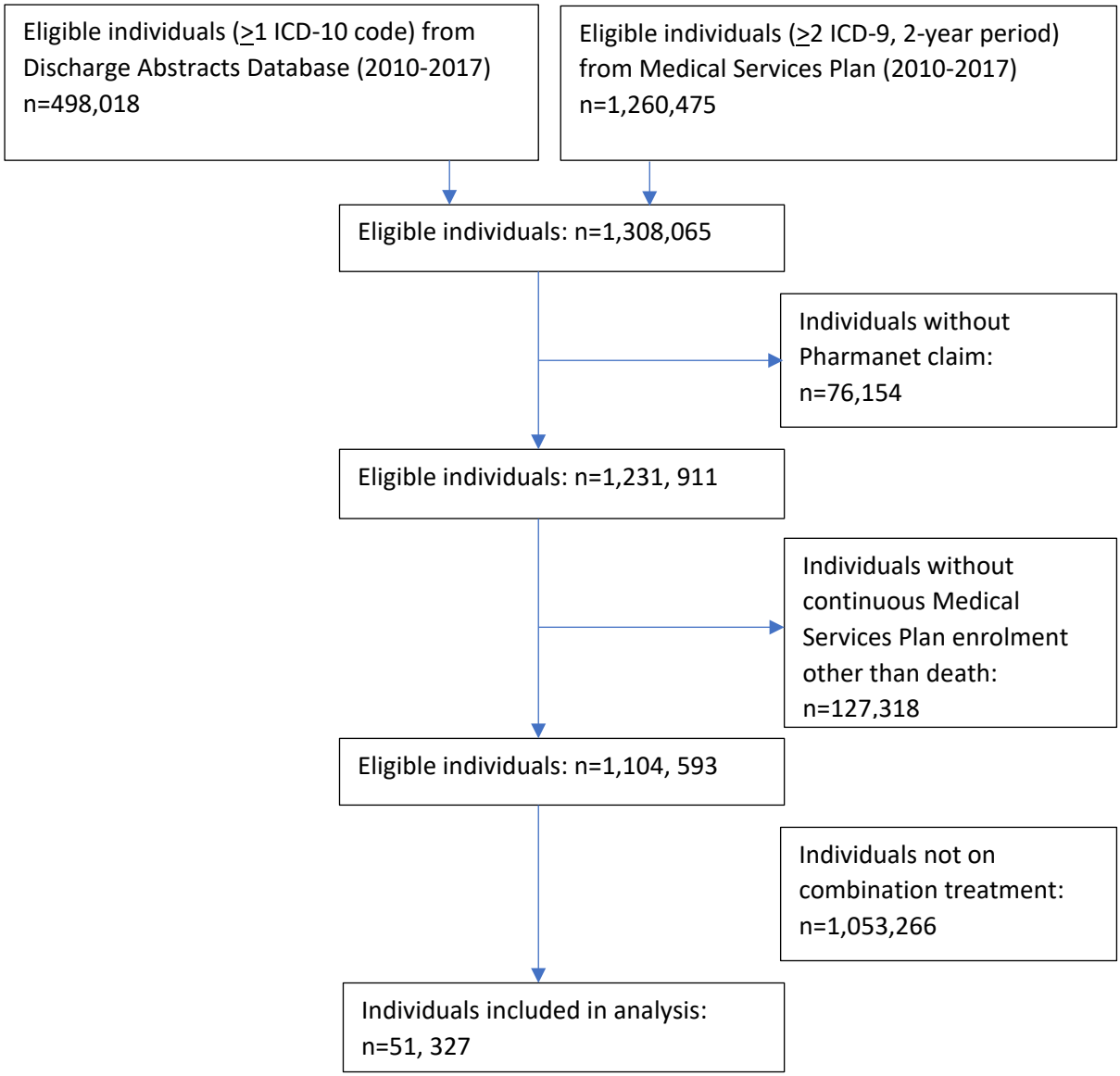
**Supplementary Appendix**

**Table S1 Administrative data sources used**

| Dataset | Data included | Use in this study | Specific item codes used |
|---|---|---|---|
| Medical Services Plan Payment Information File | Data on medically necessary services provided by fee-for-service practitioners to individuals covered by the Medical Services Plan (MSP), BC's universal insurance program. | To define study population | (ICD-9: 401; 402; 403; 404; 405; 250.4*; 428; 250 AND (580 OR 581 OR 582 OR 583 OR 585 OR 586 OR 592 OR 593.3 OR 584; ICD-10: I10; I11; I12; I13; I15; E10.2*; E11.2*; E13.2*; I50, E10-14 AND (N00-N23)) |
| Discharge Abstract Database (hospital separations) | Data on discharges, transfers and deaths of in-patients and day surgery patients from acute care hospitals in BC. | To define study population | |
| PharmaNet | An online, real-time system that captures all prescriptions for drugs and medical supplies dispensed from community pharmacies in BC as well as prescriptions dispensed from hospital outpatient pharmacies for patient use at home. | To track utilisation of ACEI and ARB. | Anatomical Therapeutic Classification codes C09A-D. |
| Medical Services Plan Registry File | Basic demographic information about individuals covered by the Medical Services Plan (MSP), BC's universal insurance program, and registration data. | Demographic information | N/A |
| BC Vital Events and Statistics: Deaths | Includes all deaths registered in the province of BC. | To define the population | N/A |
| College of Physicians and Surgeons of British Columbia | Physician speciality, age, year and place of graduation | To identify prescriber speciality | N/A |

**Supplementary Appendix**

**Figure S1: Flowchart of participants**



**R-Code for weighting**

```
library(xlsx)
library(nlme)
library(lattice)
library(tseries)

data <- read.xlsx("Data.xlsx",sheetName="Overall")

# Setup phase-in periods of 2 months
modeldata <- data[c(1:24,27:49),]
modeldata$trend[25:47] <- modeldata$trend[25:47] - 2
modeldata

modeldata["outcome"]<-NA
modeldata$outcome<-modeldata$mean_combo_start
outcome_full <- data$mean_combo_start
ylabel <- "Mean Proportion of Combination Treatment initiated (%)"
modeldata$outcome<-ts(modeldata$outcome,start=c(2010,01),end=c(2013,11),frequency=12)


##Check if there is seasonal component
library(fma)
fit <- tbats(modeldata$outcome)
s <- !is.null(fit$seasonal)
s


################################################################################
##########
##seasonally adjusted
Series<-ts(modeldata$outcome,start=c(2010,01),end=c(2013,11),frequency=12)##time series
```

```
m<-decompose(Series)

plot(m)

modeldata$outcome<-Series-m$seasonal

##############################################################################################

##If also consider weighted Least Square

ols<-lm(outcome ~ month + intervention + trend ,weights=W_alive,data=modeldata)



##Check the autocorrelation and partial-autocorrelation functions

##Determine the appropriate lag p in ARMA model

acf(residuals(ols))

acf(residuals(ols),type='partial')

##ACF decays quickly showing starionary

##p=2,Spikes at lag 2,then zero

##Fit the ARMA regression model

##We use the gls function to fit regression model with a variety of correlated-error and non-constant error-variance structures

##If also consider weighted Least Square

model_on <- gls(model =outcome ~ month + intervention + trend ,

        data=modeldata, correlation=corARMA(p=2,form=~month),weights=~W_alive^-1, method="ML")

##Use the likelihood-ratio tests to check whether the parameters of the AR process for the errors are necessary and sufficient

mod0<-update(model_on,correlation=NULL)

mod_bf<-update(model_on,correlation=corARMA(p=1))

mod_af<-update(model_on,correlation=corARMA(p=3))
```

```
anova(model_on,mod0)

anova(model_on,mod_bf)

anova(model_on,mod_af)

##The tests support the AR specification since it always has the lowest AIC


summary(model_on)


##Produce the plot of fitted line with observations and counterfactual line

# Produce the plot, first plotting the raw data points

plot(data$month,outcome_full,

    ylab=ylabel,

    ylim=c(0,max(outcome_full)*1.1),

    xlab="Month",

    pch=20,

    col="pink",

    xaxt="n")



# Add x axis with dates

axis(1, at=c(1:49),lwd=0.75,tck=-.01,labels=rep("",49))

axis(1, at=c(3,15,27,39),tck=-.06,labels=c("","","",""),lwd=1.25)

axis(1, tick=F,at=c(0.5,10.5,24,36,48),labels=c("2012","2013","2014","2015","2016"))



# Add line indicating the policy changes

abline(v=24.5,lty=2)

##text(30, 50, "intervention", col = "green")

text(30, 0, "intervention", col = "green")


# Plot the first line segment

lines(data$month[1:24], fitted(model_on)[1:24], col="red",lwd=2)
```

```
# Plot the second line segment
lines(data$month[27:49], fitted(model_on)[25:47], col="red",lwd=2)



# And the first counterfactual
segments(25, model_on$coef[1]+model_on$coef[2]*25,
      49, model_on$coef[1]+model_on$coef[2]*49,
      lty=2, lwd=2, col='pink')
```