Michelle Machado

Student topic write-up

November 30th, 2024

Evaluating Timely Treatment for Colorectal Cancer in Canadian Healthcare

Timeliness in healthcare refers to the system's capacity to provide care promptly, minimizing delays that could lead to adverse health outcomes. The Institute of Medicine identifies timeliness as one of the six aims for improving healthcare quality, emphasizing the reduction of waits and harmful delays for both those who receive and provide care[1]. In the Canadian healthcare system, which is universally funded and guarantees access to care, timeliness remains a critical challenge. The Canadian Institute for Health Information (CIHI) reports that many patients wait longer than the recommended benchmarks for procedures like cancer treatment, joint replacements, and diagnostic imaging[2]. Therefore, evaluating whether these expectations are consistently met is essential for assessing the system's effectiveness.

Timeliness is particularly crucial in cancer treatment, where delays can significantly impact survival rates and quality of life. Colorectal cancer (CRC), the third most common cancer in Canada, provides a compelling case study for measuring timeliness[3]. Research indicates that delays in initiating treatment for CRC, specifically beyond six weeks of diagnosis, are associated with poorer outcomes, including decreased survival rates[4]. Thus, assessing the healthcare system's performance in meeting recommended timeframes is vital. This project will focus on evaluating the proportion of CRC patients who receive treatment within the benchmark of six weeks from diagnosis. By examining this metric, the study aims to contribute to the broader evaluation and enhancement of healthcare system performance, ensuring better outcomes for all patients.

The dataset is constructed using three primary files-

1. **sd_table_demo_rev**: Contains demographic information such as patient id, birthdate, sex and income information.
2. **sd_table_death**: Contains data on patient health coverage start and end along with death.
3. **sd_table_diag**: Includes colorectal cancer diagnosis data with ICD-9 and ICD-10 codes to identify CRC diagnoses (also used to get comorbidity data), along with the diagnosis date.
4. **sd_table_drug**: Tracks treatment using ATC codes for CRC therapies and contains the treatment initiation date.

Inclusion Criteria

- Patients diagnosed with CRC for the first time with no prior cancer diagnosis.
- Patients 18 years and older at the time of CRC diagnosis.
- Patients diagnosed with CRC atleast 42 days prior to their coverage end date
- A 2-year wash-in period will be applied, requiring patients to have at least 2 years of healthcare history prior to diagnosis. This ensures sufficient history to rule out prior cancers.

Exclusion Criteria

- Patients with other cancers before their CRC diagnosis or during the period between their CRC diagnosis and first treatment initiation.

  [Note: Determined using ICD codes for other cancers (ICD10: C00-C97, D00-D48; ICD9: 140-239)[5,6]. This is to make sure that the treatment timelines being analyzed are specifically related to new colorectal cancer cases, without influence from past cancer treatment]

Special considerations include the exclusion of patients with prior cancer histories to focus exclusively on new colorectal cancer cases. This exclusion helps eliminate potential confounding variables arising from prior treatments or concurrent cancers that might influence patient outcomes.

Variables will be integrated at the patient level to provide a detailed view of how individual demographics impact the initiation and timeliness of colorectal cancer treatments giving insights on access issues due to income disparities. The main variable of interest is **within_time**, indicating whether treatment was initiated within 6 weeks (42 days) of diagnosis, based on research showing that delays beyond this period lead to poorer outcomes[4,7]. It is derived by calculating the diagnosis-to-treatment interval (DTI) (time between the diagnosis date and treatment initiation). If **dti** is 42 days or fewer, **within_time** is set to 1; otherwise, it is set to 0. This measure will be reported as the proportion of patients receiving within this 6-week window, to evaluate the health systems performance.

While the data offers extensive coverage of patient demographics, diagnostics, and treatment information, certain limitations inherent to the study's design and available data must be acknowledged-

1. Patients with missing ATC codes may represent those undergoing surgery, palliative care, or facing treatment delays, potentially skewing the timeliness analysis. To address this, two separate analyses will be conducted: one including the full cohort and another excluding these patients, to ensure a balanced perspective on timeliness.
2. Treatment preferences and options may vary significantly due to differences in patients' health statuses and comorbidities, potentially confounding the analysis of treatment timeliness. Including the Charlson Comorbidity Index (CCI) score, derived from diagnosis history, as a variable in the analysis can allow for future stratifications.
3. Sex and income can play a significant role in treatment access and timeliness. By including and examining the distribution of these variables in the dataset, patterns of disparity or variability can be identified for further subgroup analysis.

Plans for Validity Testing

1. **Face Validity:** Use PROC PRINT to inspect cases where **dti** exceeds 12 weeks. This helps identify outliers that might skew results, such as potential surgical intervention before chemotherapy or data entry errors.
2. **Construct Validity:** Check that the rate of CRC diagnosis in the dataset aligns with expected epidemiological trends. For example, CRC incidence is generally higher in older adults (ages 50+)[8]. Use PROC FREQ to calculate the frequency of CRC diagnoses by age group and compare it to known incidence rates.
3. **Predictive Validity:** Use the **within_time** variable (treatment within 6 weeks) as a predictor for survival outcomes. Survival data can be collected using the **coverage_end** and **death** variables. By incorporating these variables into PROC PHREG, it can be determined whether treatment timeliness effectively predicts longer-term survival. This analysis is limited to patients with survival data available for at least 2 years after treatment initiation.

References

[1] Agency for Healthcare Research and Quality. Six domains of healthcare quality. https://www.ahrq.gov/talkingquality/measures/six-domains.html

[2] Canadian Institute for Health Information. More surgeries being done, but wait times are still long. Accessed November 30, 2024.

[3] Lee L, Cheung WY, Atkinson E, Krzyzanowska MK. Impact of delayed cancer treatment on mortality: A systematic review and meta-analysis. BMJ. 2020;371 https://doi.org/10.1136/bmj.m4087

[4] Khorana, A. A., Tullio, K., Elson, P., Pennell, N. A., Grobmyer, S. R., & et al. (2019). Time to initial cancer treatment in the United States and association with survival over time: An observational study. *PLOS ONE, 14*(3), e0213209. https://doi.org/10.1371/journal.pone.0213209

[5] Government of British Columbia. (n.d.). *Diagnostic code descriptions (ICD-9) neoplasms*. https://www2.gov.bc.ca/assets/gov/health/practitioner-pro/medical-services-plan/diag-codes_neoplasms.pdf

[6] World Health Organization. (2016). *International statistical classification of diseases and related health problems 10th revision (ICD-10)*. https://icd.who.int/browse10/2016/en#/II

[7] Cone, E. B., Marchese, M., Paciotti, M., & et al. (2020). Assessment of time-to-treatment initiation and survival in a cohort of patients with common cancers. *JAMA Network Open, 3*(12), e2030072. https://doi.org/10.1001/jamanetworkopen.2020.30072

[8] O'Donnell, C. D. J., Hubbard, J., & Jin, Z. (2024). Updates on the management of colorectal cancer in older adults. *Cancers, 16*(10), 1820. https://doi.org/10.3390/cancers16101820

Michelle Machado

Metadata

November 30th, 2024

Table of Variables

Variable Name	Description	Format
id	Unique patient identifier. Derived from sd_table_demo_rev. Used to link across all datasets (sd_table_diag, sd_table_drug, etc.).	Numeric
sex	Patient's sex (1=male, 0=female). Derived from sd_table_demo_rev.	Binary
age	Patient age at the time of CRC diagnosis. Calculated using birthdate (from sd_table_demo_rev) and diagnosis date (from sd_table_diag).	Continuous
income	Patient's income decile at the time of diagnosis (1=lowest, 10=highest). Derived from sd_table_demo_rev.	Categorical
start	Start date of healthcare coverage. Derived from sd_table_demo_rev to get healthcare coverage history.	Date
end	End date of healthcare coverage. Derived from sd_table_demo_rev	Date
death	Dead (1=dead, 0=alive). Derived from sd_table_demo_rev.	Binary
dx	Patients CRC diagnoses. Patients will be recognized using diagnosis codes for colorectal cancer (ICD10: C18- C20; ICD9: 153-154 for malignant neoplasms of colon, rectosigmoid junction, and rectum)[1]. Derived from sd_table_diag.	Categorical
crc_date	Date of colorectal cancer diagnosis. Derived from sd_table_diag.	Date
atc	Treatment received for CRC. Following antineoplastic agent codes are used to identify CRC treatments- (L01BA04, L01BC02, L01BC06, L01BC59, L01XA03, L01XC07, L01XC08, L01XC11, L01XC17, L01XX19)[2,3]. Derived from sd_table_drug.	Categorical
rx_date	Start date of CRC treatment. Derived from sd_table_drug.	Date
cci	Charlson Comorbidity Index score calculated for comorbid conditions prior to CRC diagnosis. Comorbidities will be identified using ICD codes for Charlson Comorbidities[4] (e.g., rheumatic disease, myocardial infarction, renal disease, etc.). Will use macro developed by National Cancer Institute.[5]	Continuous
dti	Time (in days) between diagnosis and treatment initiation. Calculated by subtracting crc_date from rx_date.	Continuous
within_time	Whether treatment was initiated within 6 weeks of diagnosis (1=Yes, 0=No). Derived by comparing the dti variable to the 6-week threshold. This is a binary variable indicating whether the patient received timely treatment.	Binary

References

1. Sun, J., et al. (2023). Statin use and risk of colorectal cancer in patients with inflammatory bowel disease. *eClinicalMedicine*, 63, 102182. https://doi.org/10.1016/j.eclinm.2023.102182
2. Canadian Cancer Society. (n.d.). *Chemotherapy for colorectal cancer*. https://www.cancer.ca/en/cancer-information/cancer-types/colorectal/treatment/chemotherapy
3. Ma, B., King, A. D., Leung, L., Wang, K., Poon, A., Ho, W. M., Mo, F., Chan, C. M. L., Chan, A. T. C., & Wong, S. C. C. (2017). Identifying an early indicator of drug efficacy in patients with metastatic colorectal cancer—a prospective evaluation of circulating tumor cells, 18F-fluorodeoxyglucose positron-emission tomography and the RECIST criteria. *Annals of Oncology*, 28(7), 1576-1581. https://doi.org/10.1093/annonc/mdx149
4. University of Manitoba. (2024). ICD-9-CM and ICD-10 coding algorithms for Charlson comorbidities. Retrieved from http://mchp-appserv.cpe.umanitoba.ca/concept/Charlson%20Comorbidities%20-%20Coding%20Algorithms%20for%20ICD-9-CM%20and%20ICD-10.pdf
5. Healthcare Delivery Research Program. (2021). *SEER-Medicare: Comorbidity SAS macro (2021 version)*. National Cancer Institute. Retrieved October 11, 2024, from https://healthcaredelivery.cancer.gov/seermedicare/considerations/NCI.comorbidity.macro.sas

Michelle Machado
261085013

SAS code

```
/*************************************************************************************************
*****/
/* Log Information */
/* Project Name: CRC Diagnosis Data Preparation */
/* Author: Michelle Machado */
/* Date: November 1st, 2024 */
/* Purpose: Filter and prepare data to assess timely CRC treatment */
/* Last Updated: November 4th, 2024 */
/*************************************************************************************************
*****/

/* Libname statements */
libname mydata '/home/u64020049/my_shared_file_links/u45035527';
libname myset '/home/u64020049/sasuser.v94';

/*************************************************************************************************
*****/
/* Filter patients with CRC diagnosis from sd_table_diag */
data diag;
    set mydata.sd_table_diag;
run;

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from diag;
quit; /*n = 993953*/

/* Keep only records with CRC diagnoses based on ICD codes */
data diag_crc;
    set diag;
```

```
    if dx in ('C18', 'C19', 'C20', '153', '154');
run;

/*Rename date column*/
data diag_crc;
    set diag_crc(rename=(date= crc_date));
run;

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from diag_crc;
quit; /*n = 13670*/

/* Sort by patient ID and crc date */
proc sort data=diag_crc;
    by id crc_date;
run;

/* Keep only the first diagnosis for each patient */
data diag_crc;
    set diag_crc;
    by id;
    if first.id;
run;

/*************************************************************************************/
/*Join with death data to get coverage dates*/
data coverage;
    set mydata.sd_table_death;
    keep id start end death;
run;
```

```
/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from coverage;
quit; /*n = 1000000*/

proc sql;
    create table diag_crc_start_end as
    select a.*, b.start, b.end, b.death
    from diag_crc as a
    left join coverage as b
    on a.id = b.id;
quit;

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from diag_crc_start_end;
quit; /*n = 13670*/

/*************************************************************************************
*************************/
/* Remove patients whose start is less than 2 years before crc_date and whose diag_date is not >42
days before end */
data diag_crc_final;
    set diag_crc_start_end;

    /* Apply both conditions */
    if intck('year', start, crc_date) >= 2 and crc_date <= intnx('day', end, -42);
run;

/* Check unique IDs */
proc sql;
```

```
    select count(distinct id) as unique_ids
    from diag_crc_final;
quit; /*n = 9981*/

/*************************************************************************************
************************/
/*Join with drug dataset*/

data drug;
    set myset.machado_atc; /*replace with the dataset shared with you*/
run; /*n = 8149*/

/*The next code is to output my required atc codes from the bigger drug dataset
  But since I have received my datacut the number of unique ids remains the same in both*/
data drug;
    set drug(rename=(date=rx_date));
    if atc in ('L01BA04', 'L01BC02', 'L01BC06', 'L01BC59',
               'L01XA03', 'L01XC07', 'L01XC08', 'L01XC11',
               'L01XC17', 'L01XX19');
run;

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from drug;
quit; /*n = 8149*/

/*************************************************************************************
************************/
/*Joining diag data*/
proc sql;
    create table drug_diag as
    select a.*, b.atc, b.rx_date
```

```
    from diag_crc_final as a
    left join drug as b
    on a.id = b.id;
quit;

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from drug_diag;
quit; /*n = 9981*/

/*************************************************************************************************
************************/
/* Filter to keep only instances where rx_date is after crc_date */
data drug_diag;
    set drug_diag;
    where missing(rx_date) or rx_date > crc_date;
run;

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from drug_diag;
quit; /*n = 9825*/

proc sort data=drug_diag;
    by id rx_date;
run;

/*Keep the rx_date closest to crc_date for each patient
*****************************************/
data drug_diag_final;
    set drug_diag;
```

```
by id;

/* If rx_date is missing, output the row as-is */
if missing(rx_date) then do;
    output;
    return;
end;

/* Calculate the difference in days between rx_date and crc_date */
days_diff = abs(rx_date - crc_date);

/* Retain the closest rx_date for each patient */
retain closest_rx_date closest_atc min_days_diff;

/* Reset at the start of each patient */
if first.id then do;
    min_days_diff = days_diff;
    closest_rx_date = rx_date;
    closest_atc = atc;
end;

/* Update if a closer rx_date is found */
else if days_diff < min_days_diff then do;
    min_days_diff = days_diff;
    closest_rx_date = rx_date;
    closest_atc = atc;
end;

/* Output only the record with the closest rx_date for each patient */
if last.id then do;
    rx_date = closest_rx_date;
    atc = closest_atc;
    output;
```

```
    end;
    drop days_diff min_days_diff closest_rx_date closest_atc;
run; /*n=9825*/

/*************************************************************************************
************************/
/*Remove patients that were diagnosed with other cancers before their CRC diagnosis*/

proc sql;
    create table patients_no_prior_cancer as
    select distinct a.*
    from drug_diag_final as a
    left join diag as b
    on a.id = b.id
    where not (
        /* Check for ICD-10 codes starting with 'C' or 'D' */
        (substr(b.dx, 1, 1) in ('C', 'D'))
        or
        /* Check for ICD-9 codes in the range 140-239 */
        (b.dx between '140' and '239')
    )
    and b.date < a.crc_date;
quit;

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from patients_no_prior_cancer;
quit; /*n = 9738*/

/*************************************************************************************
************************/
```

```
/* Remove patients diagnosed with other cancer between their CRC diagnosis and first treatment
initiation */
proc sql;
    create table final_drug_diag as
    select distinct a.*
    from patients_no_prior_cancer as a
    left join diag as b
    on a.id = b.id
    where not (
        /* Check for ICD-10 codes starting with 'C' or 'D' */
        (substr(b.dx, 1, 1) in ('C', 'D'))
        or
        /* Check for ICD-9 codes in the range 140-239 */
        (b.dx between '140' and '239')
    )
    and b.date > a.crc_date
    and b.date < coalesce(a.rx_date, a.end); /* Use end if rx_date is missing */
quit;

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from final_drug_diag;
quit; /* n = 9046 */

/*************************************************************************************
************************/
/*Join all demo characteristics*/
/*Bring in sex, birthdate and calculate age at diagnosis*/

data age_sex;
    set mydata.sd_table_demo_rev;
    keep id male birthdate;
```

```
run;

proc sort data=age_sex nodupkey;
    by id;
run;

proc sql;
    create table age_sex_crc as
    select a.*, b.male, b.birthdate
    from final_drug_diag as a
    left join age_sex as b
    on a.id = b.id;
quit;

data age_sex_crc;
    set age_sex_crc;
    /* Calculate age in years at diagnosis */
    age = intck('year', birthdate, crc_date)
                      - (month(crc_date) < month(birthdate)
                      or (month(crc_date) = month(birthdate)
                      and day(crc_date) < day(birthdate)));
                      drop birthdate;
run;

/*************************************************************************************
************************/
/*Add income data*/
data income;
    set mydata.sd_table_demo_rev;
    keep id start end income;
run;

proc sql;
```

```
    create table crc_demo as
    select a.*, b.start as new_start, b.end as new_end, b.income
    from age_sex_crc as a
    left join income as b
    on a.id = b.id;
quit;

/*Keep income at diagnosis*/
data crc_demo_final;
    set crc_demo;
    /* Keep only rows where crc_date falls within the new_start and new_end date range */
    if crc_date >= new_start and crc_date <= new_end;
    drop new_start new_end;
run;

/*************************************************************************************
************************/
/*Remove patients less than 18 years at diagnosis*/
data crc_demo_adults;
    set crc_demo_final;
    /* Keep only patients who are 18 years or older at diagnosis */
    if age >= 18;
run; /*n=9041*/

/* Check unique IDs */
proc sql;
    select count(distinct id) as unique_ids
    from crc_demo_adults;
quit; /* n = 9041 */

/*************************************************************************************
************************/
/*Make comorbidity score*/
```

```
data comorbidity;
    set mydata.sd_table_diag;
run;

proc sql;
    create table crc_comorbidity as
    select a.id, a.crc_date, b.dx, b.date
    from crc_demo_adults as a
    left join comorbidity as b
    on a.id = b.id
    where b.date < a.crc_date;
quit;

/*************************************************************************************************
************************/
/*USING THE CCI MACRO DEVELOPED BY NCI*/
/*Modified to fit this datset*/
%macro Simple_Comorbidity_Score(DATASET=crc_comorbidity, OUTFILE=Comorbidity_Score);

/* Define comorbid conditions */
%let conditions = acute_mi history_mi chf pvd cvd copd dementia paralysis diabetes diabetes_comp
renal_disease
                  mild_liver_disease liver_disease ulcers rheum_disease aids;

data comorbidity_flags;
    set &DATASET;
    by id;

    /* Initialize comorbidity flags */
    array comorb_flags[*] acute_mi history_mi chf pvd cvd copd dementia paralysis diabetes
diabetes_comp renal_disease
                        mild_liver_disease liver_disease ulcers rheum_disease aids;
    do i = 1 to dim(comorb_flags);
```

```
        comorb_flags[i] = 0;
    end;

    /* Determine comorbidities based on ICD codes using substr function */
    if substr(dx, 1, 3) = '410' or dx = 'I21' or dx = 'I22' then acute_mi = 1;               /*
Acute MI */
    if substr(dx, 1, 3) = '412' or dx = 'I252' then history_mi = 1;                          /*
History MI */
    if substr(dx, 1, 3) = '428' or dx = 'I50' or dx in ('I099', 'I110', 'I130', 'I132', 'I255') then
chf = 1;  /* CHF */
    if substr(dx, 1, 3) = '440' or dx in ('I70', 'I71', '0930', 'V434', '5571') then pvd = 1; /* PVD
*/
    if substr(dx, 1, 3) = '430' or dx = 'G45' or substr(dx, 1, 2) = 'I6' then cvd = 1;       /* CVD
*/
    if substr(dx, 1, 3) = '490' or dx = 'J40' then copd = 1;                                 /* COPD
*/
    if substr(dx, 1, 3) = '290' or dx in ('F051', 'G30') then dementia = 1;                  /*
Dementia */
    if substr(dx, 1, 4) = '3341' or dx in ('G81', 'G82', '342', 'G041') then paralysis = 1;  /*
Paralysis */
    if (substr(dx, 1, 3) = '250' or dx in ('E10', 'E11')) and not (dx = '2504' or dx = 'E14') then
diabetes = 1; /* Diabetes */
    if dx in ('2504', '2505', 'E10', 'E11') then diabetes_comp = 1;                          /*
Diabetes Complications */
    if substr(dx, 1, 3) = '585' or dx in ('N18', 'N19') then renal_disease = 1;              /*
Renal Disease */
    if substr(dx, 1, 3) = '070' or dx in ('K73', 'K74') then mild_liver_disease = 1;         /* Mild
Liver Disease */
    if substr(dx, 1, 4) = '4560' or substr(dx, 1, 4) = '5722' or dx = 'K70' then liver_disease = 1;
/* Severe Liver Disease */
    if substr(dx, 1, 3) = '531' or substr(dx, 1, 3) = 'K25' then ulcers = 1;                 /*
Peptic Ulcer Disease */
```

```
    if substr(dx, 1, 3) = '714' or dx in ('M05', 'M06', 'M32') then rheum_disease = 1;       /*
Rheumatic Disease */
    if substr(dx, 1, 3) = '042' or dx in ('B20', 'B21', 'B22') then aids = 1;                /* AIDS
*/

    /* Only keep distinct patient ID with comorbidity flags */
    if last.id;
    keep id acute_mi history_mi chf pvd cvd copd dementia paralysis diabetes diabetes_comp
renal_disease
         mild_liver_disease liver_disease ulcers rheum_disease aids;
run;

/* Calculate Charlson score */
data &OUTFILE;
    set comorbidity_flags;

    /* Calculate Charlson score */
    cci =
        1 * (acute_mi or history_mi) +
        1 * (chf) +
        1 * (pvd) +
        1 * (cvd) +
        1 * (copd) +
        1 * (dementia) +
        2 * (paralysis) +
        1 * (diabetes and not diabetes_comp) +
        2 * (diabetes_comp) +
        2 * (renal_disease) +
        1 * (mild_liver_disease and not liver_disease) +
        3 * (liver_disease) +
        1 * (ulcers) +
        1 * (rheum_disease) +
        6 * (aids);
```

```
    /* Keep final output */
    keep id cci acute_mi history_mi chf pvd cvd copd dementia paralysis diabetes diabetes_comp
renal_disease
         mild_liver_disease liver_disease ulcers rheum_disease aids;
run;

%mend Simple_Comorbidity_Score;

/* Run the macro with the specified dataset */
%Simple_Comorbidity_Score(DATASET=crc_comorbidity, OUTFILE=comorbidity_score);

/*************************************************************************************
************************/
/*Join comorbidity score with crc data*/
proc sql;
    create table crc_cohort as
    select a.*, b.cci
    from crc_demo_adults as a
    left join comorbidity_score as b
    on a.id = b.id;
quit;

/*************************************************************************************
************************/
/*Final measure*/
data crc_cohort;
    set crc_cohort;

    /* Calculate DTI (diagnosis-to-treatment interval) in days */
    dti = rx_date - crc_date;
```

```
    /* Create within_time column: 1 if DTI is <= 42 days, else 0. If DTI is missing, set within_time
to 0 */
    if missing(dti) then within_time = 0;
    else within_time = (dti <= 42);
run;

/*************************************************************************************************
************************/
/*VALIDITY CHECKS*****************************************************************/

/*Looking at age at diagnosis trend in the cohort*************************************/
/* Create age groups in a new dataset */
data crc_age_groups;
    set crc_cohort;
    length age_group $5;  /* Adjusting to ensure space for the longest string "70-79" */
    /* Define age groups */
    if age < 50 then age_group = '<50';
    else if age >= 50 and age < 60 then age_group = '50-59';  /* Corrected the logic */
    else if age >= 60 and age < 70 then age_group = '60-69';
    else if age >= 70 and age < 80 then age_group = '70-79';
    else age_group = '80+';
run;

/* Calculate frequency of CRC diagnoses by age group */
proc freq data=crc_age_groups;
    tables age_group ;
    title "Frequency of CRC Diagnoses by Age Group";
run;
/*17% patients are <50 years old. This is in line with the epidemiology of CRC*/

/*Patients with DTI > 12
weeks***************************************************************************/
proc print data=crc_cohort;
```

```
    where dti > 84;
    var id crc_date rx_date dti within_time;
    title "Cases with DTI Exceeding 12 Weeks (Potential Outliers)";
run;

/*Make datsets with DTI > 12 weeks and missing dti*/
data dti_over_12weeks dti_missing;
    set crc_cohort;

    /* Output records with DTI > 12 weeks to dti_over_12weeks */
    if dti > 84 then output dti_over_12weeks;

    /* Output records with missing DTI to dti_missing */
    else if missing(dti) then output dti_missing;
run;

/*Note: We might want to remove these patients from the denominator
  while calculating proportion within time
  As we can assume that these cases probably received radiotherapy/surgery prior to chemo,
  or have no chemo data as they were ineligible or did not choose to take chemo/*

/*Create a combined dataset with IDs to exclude from the validity check*/
data exclude_ids;
    set dti_over_12weeks dti_missing;
    keep id;
run;

/*Remove these IDs from crc_cohort */
proc sql;
    create table crc_cohort_excl as
    select *
    from crc_cohort
    where id not in (select id from exclude_ids);
```

```
quit;

/*************************************************************************************
************************/
/*Evaluating timeliness of the healthcare system*/

/*Depending on whether or not exclusion is assumed the measure can
use the crc_cohort or crc_cohort_excl datasets*/
/* Summary statistics for the full cohort */
proc sql;
    title "Summary Statistics: Proportion of Patients Receiving Treatment Within 6 Weeks (Full
Cohort)";
    select within_time,
           count(*) as n format=8.,
           (count(*) * 100 / (select count(*) from crc_cohort)) as percent format=8.2
    from crc_cohort
    group by within_time;
quit;

/* Summary statistics for the cohort after exclusion */
proc sql;
    title "Summary Statistics: Proportion of Patients Receiving Treatment Within 6 Weeks (After
Exclusion)";
    select within_time,
           count(*) as n format=8.,
           (count(*) * 100 / (select count(*) from crc_cohort_excl)) as percent format=8.2
    from crc_cohort_excl
    group by within_time;
quit;

/*************************************************************************************
************************/
/*Checking predictive validity of the measure within time*/
```

```
/* Create a variable to indicate survival time capped at 2 years (730 days) */
/* Include all patients in the survival analysis
************************************************/
data crc_cohort_full_2yr_survival;
    set crc_cohort;

    /* Calculate survival time for patients with rx_date */
    if not missing(rx_date) then do;
        if end - rx_date > 730 then survival_time_2yr = 730;
        else survival_time_2yr = end - rx_date;
    end;
    /* For patients without rx_date, use coverage_end to calculate survival time */
    else do;
        survival_time_2yr = end - crc_date;
        if survival_time_2yr > 730 then survival_time_2yr = 730;
    end;

    /* Create an event indicator for death within 2 years */
    death_2yr = (death = 1 and survival_time_2yr <= 730);
run;

/* Conduct survival analysis for 2-year survival using PROC PHREG on the full cohort*/
proc phreg data=crc_cohort_full_2yr_survival;
    class within_time (ref='0');

    /* Specify survival time capped at 2 years and 2-year death indicator */
    model survival_time_2yr * death_2yr(0) = within_time / ties=efron;

    /* Hazard Ratio for treatment timeliness */
    hazardratio 'Within 6 Weeks vs. After 6 Weeks' within_time;
    title "2-Year Survival Analysis: Effect of Treatment Timeliness on Survival Outcomes (Full
Cohort)";
run;
```

```
/*The hazard ratio of 0.404 for within_time suggests that patients who received timely treatment
 have a 59.6% lower hazard of dying within 2 years compared to those who received
 treatment after 6 weeks. However, this difference is not statistically significant, as indicated
 by the high p-value (0.3646).*/

/* Not including all patients in the survival analysis
*************************************************/
data crc_cohort_2yr_survival;
    set crc_cohort_excl;

    /* Calculate time from treatment initiation (rx_date) to either 2 years or coverage_end */
    if end - rx_date > 730 then survival_time_2yr = 730;
    else survival_time_2yr = end - rx_date;

    /* Create an event indicator for death within 2 years */
    death_2yr = (death = 1 and (end - rx_date) <= 730);
run;

/*Conduct survival analysis for 2-year survival using PROC PHREG after exclusion*/
proc phreg data=crc_cohort_2yr_survival;
    class within_time (ref='0'); /* Reference group is '0' (treatment not within 6 weeks) */

    /* Specify survival time capped at 2 years and 2-year death indicator */
    model survival_time_2yr * death_2yr(0) = within_time / ties=efron;

    /* Hazard Ratio for treatment timeliness */
    hazardratio 'Within 6 Weeks vs. After 6 Weeks' within_time;
    title "2-Year Survival Analysis: Effect of Treatment Timeliness on Survival Outcomes (After
Exclusion)";
run;

/*The hazard ratio of 0.749 for within_time suggests that patients who received timely treatment
```

```
 have a 25.1% lower hazard (or risk) of dying within 2 years compared to those who received
treatment
 after 6 weeks. However, this difference is not statistically significant, as shown by the high
 p-value (0.8021). probably due to the low sample size and other rescrictions of the data*/

/*************************************************************************************************
************************/
/*Looking at distributipn of patients in the within time vs not within time categories*/
/* Create categories based on CCI scores full cohort*/
data crc_cohort_categorized;
    set crc_cohort;
    length cci_category $8;
    if cci >= 1 and cci <= 2 then cci_category = 'Mild';
    else if cci >= 3 and cci <= 4 then cci_category = 'Moderate';
    else if cci >= 5 then cci_category = 'Severe';
    else cci_category = 'None'; /* For CCI scores of 0 or missing */
run;

/* Frequency distribution for visualization */
proc report data=crc_cohort_categorized nowd;
    column cci_category within_time, (n pctn);
    define cci_category / group 'CCI Category';
    define within_time / across 'Within Time (Full Cohort)' format=within_time_fmt.;
    define n / 'Count' f=comma8.;
    define pctn / 'Percent' f=percent8.2;

    /* Compute custom combined 'n(%)' */
    compute pctn;
        pctn = cats(put(n, comma8.), ' (', put(pctn, percent8.2), ')');
    endcomp;
    rbreak after / summarize;
    where within_time in (0, 1);
    rbreak after / summarize style=[font_weight=bold];
```

```
    title "Distribution of Within Time by CCI";
run;

/*****************************************************************************************
************************/
/* Create categories based on CCI scores after exclusion*/
data crc_cohort_categorized;
    set crc_cohort_excl;
    length cci_category $8;
    if cci >= 1 and cci <= 2 then cci_category = 'Mild';
    else if cci >= 3 and cci <= 4 then cci_category = 'Moderate';
    else if cci >= 5 then cci_category = 'Severe';
    else cci_category = 'None'; /* For CCI scores of 0 or missing */
run;

/* Frequency distribution for visualization */
proc report data=crc_cohort_categorized nowd;
    column cci_category within_time, (n pctn);
    define cci_category / group 'CCI Category';
    define within_time / across 'Within Time (After Exclusion)' format=within_time_fmt.;
    define n / 'Count' f=comma8.;
    define pctn / 'Percent' f=percent8.2;

    /* Compute custom combined 'n(%)' */
    compute pctn;
        pctn = cats(put(n, comma8.), ' (', put(pctn, percent8.2), ')');
    endcomp;
    rbreak after / summarize;
    where within_time in (0, 1);
    rbreak after / summarize style=[font_weight=bold];
    title "Distribution of Within Time by CCI";
run;
```

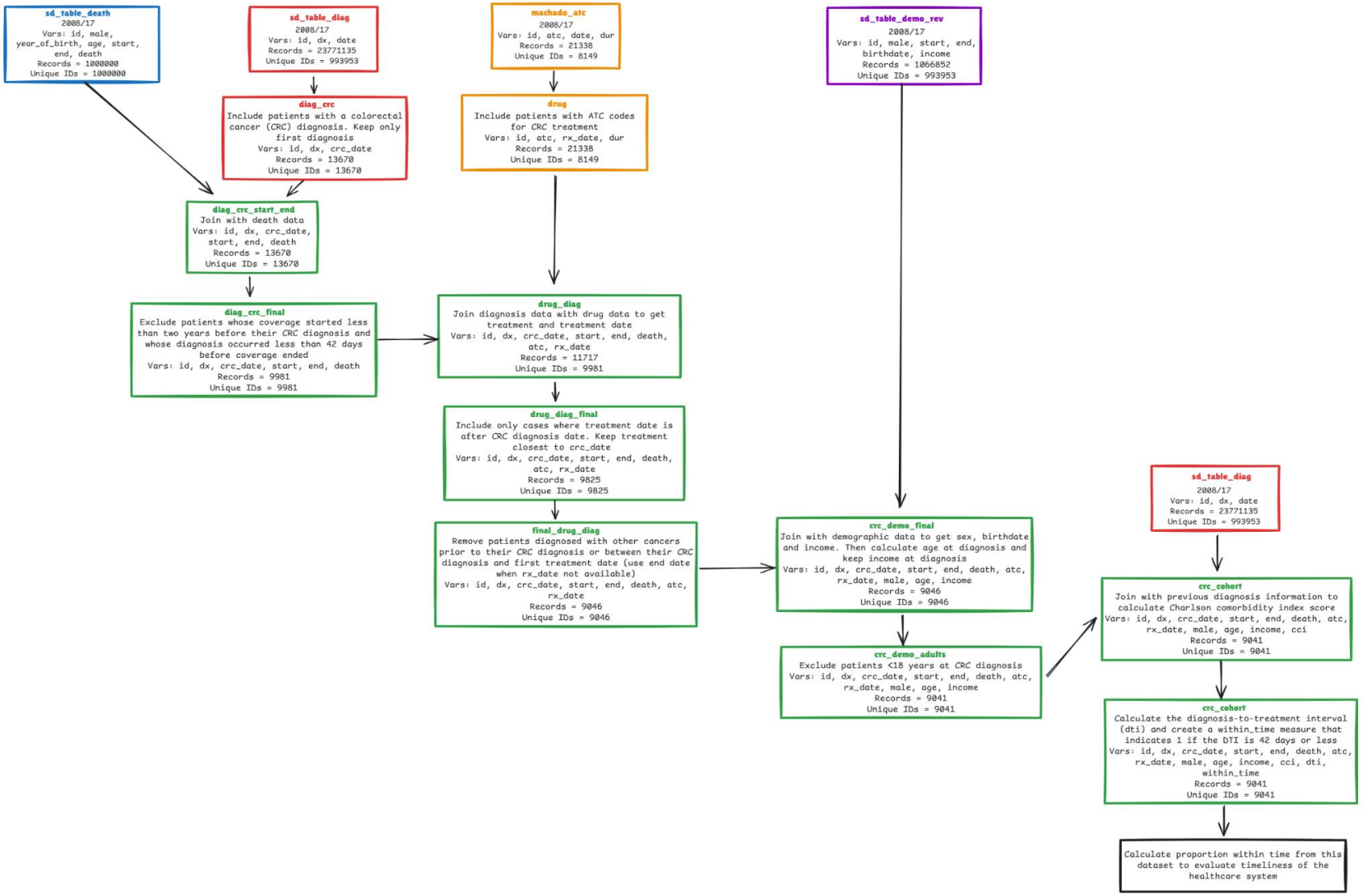
```
/*************************************************************************************************
************************/
/* Create categories based on sex from the 'male' variable using full cohort*/
data crc_cohort_categorized_sex;
    set crc_cohort;
    length sex_category $6;  /* Ensure enough space for category names */

    /* Categorize sex based on the 'male' variable */
    if male = 1 then sex_category = 'Male';
    else if male = 0 then sex_category = 'Female';
    else sex_category = 'Other'; /* This is optional, only if there are missing/other values */
run;

/* Frequency distribution for visualization based on sex */
proc report data=crc_cohort_categorized_sex nowd;
    column sex_category within_time, (n pctn);
    define sex_category / group 'Sex Category' width=8;
    define within_time / across 'Within Time (Full Cohort)' format=within_time_fmt.;
    define n / 'Count' f=comma8.;
    define pctn / 'Percent' format=percent8.2;

    /* Compute custom combined 'n(%)' */
    compute pctn;
        pctn = cats(put(n, comma8.), ' (', put(pctn, percent8.2), ')');
    endcomp;

    where within_time in (0, 1);
    rbreak after / summarize style=[font_weight=bold];
    title "Distribution of Within Time by Sex";
run;

/*************************************************************************************************
************************/
```

```
/* Create categories based on sex from the 'male' variable using excluded cohort*/
data crc_cohort_categorized_sex;
    set crc_cohort_excl;
    length sex_category $6;  /* Ensure enough space for category names */

    /* Categorize sex based on the 'male' variable */
    if male = 1 then sex_category = 'Male';
    else if male = 0 then sex_category = 'Female';
    else sex_category = 'Other'; /* This is optional, only if there are missing/other values */
run;

/* Frequency distribution for visualization based on sex */
proc report data=crc_cohort_categorized_sex nowd;
    column sex_category within_time, (n pctn);
    define sex_category / group 'Sex Category' width=8;
    define within_time / across 'Within Time (After Exclusion)' format=within_time_fmt.;
    define n / 'Count' f=comma8.;
    define pctn / 'Percent' format=percent8.2;

    /* Compute custom combined 'n(%)' */
    compute pctn;
        pctn = cats(put(n, comma8.), ' (', put(pctn, percent8.2), ')');
    endcomp;

    where within_time in (0, 1);
    rbreak after / summarize style=[font_weight=bold];
    title "Distribution of Within Time by Sex";
run;

/*************************************************************************************
************************/
/* Create categories based on income levels full cohort */
data crc_cohort_categorized_income;
```

```
    set crc_cohort;
    length income_category $8;  /* Ensure enough space for category names */

    /* Categorize income based on the given scale */
    if income >= 1 and income <= 3 then income_category = 'Low';
    else if income >= 4 and income <= 6 then income_category = 'Moderate';
    else if income >= 7 and income <= 10 then income_category = 'High';
    else income_category = 'Undefined'; /* Include this to handle any unexpected values or missing
data */
run;

/* Frequency distribution for visualization based on income */
proc report data=crc_cohort_categorized_income nowd;
    column income_category within_time, (n pctn);
    define income_category / group 'Income Category' width=10;
    define within_time / across 'Within Time (Full Cohort)' format=within_time_fmt.; /* You may need
to define this format */
    define n / 'Count' f=comma8.;
    define pctn / 'Percent' format=percent8.2;

    /* Compute custom combined 'n(%)' */
    compute pctn;
        pctn = cats(put(n, comma8.), ' (', put(pctn, percent8.2), ')');
    endcomp;

    where within_time in (0, 1);
    rbreak after / summarize style=[font_weight=bold];
    title "Distribution of Within Time by Income Level";
run;

/*************************************************************************************************
************************/
/* Create categories based on income levels excluded cohort */
```

```
data crc_cohort_categorized_income;
    set crc_cohort_excl;
    length income_category $8;  /* Ensure enough space for category names */

    /* Categorize income based on the given scale */
    if income >= 1 and income <= 3 then income_category = 'Low';
    else if income >= 4 and income <= 6 then income_category = 'Moderate';
    else if income >= 7 and income <= 10 then income_category = 'High';
    else income_category = 'Undefined'; /* Include this to handle any unexpected values or missing
data */
run;

/* Frequency distribution for visualization based on income */
proc report data=crc_cohort_categorized_income nowd;
    column income_category within_time, (n pctn);
    define income_category / group 'Income Category' width=10;
    define within_time / across 'Within Time (After Exclusion)' format=within_time_fmt.; /* You may
need to define this format */
    define n / 'Count' f=comma8.;
    define pctn / 'Percent' format=percent8.2;

    /* Compute custom combined 'n(%)' */
    compute pctn;
        pctn = cats(put(n, comma8.), ' (', put(pctn, percent8.2), ')');
    endcomp;

    where within_time in (0, 1);
    rbreak after / summarize style=[font_weight=bold];
    title "Distribution of Within Time by Income Level";
run;

================================================================================================
---next page---
```

Data Preparation Flowchart to assess Timeliness of the Healthcare System



----next page----

EPIB675 (Use McGill ID to for this link)
This folder has-

1. Data cut from the sd_trug_table data (machado_atc.sas7bdat)
2. SAS code file (EPIB675-Machado-code.sas)