'NEWCESD1_D', 'NEWCESD2_D', 'NEWCESD3_D', 'NEWCESD5_D', 'NEWCESD6_D',
      'NEWCESD7_D', 'NEWCESD9_D', 'NEWCESD10_D', 'NEWCESD11_D',
      'NEWCESD13_D', 'NEWCESD14_D', 'NEWCESD15_D', 'NEWCESD17_D',
      'NEWCESD18_D', 'NEWCESD19_D', 'NEWCESD20_D', 'NEWCESD1_F',
      'NEWCESD2_F', 'NEWCESD3_F', 'NEWCESD5_F', 'NEWCESD6_F', 'NEWCESD7_F',
      'NEWCESD9_F', 'NEWCESD10_F', 'NEWCESD11_F', 'NEWCESD13_F',
      'NEWCESD14_F', 'NEWCESD15_F', 'NEWCESD17_F', 'NEWCESD18_F',
      'NEWCESD19_F', 'NEWCESD20_F', 'NEWCESD1_Y', 'NEWCESD2_Y',
      'NEWCESD3_Y', 'NEWCESD5_Y', 'NEWCESD6_Y', 'NEWCESD7_Y', 'NEWCESD9_Y',
      'NEWCESD10_Y', 'NEWCESD11_Y', 'NEWCESD13_Y', 'NEWCESD14_Y',
      'NEWCESD15_Y', 'NEWCESD17_Y', 'NEWCESD18_Y', 'NEWCESD19_Y',
      'NEWCESD20_Y')),
        ~ factor(
            case_when(
              .x == 0 ~ 'Rarely or none of the time (Less than 1 day)',
              .x == 1 ~ 'Some or little of the time (1-2 days)',
              .x == 2 ~ 'Occasionally or a moderate amount of time (3-4
      days)',
              .x == 3 ~ 'Most or all of the time (5-7 days)'
            ),
            levels = c(
              'Rarely or none of the time (Less than 1 day)',
              'Some or little of the time (1-2 days)',
              'Occasionally or a moderate amount of time (3-4 days)',
              'Most or all of the time (5-7 days)'
            ), ordered = TRUE
            )
    ),
    across(
      all_of(c('NEWCESD4_D', 'NEWCESD8_D', 'NEWCESD12_D', 'NEWCESD16_D',
      'NEWCESD4_F', 'NEWCESD8_F', 'NEWCESD12_F', 'NEWCESD16_F',
      'NEWCESD4_Y', 'NEWCESD8_Y', 'NEWCESD12_Y', 'NEWCESD16_Y')),
        ~ factor(
            case_when(
```

```
            .x == 0 ~ 'Most or all of the time (5-7 days)',
            .x == 1 ~ 'Occasionally or a moderate amount of time (3-4
    days)',
            .x == 2 ~ 'Some or little of the time (1-2 days)',
            .x == 3 ~ 'Rarely or none of the time (Less than 1 day)'
          ),
          levels = c(
            'Most or all of the time (5-7 days)',
            'Occasionally or a moderate amount of time (3-4 days)',
            'Some or little of the time (1-2 days)',
            'Rarely or none of the time (Less than 1 day)'
          ), ordered=T
        )
      ),
    across(
      all_of(c('COMPAR_SOCIALOUT_F', 'COMPAR_SOCIALIN_F',
    'COMPAR_HOBBY_F', 'COMPAR_RELIGIOUS_F', 'COMPAR_SOCIALOUT_Y',
    'COMPAR_SOCIALIN_Y', 'COMPAR_HOBBY_Y', 'COMPAR_RELIGIOUS_Y')),
      ~ factor(
        case_when(
            .x == 1 ~ 'Did not participate in this life situation',
            .x == 2 ~ 'Participated monthly (Once every 3-4 weeks)',
            .x == 3 ~ 'Participated bi-weekly (Once every 2 weeks)',
            .x == 4 ~ 'Participated weekly (1-4 days per week)',
            .x == 5 ~ 'Participated daily/Almost daily (5 or more days
    per week)'
          ),
        levels = c(
            'Did not participate in this life situation',
            'Participated monthly (Once every 3-4 weeks)',
            'Participated bi-weekly (Once every 2 weeks)',
            'Participated weekly (1-4 days per week)',
            'Participated daily/Almost daily (5 or more days per week)'
          ), ordered=T
        )
      )
    )

head(df)


##
   ----------------------------------------------------------------------------------
df <- df %>%
  mutate(across(everything(), ~ replace(.x, .x == -9, NA)))
```

```
##
    ----------------------------------------------------------------------------
missing_percent <- df %>%
  summarise(across(everything(), ~ mean(is.na(.)) * 100)) %>%
  pivot_longer(cols = everything(), names_to = 'Variable', values_to =
    'PercentMissing') %>%
  arrange(desc(PercentMissing))

missing_percent
missing_percent %>% filter(Variable == 'DUKE_Y')
missing_percent %>% filter(str_detect(Variable,
    regex('FIMTOTAL|FIMMOTOR|FIMCOGN', ignore_case=T)))
missing_percent %>% filter(str_detect(Variable, regex('DUKE',
    ignore_case=T)))


##
    ----------------------------------------------------------------------------
df <- df %>%
  mutate(
    support_high_Y = factor(
      case_when(
        DUKE_Y >= 50 ~ 1,
        DUKE_Y < 50 ~ 0
      )
    ),
    support_high_F = factor(
      case_when(
        DUKE_F >= 50 ~ 1,
        DUKE_F < 50 ~ 0
      )
    )
  )


##
    ----------------------------------------------------------------------------
df <- df %>%
  mutate(
    FIM_MOTOR_CHANGE_Y = FIMMOTOR_Y - FIMMOTOR_A,
    FIM_COGN_CHANGE_Y = FIMCOGN_Y - FIMCOGN_A,
    FIM_TOTAL_CHANGE_Y = FIMTOTAL_Y - FIMTOTAL_A,
```

```
    FIM_MOTOR_CHANGE_F = FIMMOTOR_F - FIMMOTOR_A,
    FIM_COGN_CHANGE_F = FIMCOGN_F - FIMCOGN_A,
    FIM_TOTAL_CHANGE_F = FIMTOTAL_F - FIMTOTAL_A,
  )


## ----------------------------------------------------------------------------------

df %>%
  saveRDS('data/data.rds')
df %>%
  write_csv('data/data_cleaned.csv')


## ----------------------------------------------------------------------------------

vars_F <- c(
  "support_high_F",                   # treatment
  "FIM_TOTAL_CHANGE_F",               # outcome
  "FIM_MOTOR_CHANGE_F",
  "FIM_COGN_CHANGE_F",
  "GENDER", "RACE", "AGE_CAT",        # demographics
  "SCHOOL", "MARITAL",
  "FIMTOTAL_A",                       # baseline functional status
  "LOS",                              # stroke severity proxy
  "COMORB_DIABETES",                  # comorbidities
  "COMORB_HTN",
  "COMORB_HEART",
  "COMORB_OBESITY",
  "COMORB_CANCER",
  "COMORB_MENTAL",
  "COMORB_RESPIRATORY",
  "COMORB_ARTHRITIS",
  "COMORB_KIDNEY",
  "COMORB_OTHER_CIRC",
  "COMORB_FRACTURES",
  "COMORBIDSUM"

)

df_F <- df %>%
  select(all_of(vars_F)) %>%
  drop_na()
```

```r
head(df_F)


## 
   ---------------------------------------------------------------------------------------------
all_vars_F <- c("GENDER", "RACE", "AGE_CAT", "SCHOOL", "MARITAL",
    "FIMTOTAL_A", "support_high_F", "FIM_TOTAL_CHANGE_F",
    "FIM_MOTOR_CHANGE_F", "FIM_COGN_CHANGE_F", "LOS", "COMORB_DIABETES",
    "COMORB_HTN", "COMORB_HEART", "COMORB_OBESITY", "COMORB_CANCER",
    "COMORB_MENTAL", "COMORB_RESPIRATORY", "COMORB_ARTHRITIS",
    "COMORB_KIDNEY", "COMORB_OTHER_CIRC", "COMORB_FRACTURES",
    "COMORBIDSUM")

factor_vars_F <- names(df_F)[sapply(df_F, is.factor)]
table1_F <- CreateTableOne(vars=all_vars_F, strata='support_high_F',
    data=df_F, factorVars=factor_vars_F)
t1_F <- print(table1_F, printToggle = F, noSpaces = T, showAllLevels = T,
    missing=F, quote=F)
kbl(t1_F, booktabs = T, longtable=T, format = "latex") %>%
  kable_styling(latex_options = c('striped', 'repeat_header')) %>%
  write_clip()


# kableone(t1_F)


## 
   ---------------------------------------------------------------------------------------------
str(df)


## 
   ---------------------------------------------------------------------------------------------
tst <- print(table1_F, printToggle=F, noSpaces=T, showAllLevels=F,
    missing=F, quote=F)
print(tst)


## 
   ---------------------------------------------------------------------------------------------
gt_table_F <- df_F %>%
  as_tibble() %>%
  rename(
    'Gender' = GENDER,
    'Race/Ethnicity' = RACE,
```

```r
    `Age Category` = AGE_CAT,
    `Education Level` = SCHOOL,
    `Marital Status` = MARITAL,
    `Admission FIM Total` = FIMTOTAL_A,
    # `FIM Total Change` = FIM_TOTAL_CHANGE_F,
    # `FIM Motor Change` = FIM_MOTOR_CHANGE_F,
    # `FIM Cognitive Change` = FIM_COGN_CHANGE_F,
    `Length of Stay` = LOS,
    `Comorbidity: Diabetes` = COMORB_DIABETES,
    `Comorbidity: Hypertension` = COMORB_HTN,
    `Comorbidity: Heart Disease` = COMORB_HEART,
    `Comorbidity: Obesity` = COMORB_OBESITY,
    `Comorbidity: Cancer` = COMORB_CANCER,
    `Comorbidity: Mental Illness` = COMORB_MENTAL,
    `Comorbidity: Respiratory Disease` = COMORB_RESPIRATORY,
    `Comorbidity: Arthritis` = COMORB_ARTHRITIS,
    `Comorbidity: Kidney Disease` = COMORB_KIDNEY,
    `Comorbidity: Other Circulatory` = COMORB_OTHER_CIRC,
    `Comorbidity: Fractures` = COMORB_FRACTURES,
    `Comorbidity Count` = COMORBIDSUM,
    `Support Level` = support_high_F
  ) %>%
  select(-c(FIM_TOTAL_CHANGE_F, FIM_MOTOR_CHANGE_F, FIM_COGN_CHANGE_F))
    %>%
  # select(all_of(c("Support Level", all_vars_F))) %>%
  tbl_summary(
    by=`Support Level`,
    digits = all_continuous() ~ 1,
    type = all_categorical() ~ "categorical"
    ) %>%
  add_p(pvalue_fun = ~style_pvalue(.x, digits=3)) %>%
  # add_overall() %>%
  # add_n() %>%
  modify_spanning_header(c("stat_1", "stat_2") ~ "Stratified by social
    support.") %>%
  bold_labels() %>%
  italicize_levels()


gt_table_F


## 
    ------------------------------------------------------------------------------
df_F %>%
```

```r
  tbl_summary()


##
    ----------------------------------------------------------------------------------
library(huxtable)
library(xtable)
gt_table_F %>% as_gt() %>% gt::as_latex() %>% write_clip()
# as_kable_extra(gt_table_F, format = "latex") %>% write_clip()
as_kable(gt_table_F, format = "latex", longtable=T, booktabs=T, escape=T)
    %>% write_clip()
as_hux_table(tbl) %>% huxtable::to_latex() %>% write_clip()
kbl(gt_table_F, booktabs = T, longtable=T, format = "latex") #%>%
    write_clip()


##
    ----------------------------------------------------------------------------------
latex_table_code <- as_kable(
  gt_table_F,
  format = "latex",
  longtable = TRUE,
  booktabs = TRUE,
  escape = FALSE  # Allow special LaTeX characters in labels
)
# latex_table_code <- str_replace_all(latex_table_code,
    "(\\d)%(?=[^\\d])", "\\1\\\\%")



##
    ----------------------------------------------------------------------------------
kable(gt_table_F, format = "latex", booktabs = TRUE, longtable=TRUE) %>%
  kable_styling(latex_options = c("striped", "hold_position")) %>%
  column_spec(1, width = "25em") %>%
  add_header_above(c(" " = 4)) %>%
  kable_classic(full_width = F, html_font = "Helvetica") %>% write_clip()


##
    ----------------------------------------------------------------------------------
library(WeightIt)
ipw_model <- weightit(
  support_high_F ~ AGE_CAT + GENDER + RACE + SCHOOL + MARITAL +
                   FIMTOTAL_A + LOS +
```

```
                     COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
                     COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
                     COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY +
                     COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
   data = df_F,
   method = "glm",
   estimand = "ATE"
)

summary(ipw_model)


## -------------------------------------------------------------------------------
library(survey)

# Attach weights to original data
df_F$weights <- ipw_model$weights

# Set up survey design object
svy_design_F <- svydesign(ids = ~1, data = df_F, weights = ~weights)
summary(svy_design_F)


## -------------------------------------------------------------------------------
library(cobalt)
# Generate a balance table
bt <- bal.tab(ipw_model, un = TRUE, m.threshold = 0.1)
bt$Balance %>%
  gt() %>%
  tab_header(
    title = "Covariate Balance Before and After Weighting"
  ) %>%
  fmt_number(
    columns = where(is.numeric),
    decimals = 3
  )


## -------------------------------------------------------------------------------
love.plot(ipw_model, binary = "std", thresholds = c(m = 0.1))
```

31

```
##
    ---------------------------------------------------------------------------
# lm_ipw_dr <- svyglm(FIM_TOTAL_CHANGE_F ~ support_high_F + AGE + GENDER
    + RACE + SCHOOL + MARITAL +
#                       FIMTOTAL_A + LOS + COMORB_DIABETES + COMORB_HTN +
    COMORB_HEART,
#                       design = svy_design)
lm_ipw_total_F <- svyglm(
  FIM_TOTAL_CHANGE_F ~ support_high_F + AGE_CAT + GENDER + RACE + SCHOOL
    + MARITAL +
    FIMTOTAL_A + LOS +
    COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
    COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
    COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY +
    COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
  design = svy_design_F
)

summary(lm_ipw_total_F)


##
    ---------------------------------------------------------------------------
lm_ipw_motor_F <- svyglm(
  FIM_MOTOR_CHANGE_F ~ support_high_F + AGE_CAT + GENDER + RACE + SCHOOL
    + MARITAL +
    FIMTOTAL_A + LOS +
    COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
    COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
    COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY +
    COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
  design = svy_design_F
)


summary(lm_ipw_motor_F)

##
    ---------------------------------------------------------------------------
lm_ipw_cogn_F <- svyglm(
  FIM_COGN_CHANGE_F ~ support_high_F + AGE_CAT + GENDER + RACE + SCHOOL +
    MARITAL +
    FIMTOTAL_A + LOS +
    COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
```

```
      COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
      COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY +
      COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
    design = svy_design_F
)


summary(lm_ipw_cogn_F)



##
    --------------------------------------------------------------------------------
# Create individual regression tables
tbl_total_F <- tbl_regression(lm_ipw_total_F, exponentiate = FALSE)
tbl_cogn_F  <- tbl_regression(lm_ipw_cogn_F,  exponentiate = FALSE)
tbl_motor_F <- tbl_regression(lm_ipw_motor_F, exponentiate = FALSE)

# Merge them side-by-side
merged_tbl_F <- tbl_merge(
  tbls = list(tbl_total_F, tbl_cogn_F, tbl_motor_F),
  tab_spanner = c("**Total**", "**Cognition**", "**Motor**")
)

tbl_total_F %>%
  as_kable_extra(
    format = "latex",
    longtable = TRUE,
    booktabs = FALSE,  # Set to FALSE to allow borders
    caption = "Survey-weighted linear regression estimating the effect of
    high support on FIM total score change at 3 months.",
    label = 'tab:ipw_fim_total_F'
  ) %>%
  column_spec(1, border_left = TRUE) %>%
  column_spec(4, border_right = TRUE) %>%
  row_spec(0, bold = TRUE, background = "gray!10") %>%  # Bold header
    with dark gray background
  # row_spec(1:10, background = "gray90") %>%   # Shade all data rows
    light gray
  # str_replace_all('gray90', 'gray!90') %>%
  kable_styling(
    latex_options = c("hold_position"),
    full_width = FALSE,
    font_size = 8,          # Smaller text
    position = "center"     # Center the table
  ) %>%
  write_clip()
```

```r
##
  ----------------------------------------------------------------------------------------------------------
vars_Y <- c(
  "FIM_TOTAL_CHANGE_Y", "FIM_MOTOR_CHANGE_Y", "FIM_COGN_CHANGE_Y",
  "support_high_Y",
  "AGE", "GENDER", "RACE", "SCHOOL", "MARITAL", "AGE_CAT",
  "FIMTOTAL_A", "LOS",
  "COMORB_DIABETES",                    # comorbidities
  "COMORB_HTN",
  "COMORB_HEART",
  "COMORB_OBESITY",
  "COMORB_CANCER",
  "COMORB_MENTAL",
  "COMORB_RESPIRATORY",
  "COMORB_ARTHRITIS",
  "COMORB_KIDNEY",
  "COMORB_OTHER_CIRC",
  "COMORB_FRACTURES",
  "COMORBIDSUM"
)

df_Y <- df %>%
  select(all_of(vars_Y)) %>%
  drop_na()


##
  ----------------------------------------------------------------------------------------------------------
ipw_model_Y <- weightit(
  support_high_Y ~ AGE_CAT + GENDER + RACE + SCHOOL + MARITAL +
    FIMTOTAL_A + LOS +
    COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
    COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
    COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY +
    COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
  data = df_Y,
  method = "ps",
  estimand = "ATE"
)

# Add weights to your data
df_Y$weights <- ipw_model_Y$weights
```

```
## 
  -----------------------------------------------------------------------------------
svy_design_Y <- svydesign(ids = ~1, data = df_Y, weights = ~weights)


## 
  -----------------------------------------------------------------------------------
lm_ipw_total_Y <- svyglm(
  FIM_TOTAL_CHANGE_Y ~ support_high_Y + AGE_CAT + GENDER + RACE +
    SCHOOL + MARITAL + FIMTOTAL_A + LOS +
    COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
    COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
    COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY +
    COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
  design = svy_design_Y
)

summary(lm_ipw_total_Y)


## 
  -----------------------------------------------------------------------------------
lm_ipw_motor_Y <- svyglm(
  FIM_MOTOR_CHANGE_Y ~ support_high_Y + AGE_CAT + GENDER + RACE +
    SCHOOL + MARITAL + FIMTOTAL_A + LOS +
    COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
    COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
    COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY +
    COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
  design = svy_design_Y
)

summary(lm_ipw_motor_Y)


## 
  -----------------------------------------------------------------------------------
lm_ipw_cogn_Y <- svyglm(FIM_COGN_CHANGE_Y ~ support_high_Y + AGE_CAT +
    GENDER + RACE +
                        SCHOOL + MARITAL + FIMTOTAL_A + LOS +
                        COMORB_DIABETES + COMORB_HTN + COMORB_HEART,
                        design = svy_design_Y)
```

```
summary(lm_ipw_cogn_Y)


##
    -----------------------------------------------------------------------------
# Create individual regression tables
tbl_total_Y <- tbl_regression(lm_ipw_total_Y, exponentiate = FALSE)
tbl_cogn_Y  <- tbl_regression(lm_ipw_cogn_Y,  exponentiate = FALSE)
tbl_motor_Y <- tbl_regression(lm_ipw_motor_Y, exponentiate = FALSE)

# Merge them side-by-side
merged_tbl_Y <- tbl_merge(
  tbls = list(tbl_total_Y, tbl_cogn_Y, tbl_motor_Y),
  tab_spanner = c("**Total**", "**Cognition**", "**Motor**")
)

tbl_total_Y %>%
  as_kable_extra(
    format = "latex",
    longtable = TRUE,
    booktabs = FALSE,  # Set to FALSE to allow borders
    caption = "Survey-weighted linear regression estimating the effect of
    high support on FIM total score change at 1 year.",
    label = 'tab:ipw_fim_total_F'
  ) %>%
  column_spec(1, border_left = TRUE) %>%
  column_spec(4, border_right = TRUE) %>%
  row_spec(0, bold = TRUE, background = "gray!10") %>%  # Bold header
    with dark gray background
  # row_spec(1:10, background = "gray90") %>%    # Shade all data rows
    light gray
  # str_replace_all('gray90', 'gray!90') %>%
  kable_styling(
    latex_options = c("hold_position"),
    full_width = FALSE,
    font_size = 8,          # Smaller text
    position = "center"     # Center the table
  ) %>%
  write_clip()


##
    -----------------------------------------------------------------------------
tbl_3mo <- tbl_regression(lm_ipw_dr, exponentiate = FALSE, label =
    list(support_high_F ~ "High Support"), intercept=T) %>%
```

```r
  add_significance_stars()
tbl_12mo <- tbl_regression(lm_ipw_total_Y, exponentiate = FALSE, label =
    list(support_high_Y ~ "High Support"), intercept=T) %>%
  add_significance_stars()

tbl_3mo$table_body$variable[tbl_3mo$table_body$variable ==
    "support_high_F"] <- 'support_high'
tbl_12mo$table_body$variable[tbl_12mo$table_body$variable ==
    "support_high_Y"] <- 'support_high'

tbl_combined <- tbl_merge(
  list(tbl_3mo, tbl_12mo),
  tab_spanner = c("**3-Month Outcome**", "**12-Month Outcome**")
)
tbl_combined %>%
  # as_flex_table() %>%
  as_gt() %>%
  gt::opt_row_striping()

tbl_combined %>%
  as_kable_extra(
    format = "latex",
    longtable = TRUE,
    booktabs = FALSE,  # Set to FALSE to allow borders
    caption = "Survey-weighted linear regression estimating the effect of
    high support on FIM total score change at 3 months and 1 year.",
    label = 'tab:ipw_fim_total_combined'
  ) %>%
  add_header_above(
    c(" " = 1,
      "\\cellcolor{gray!20}\\textbf{3-Month Outcome}" = 2,
      "\\cellcolor{gray!20}\\textbf{12-Month Outcome}" = 2),
    escape = FALSE
  ) %>%
  # column_spec(1, border_left = TRUE) %>%
  # column_spec(4, border_right = TRUE) %>%
  row_spec(0, bold = TRUE, background = "gray!10") %>%  # Bold header
    with dark gray background
  # row_spec(1:10, background = "gray90") %>%   # Shade all data rows
    light gray
  # str_replace_all('gray90', 'gray!90') %>%
  kable_styling(
    latex_options = c("hold_position"),
    full_width = FALSE,
    font_size = 8,         # Smaller text
```

```r
    position = "center"     # Center the table
  ) %>%
  write_clip()



##
    --------------------------------------------------------------------------------------------
# One-hot encode categorical vars
X <- model.matrix(~ AGE_CAT + GENDER + RACE + SCHOOL + MARITAL +
    FIMTOTAL_A + LOS +
                    COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
                    COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
                    COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY
    +
                    COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
                data = df_F)[, -1]   # Drop intercept

W <- as.numeric(as.character(df_F$support_high_F))
Y <- df_F$FIM_TOTAL_CHANGE_F

rf_Y <- regression_forest(X, Y, seed=123)
rf_W <- regression_forest(X, W, seed=123)

cf_model_F <- causal_forest(
  X = X,
  Y = Y,
  W = W,
  Y.hat = predict(rf_Y)$predictions,
  W.hat = predict(rf_W)$predictions,
  seed=123
)



##
    --------------------------------------------------------------------------------------------
average_treatment_effect(cf_model_F)


##
    --------------------------------------------------------------------------------------------
# Calibrate causal forest
calib_test_F <- test_calibration(cf_model_F)

# Print the results
print(calib_test_F)
```

```
##
-----------------------------------------------------------------------------------------------
tbl_regression(calib_test_F) %>%
  as_kable_extra(
    format = "latex",
    longtable = TRUE,
    booktabs = FALSE,  # Set to FALSE to allow borders
    caption = "Calibration test results for the Causal Forest model.",
    label = 'cf_calib_test'
  ) %>%
  # column_spec(1, border_left = TRUE) %>%
  # column_spec(4, border_right = TRUE) %>%
  row_spec(0, bold = TRUE, background = "gray!10") %>%  # Bold header
    with dark gray background
  # row_spec(1:10, background = "gray90") %>%   # Shade all data rows
    light gray
  # str_replace_all('gray90', 'gray!90') %>%
  kable_styling(
    latex_options = c("hold_position"),
    full_width = FALSE,
    font_size = 8,          # Smaller text
    position = "center"     # Center the table
  ) %>%
  write_clip()


##
-----------------------------------------------------------------------------------------------
cates_F <- predict(cf_model_F)$predictions

cate_p <- data.frame(CATE=cates_F) %>%
  ggplot(aes(x=CATE)) +
  geom_histogram(fill='steelblue', color='black') +
  labs(
    # title = 'Distribution of  Estimated CATEs (3 months)',
    x = 'Estimated Treatment Effect',
    y = 'Count'
  ) +
  theme_classic() +
  theme_custom(axis_text_size = 20, axis_title_size = 22)

ggsave('images/cate_histogram_F.png', plot=cate_p, width=6, height=4,
    dpi=300)
```

```
print(cate_p)


##
    -------------------------------------------------------------------------------
vimp_F <- variable_importance(cf_model_F)

# Pair with variable names
importance_df_F <- data.frame(
  Variable = colnames(X),
  Importance = vimp_F
) %>%
  arrange(desc(Importance))

# Show top 10
head(importance_df_F, 10)


##
    -------------------------------------------------------------------------------
vimp_p <- ggplot(importance_df_F[1:10, ], aes(x = reorder(Variable,
    Importance), y = Importance)) +
  geom_col(fill='steelblue', color='black') +
  coord_flip() +
  labs(
    # title = "Top Variables Driving Heterogeneity (3 months)",
    x = "Variable",
    y = "Importance Score"
    ) +
  theme_classic() +
  theme_custom(axis_text_size = 16, axis_title_size = 22)

ggsave('images/vimp_F.png', plot=vimp_p, width=8, height=4, dpi=300)
print(vimp_p)


##
    -------------------------------------------------------------------------------
df_F$cate <- cates_F
# Create quartiles based on baseline FIMTOTAL_A
df_F$fim_quartile <- cut(
  df_F$FIMTOTAL_A,
  breaks = quantile(df_F$FIMTOTAL_A, probs = seq(0, 1, 0.25), na.rm =
    TRUE),
  include.lowest = TRUE,
```

```r
    labels = c("Q1 (lowest)", "Q2", "Q3", "Q4 (highest)")
)

df_F %>%
  group_by(fim_quartile) %>%
  summarise(
    mean_cate = mean(cate, na.rm = TRUE),
    sd_cate = sd(cate, na.rm = TRUE),
    n = n()
  )



## ------------------------------------------------------------------------------------------
df_F %>%
  group_by(fim_quartile) %>%
  summarise(
    mean_cate = mean(cate, na.rm = TRUE),
    sd_cate = sd(cate, na.rm = TRUE),
    n = n()
  ) %>%
  kable(
    format = "latex",
    longtable = TRUE,
    booktabs = FALSE,  # Set to FALSE to allow borders
    caption = "Mean estimated (CATE) and standard deviation by baseline
    FIM quartile.",
    label = 'cate_by_fim_table_F'
  ) %>%
  # column_spec(1, border_left = TRUE) %>%
  # column_spec(4, border_right = TRUE) %>%
  row_spec(0, bold = TRUE, background = "gray!10") %>%  # Bold header
    with dark gray background
  # row_spec(1:10, background = "gray90") %>%   # Shade all data rows
    light gray
  # str_replace_all('gray90', 'gray!90') %>%
  kable_styling(
    latex_options = c("hold_position"),
    full_width = FALSE,
    font_size = 8,         # Smaller text
    position = "center"    # Center the table
  ) %>%
  write_clip()
```

```
##
    --------------------------------------------------------------------------------
kruskal.test(cate ~ fim_quartile, data = df_Y) %>%
  tidy() %>%
  mutate(p.value = format.pval(p.value, digits = 3, eps = 1e-4)) %>%
  kable(format = "latex", booktabs = F, longtable=T,
        caption = "KruskalWallis test comparing CATE distributions across
    baseline FIM quartiles.",
        col.names = c("Test", "Statistic", "Degrees of Freedom",
    "p-value")) %>%
  row_spec(0, bold = TRUE, background = "gray!10") %>%
  kable_styling(latex_options = c("hold_position")) %>%
  write_clip()


##
    --------------------------------------------------------------------------------
cate_boxplot_p <- ggplot(df_F, aes(x = fim_quartile, y = cate)) +
  geom_boxplot(fill = "lightblue") +
  labs(
    # title = "Estimated Treatment Effect by Baseline Function
    (FIMTOTAL_A Quartiles)",
    x = "Baseline Function Quartile",
    y = "Estimated Treatment Effect (CATE)"
  ) +
  theme_classic() +
  theme_custom(axis_text_size = 20, axis_title_size = 22)
ggsave('images/cate_boxplot_by_FIMTOTAL_A_F.png', plot=cate_boxplot_p,
    width=6, height=4, dpi=300)
print(cate_boxplot_p)


##
    --------------------------------------------------------------------------------
df_F$los_quartile <- cut(
  df_F$LOS,
  breaks = quantile(df_F$LOS, probs = seq(0, 1, 0.25), na.rm = TRUE),
  include.lowest = TRUE,
  labels = c("Q1 (lowest)", "Q2", "Q3", "Q4 (highest)")
)

df_F %>%
  group_by(los_quartile) %>%
  summarise(
    mean_cate = mean(cate, na.rm = TRUE),
```

```
    sd_cate = sd(cate, na.rm = TRUE),
    n = n()
  )


## 
  -----------------------------------------------------------------------------------
ggplot(df_F, aes(x = los_quartile, y = cate)) +
  geom_boxplot(fill = "lightblue") +
  labs(
    title = "Estimated Treatment Effect by Length of Stay (LOS
    Quartiles)",
    x = "LOS Quartile",
    y = "Estimated Treatment Effect (CATE)"
  )



## 
  -----------------------------------------------------------------------------------
# One-hot encode categorical vars
X <- model.matrix(~ AGE_CAT + GENDER + RACE + SCHOOL + MARITAL +
    FIMTOTAL_A + LOS +
                     COMORB_DIABETES + COMORB_HTN + COMORB_HEART +
                     COMORB_OBESITY + COMORB_CANCER + COMORB_MENTAL +
                     COMORB_RESPIRATORY + COMORB_ARTHRITIS + COMORB_KIDNEY
    +
                     COMORB_OTHER_CIRC + COMORB_FRACTURES + COMORBIDSUM,
                data = df_Y)[, -1]  # Drop intercept

W <- as.numeric(as.character(df_Y$support_high_Y))
Y <- df_Y$FIM_TOTAL_CHANGE_Y

rf_Y <- regression_forest(X, Y, seed=123)
rf_W <- regression_forest(X, W, seed=123)

cf_model_Y <- causal_forest(
  X = X,
  Y = Y,
  W = W,
  Y.hat = predict(rf_Y)$predictions,
  W.hat = predict(rf_W)$predictions,
  seed=123
)
```

```
##
   --------------------------------------------------------------------------------
average_treatment_effect(cf_model_Y)


##
   --------------------------------------------------------------------------------
# Calibrate causal forest
calib_test_Y <- test_calibration(cf_model_Y)

# Print the results
print(calib_test_Y)


##
   --------------------------------------------------------------------------------
tbl_regression(calib_test_F) %>%
  as_kable_extra(
    format = "latex",
    longtable = TRUE,
    booktabs = FALSE,  # Set to FALSE to allow borders
    caption = "Calibration test results for the Causal Forest model.",
    label = 'cf_calib_test'
  ) %>%
  # column_spec(1, border_left = TRUE) %>%
  # column_spec(4, border_right = TRUE) %>%
  row_spec(0, bold = TRUE, background = "gray!10") %>%  # Bold header
    with dark gray background
  # row_spec(1:10, background = "gray90") %>%   # Shade all data rows
    light gray
  # str_replace_all('gray90', 'gray!90') %>%
  kable_styling(
    latex_options = c("hold_position"),
    full_width = FALSE,
    font_size = 8,          # Smaller text
    position = "center"     # Center the table
  ) %>%
  write_clip()


##
   --------------------------------------------------------------------------------
cates_Y <- predict(cf_model_Y)$predictions

cate_p <- data.frame(CATE=cates_Y) %>%
```

```r
  ggplot(aes(x=CATE)) +
  geom_histogram(fill='steelblue', color='black') +
  labs(
    # title = 'Distribution of  Estimated CATEs (1 year)',
    x = 'Estimated Treatment Effect',
    y = 'Count'
  ) +
  theme_classic() +
  theme_custom(axis_text_size = 20, axis_title_size = 22)

ggsave('images/cate_histogram_Y.png', plot=cate_p, width=6, height=4,
    dpi=300)
print(cate_p)



##
    ----------------------------------------------------------------------------
vimp_Y <- variable_importance(cf_model_Y)

# Pair with variable names
importance_df_Y <- data.frame(
  Variable = colnames(X),
  Importance = vimp_Y
) %>%
  arrange(desc(Importance))

# Show top 10
head(importance_df_Y, 10)


##
    ----------------------------------------------------------------------------
vimp_p <- ggplot(importance_df_Y[1:10, ], aes(x = reorder(Variable,
    Importance), y = Importance)) +
  geom_col(fill='steelblue', color='black') +
  coord_flip() +
  labs(title = "Top Variables Driving Heterogeneity (1 year)", x =
    "Variable", y = "Importance Score")

ggsave('images/vimp_Y.png', plot=vimp_p, width=6, height=4, dpi=300)
print(vimp_p)


##
    ----------------------------------------------------------------------------
```

```
df_Y$cate <- cates_Y
# Create quartiles based on baseline FIMTOTAL_A
df_Y$fim_quartile <- cut(
  df_Y$FIMTOTAL_A,
  breaks = quantile(df_Y$FIMTOTAL_A, probs = seq(0, 1, 0.25), na.rm =
    TRUE),
  include.lowest = TRUE,
  labels = c("Q1 (lowest)", "Q2", "Q3", "Q4 (highest)")
)

df_Y %>%
  group_by(fim_quartile) %>%
  summarise(
    mean_cate = mean(cate, na.rm = TRUE),
    sd_cate = sd(cate, na.rm = TRUE),
    n = n()
  )


## ------------------------------------------------------------------------------------------
ggplot(df_Y, aes(x = fim_quartile, y = cate)) +
  geom_boxplot(fill = "lightblue") +
  labs(
    title = "Estimated Treatment Effect by Baseline Function (FIMTOTAL_A
    Quartiles)",
    x = "Baseline Function Quartile",
    y = "Estimated Treatment Effect (CATE)"
  )


## ------------------------------------------------------------------------------------------
df_Y$los_quartile <- cut(
  df_Y$LOS,
  breaks = quantile(df_Y$LOS, probs = seq(0, 1, 0.25), na.rm = TRUE),
  include.lowest = TRUE,
  labels = c("Q1 (lowest)", "Q2", "Q3", "Q4 (highest)")
)

df_Y %>%
  group_by(los_quartile) %>%
  summarise(
    mean_cate = mean(cate, na.rm = TRUE),
    sd_cate = sd(cate, na.rm = TRUE),
```

```
    n = n()
  )


##
    ----------------------------------------------------------------------------

ggplot(df_Y, aes(x = los_quartile, y = cate)) +
  geom_boxplot(fill = "lightblue") +
  labs(
    title = "Estimated Treatment Effect by Length of Stay (LOS
    Quartiles)",
    x = "LOS Quartile",
    y = "Estimated Treatment Effect (CATE)"
  )
```

References

[1] Evelyn F Acoba. Social support and mental health: the mediating role of perceived stress. *Frontiers in Psychology*, 15:1330720, 2024.

[2] Susan Athey and Stefan Wager. Estimating treatment effects with causal forests: An application. *Observational studies*, 5(2):37–51, 2019.

[3] John Auerbach. The 3 buckets of prevention. *Journal of Public Health Management and Practice*, 22(3):215–218, 2016.

[4] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[5] W Eugene Broadhead, Stephen H Gehlbach, Frank V De Gruy, and Berton H Kaplan. The duke-unc functional social support questionnaire: Measurement of social support in family medicine patients. *Medical care*, pages 709–723, 1988.

[6] Centers for Disease Control and Prevention. Cdc health impact in 5 years (hi-5) initiative, 2024. Accessed: 2025-05-31.

[7] Centers for Disease Control and Prevention. Stroke facts and statistics, 2024. Accessed: 2025-05-31.

[8] Toughieda Elloker and Anthea J Rhoda. The relationship between social support and participation in stroke: a systematic review. *African journal of disability*, 7(1):1–9, 2018.

[9] Thomas A Glass, David B Matchar, Michael Belyea, and JR Feussner. Impact of social support on outcome in first stroke. *Stroke*, 24(1):64–70, 1993.

[10] Sarah Northcott, Becky Moss, Kirsty Harrison, and Katerina Hilari. A systematic review of the impact of stroke on social support and social networks: associated factors and patterns of change. *Clinical rehabilitation*, 30(8):811–831, 2016.

[11] Glenn Ostir, Kenneth Ottenbacher, and Yong-Fang Kuo. Stroke recovery in underserved populations 2005-2006 [united states], 2016.

[12] William J Powers, Alejandro A Rabinstein, Teri Ackerson, Opeolu M Adeoye, Nicholas C Bambakidis, Kyra Becker, José Biller, Michael Brown, Bart M Demaerschalk, Brian Hoh, et al. Guidelines for the early management of patients with acute ischemic stroke: 2019 update to the 2018 guidelines for the early management of acute ischemic stroke: a guideline for healthcare professionals from the american heart association/american stroke association. *Stroke*, 2019.

[13] Evie Tsouna-Hadjis, Kostas N Vemmos, Nikolaos Zakopoulos, and Stamatis Stamatelopoulos. First-stroke recovery process: the role of family social support. *Archives of physical medicine and rehabilitation*, 81(7):881–887, 2000.