COVID_day25_SL

Whitney Mgbara

4/15/2020

Load Libraries

Import case data and covariates

```
# Case data
day25 <- read_excel("Data/day25.xls")
day25$popland <- day25$Population/(day25$LandArea)

# Covariate Data
analytic_data2020 <- read_csv("Data/analytic_data2020.csv")
colnames(analytic_data2020)[which(names(analytic_data2020) == "State Abbreviation")] <- "State"

eco_vars_of_int <- c("State",
                     "Name",
                     "Poor or fair health raw value",
                     "Adult smoking raw value",
                     "Food environment index raw value",
                     "Physical inactivity raw value",
                     "Excessive drinking raw value",
                     "Sexually transmitted infections raw value",
                     "Primary care physicians raw value",
                     "Flu vaccinations raw value",
                     "High school graduation raw value",
                     "Unemployment raw value",
                     "Air pollution - particulate matter raw value",
                     "Drinking water violations raw value",
                     "Diabetes prevalence raw value",
                     "HIV prevalence raw value",
                     "Food insecurity raw value",
                     "Drug overdose deaths raw value",
                     "Median household income raw value",
                     "% below 18 years of age raw value",
                     "% 65 and older raw value",
                     "% Hispanic raw value",
                     "% Females raw value", "% Rural raw value",
                     "Adult obesity raw value",
                     "Income inequality raw value",
                     "Uninsured adults raw value")

eco_health_covars <- names(analytic_data2020) %in% eco_vars_of_int

covars <- analytic_data2020[eco_health_covars]
```

```r
# Merge case data with covarites
day25_covars <- merge(day25, covars, by = c("Name", "State"))

day25_covars$log_pop_dense <- log(day25_covars$Population/(day25_covars$LandArea))
```

Import Google mobility data

```r
# Data 1
grocery_pharmacy <- read_csv("Data/Mobility/google-mobility-us-groceryAndPharmacy.csv")

df.1 <-
  grocery_pharmacy %>%
  gather(key = date, value = value, -State)
df.1$type <- "grocery_pharmacy"


# Data 2
parks <- read_csv("Data/Mobility/google-mobility-us-parks.csv")

df.2 <-
  parks %>%
  gather(key = date, value = value, -State)
df.2$type <- "parks"


# Data 3
residential <- read_csv("Data/Mobility/google-mobility-us-residential.csv")

df.3 <-
  residential %>%
  gather(key = date, value = value, -State)
df.3$type <- "residential"


# Data 4
retailAndRecreation <- read_csv("Data/Mobility/google-mobility-us-retailAndRecreation.csv")

df.4 <-
  retailAndRecreation %>%
  gather(key = date, value = value, -State)
df.4$type <- "retailAndRecreation"


# Data 5
transitStations <- read_csv("Data/Mobility/google-mobility-us-transitStations.csv")

df.5 <-
  transitStations %>%
  gather(key = date, value = value, -State)
df.5$type <- "transitStations"


# Data 6
workplaces <- read_csv("Data/Mobility/google-mobility-us-workplaces.csv")
```

```r
df.6 <-
  workplaces %>%
  gather(key = date, value = value, -State)
df.6$type <- "workplaces"


mobility.data <-
  left_join(df.1, df.2, by = c("State", "date", "value", "type")) %>%
  left_join(df.3) %>%
  left_join(df.4) %>%
  left_join(df.5) %>%
  left_join(df.6)

mobility_data_long <- rbind(df.1, df.2,df.3,df.4,df.5,df.6)
mobility_data_wide <- spread(mobility_data_long, date, value)
```

Run Simple Triple Interaction Model and get Marginal Effects for Transportation

```r
## set up bootstrap CI function
 bootstrapCI <- function(model, perc, boot_pred_data, type) {
  nr <- nrow(model$data)
  data <- model$data
  new_data <- data[sample(1:nr, size = nr, replace = TRUE), ]
  up <- update(model, data = new_data)

  #norm <- sum(predict(up, newdata = new_data, type = 'response'), na.rm = TRUE)

  boot_pred_data[,"log_Pub_Trans"] <- boot_pred_data$log_Pub_Trans - (boot_pred_data$log_Pub_Trans*perc

  if (type == 'count') {
      perc_red <- sum(predict(up, newdata = boot_pred_data, type = 'response'), na.rm=TRUE)
  } else {
    perc_red <- mean(predict(up, newdata = boot_pred_data, type = 'response'), na.rm=TRUE)
  }

  return(perc_red)
 }

#set up percentiles for trans reduction
percents <- c(0.0, 0.10, 0.20, 0.30, 0.40, 0.50, 0.60, 0.70, 0.80, 0.90)

#log transformation because the data is skewed forday 25 only
day25$log_pop_dense <- log(day25$popland+1)
day25$log_GDP <- log(day25$GDP+1)
day25$log_Pub_Trans <- log(day25$CommutingByPublicTransportation + 1)

#log transformation because the data is skewed forday 25 covars - 8 regions reduced compared to day25 d

day25_covars$log_pop_dense <- log(day25_covars$popland+1)
day25_covars$log_GDP <- log(day25_covars$GDP+1)
day25_covars$log_Pub_Trans <- log(day25_covars$CommutingByPublicTransportation + 1)


#models
```

```r
est_day25_pois_model <- glm(ConfirmedCasesDay25~ log_pop_dense+log_GDP+log_Pub_Trans +
                        log(Population) +
                        `% Females raw value` +
                        `% 65 and older raw value` +
                        `Adult obesity raw value`+
                        `Physical inactivity raw value` +
                        `Unemployment raw value` +
                        `Income inequality raw value` +
                        `Poor or fair health raw value`+
                        `Uninsured adults raw value` +
                        `Adult smoking raw value`,
                       family = "poisson",
                       data = day25_covars,
                       offset(log(Population)))

est_day_snc_t1_pois_model <- glm(DayOfFirstCases~ log_pop_dense+log_GDP+log_Pub_Trans +
                        log(Population) +
                        `% Females raw value` +
                        `% 65 and older raw value` +
                        `Adult obesity raw value`+
                        `Physical inactivity raw value` +
                        `Unemployment raw value` +
                        `Income inequality raw value` +
                        `Poor or fair health raw value`+
                        `Uninsured adults raw value` +
                        `Adult smoking raw value`,
                       family = "gaussian",
                       data = day25_covars)

summary(est_day25_pois_model)
```

```
##
## Call:
## glm(formula = ConfirmedCasesDay25 ~ log_pop_dense + log_GDP +
##     log_Pub_Trans + log(Population) + `% Females raw value` +
##     `% 65 and older raw value` + `Adult obesity raw value` +
##     `Physical inactivity raw value` + `Unemployment raw value` +
##     `Income inequality raw value` + `Poor or fair health raw value` +
##     `Uninsured adults raw value` + `Adult smoking raw value`,
##     family = "poisson", data = day25_covars, weights = offset(log(Population)))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -745.51   -94.18   -34.04    25.02   878.19
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                    -6.324e+00  1.751e-02 -361.12   <2e-16 ***
## log_pop_dense                  -2.770e+01  8.210e-02 -337.41   <2e-16 ***
## log_GDP                        -1.221e-01  7.437e-04 -164.18   <2e-16 ***
## log_Pub_Trans                   3.668e-01  3.064e-04 1197.13   <2e-16 ***
## log(Population)                 3.866e-01  9.184e-04  420.96   <2e-16 ***
## `% Females raw value`           1.516e+01  3.665e-02  413.57   <2e-16 ***
## `% 65 and older raw value`     -1.870e+00  9.375e-03 -199.43   <2e-16 ***
```

```
## `Adult obesity raw value`       -1.053e+01  1.128e-02 -933.33   <2e-16 ***
## `Physical inactivity raw value`  1.828e+01  9.862e-03 1853.18   <2e-16 ***
## `Unemployment raw value`         7.787e+00  3.085e-02  252.38   <2e-16 ***
## `Income inequality raw value`    1.274e-02  4.721e-04   26.99   <2e-16 ***
## `Poor or fair health raw value` -4.566e+00  1.459e-02 -312.89   <2e-16 ***
## `Uninsured adults raw value`    -5.089e+00  7.402e-03 -687.50   <2e-16 ***
## `Adult smoking raw value`        3.363e+00  1.345e-02  250.03   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 48739096  on 445  degrees of freedom
## Residual deviance: 13412779  on 432  degrees of freedom
##   (6 observations deleted due to missingness)
## AIC: 13459324
##
## Number of Fisher Scoring iterations: 6
```

```r
summary(est_day_snc_t1_pois_model)
```

```
##
## Call:
## glm(formula = DayOfFirstCases ~ log_pop_dense + log_GDP + log_Pub_Trans +
##     log(Population) + `% Females raw value` + `% 65 and older raw value` +
##     `Adult obesity raw value` + `Physical inactivity raw value` +
##     `Unemployment raw value` + `Income inequality raw value` +
##     `Poor or fair health raw value` + `Uninsured adults raw value` +
##     `Adult smoking raw value`, family = "gaussian", data = day25_covars)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -44.831   -1.552    0.703    2.989   12.243
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       32.3189    12.2717   2.634  0.00875 **
## log_pop_dense                   -147.9050   213.4755  -0.693  0.48878
## log_GDP                           -1.0485     1.0050  -1.043  0.29739
## log_Pub_Trans                      0.5332     0.3133   1.702  0.08946 .
## log(Population)                   -1.2398     1.0829  -1.145  0.25290
## `% Females raw value`             84.3859    26.1392   3.228  0.00134 **
## `% 65 and older raw value`        -6.6039     7.5673  -0.873  0.38332
## `Adult obesity raw value`         -1.5171     9.4960  -0.160  0.87314
## `Physical inactivity raw value`    4.4885    10.2056   0.440  0.66030
## `Unemployment raw value`          57.8106    34.4287   1.679  0.09385 .
## `Income inequality raw value`     -0.9204     0.6006  -1.533  0.12611
## `Poor or fair health raw value`  -38.8263    19.1354  -2.029  0.04307 *
## `Uninsured adults raw value`       6.6476     8.2232   0.808  0.41931
## `Adult smoking raw value`         66.8955    16.4819   4.059 5.86e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 41.9019)
##
```

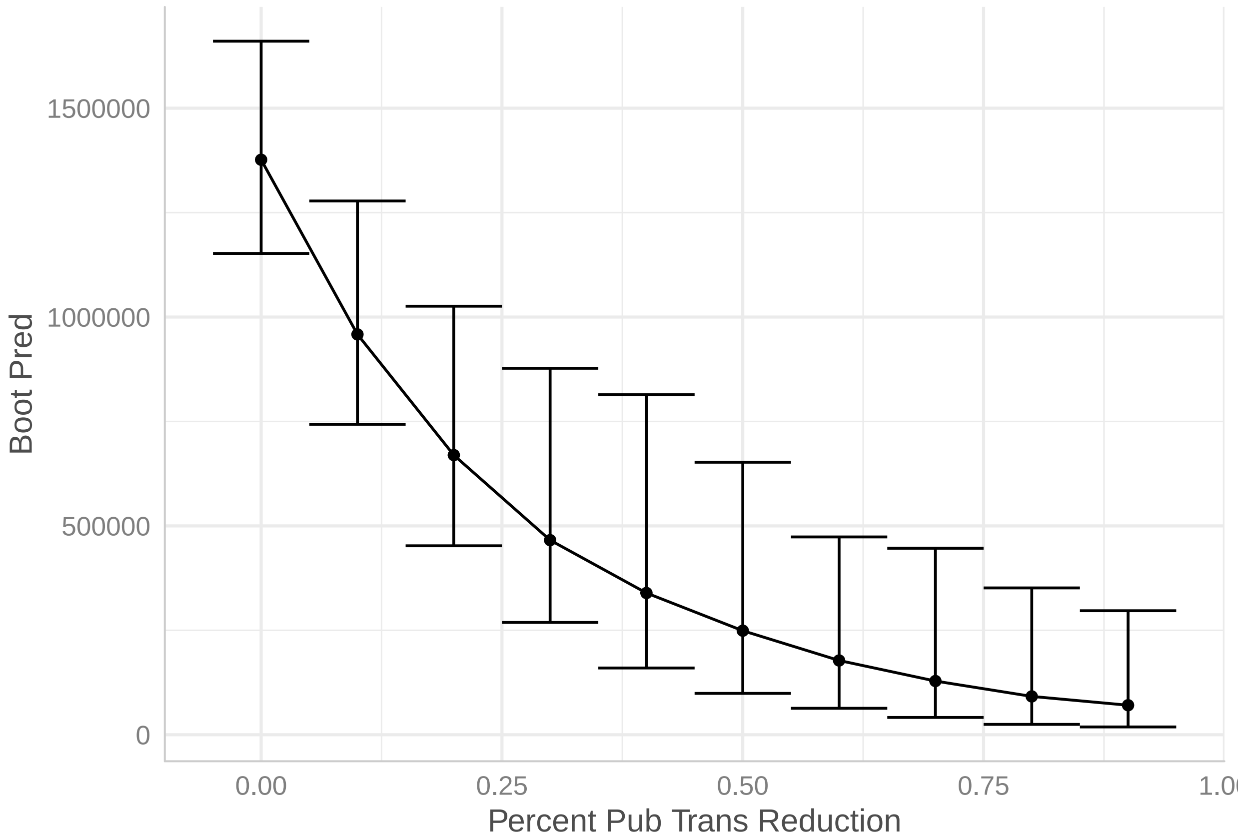
```
##     Null deviance: 22527  on 445  degrees of freedom
## Residual deviance: 18102  on 432  degrees of freedom
##   (6 observations deleted due to missingness)
## AIC: 2947.4
##
## Number of Fisher Scoring iterations: 2
```

```r
model_day25 <- est_day25_pois_model
model_snc_t1 <- est_day_snc_t1_pois_model

results_day25 <- as.data.frame(matrix(nrow = length(percents), ncol = 5))
colnames(results_day25) <- c('Percent Pub Trans Reduction',
                             'Boot Pred',
                             'Boot Low' ,
                             'Boot High',
                             'Init. Model Pred')

results_day_first <- as.data.frame(matrix(nrow = length(percents), ncol = 5))
colnames(results_day_first) <- c('Percent Pub Trans Reduction',
                                 'Boot Pred',
                                 'Boot Low' ,
                                 'Boot High',
                                 'Init. Model Pred')


for (i in 1:length(percents)) {

  perc <- percents[i]
  data_temp <- day25_covars

  data_temp$log_Pub_Trans <- day25_covars$log_Pub_Trans - (day25_covars$log_Pub_Trans*perc)

  pred_perc_red <- predict(model_day25, newdata = data_temp, type = 'response')
  pred_day_red <- predict(model_snc_t1, newdata = data_temp, type = 'response')

  sum_perc_red <- sum(pred_perc_red, na.rm = TRUE)
  mean_first_day <- mean(pred_day_red, na.rm = TRUE)

  boot_day25 <- replicate(1000, bootstrapCI(model = model_day25,
                                            perc =  perc,
                                            boot_pred_data = day25_covars,
                                            type = 'count'))

  boot_day1 <- replicate(1000, bootstrapCI(model = model_snc_t1,
                                           perc =  perc,
                                           boot_pred_data = day25_covars,
                                           type = 'day'))

  #mean_boot_diff <- mean(boot, na.rm = TRUE)

  CI_boot_day25 <- quantile(boot_day25, probs = c(0.025,0.50, 0.975), na.rm = TRUE)
  CI_boot_day1 <- quantile(boot_day1, probs = c(0.025,0.50, 0.975), na.rm = TRUE)
```

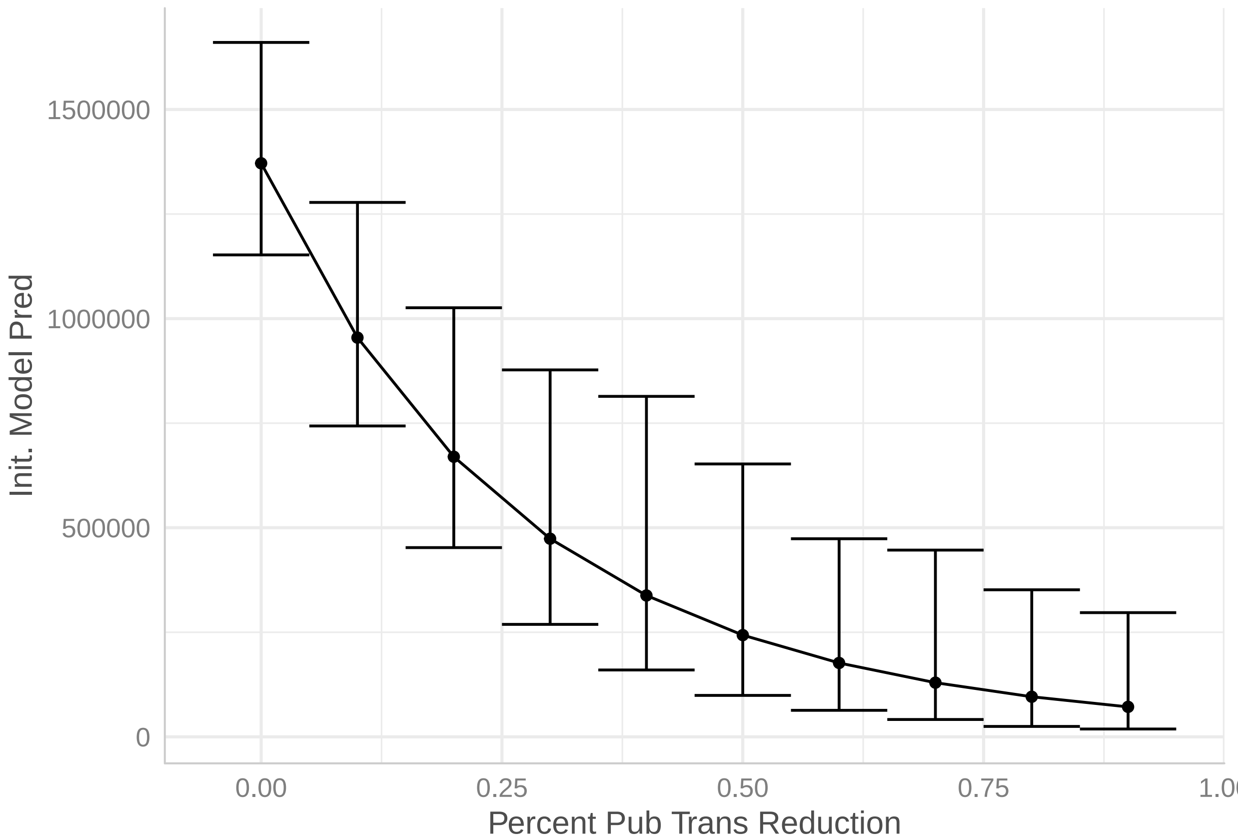
```
  results_day25[i,1] <- perc
  results_day25[i,2] <- CI_boot_day25[[2]]
  results_day25[i,3] <- CI_boot_day25[[1]]
  results_day25[i,4] <- CI_boot_day25[[3]]
  results_day25[i,5] <- sum_perc_red

  results_day_first[i,1] <- perc
  results_day_first[i,2] <- CI_boot_day1[[2]]
  results_day_first[i,3] <- CI_boot_day1[[1]]
  results_day_first[i,4] <- CI_boot_day1[[3]]
  results_day_first[i,5] <- mean_first_day

}
```

Plot cases over time marginally attributed to each reduction level - comparing bootstrap 50% percentile to initial model predictions as a sanity check of the bootstrap CI estimates:

```
ggplot(results_day25, aes(x=`Percent Pub Trans Reduction`, y=`Boot Pred`)) +
    geom_errorbar(aes(ymin=`Boot Low`, ymax=`Boot High`), width=.1) +
    geom_line() +
    geom_point()
```

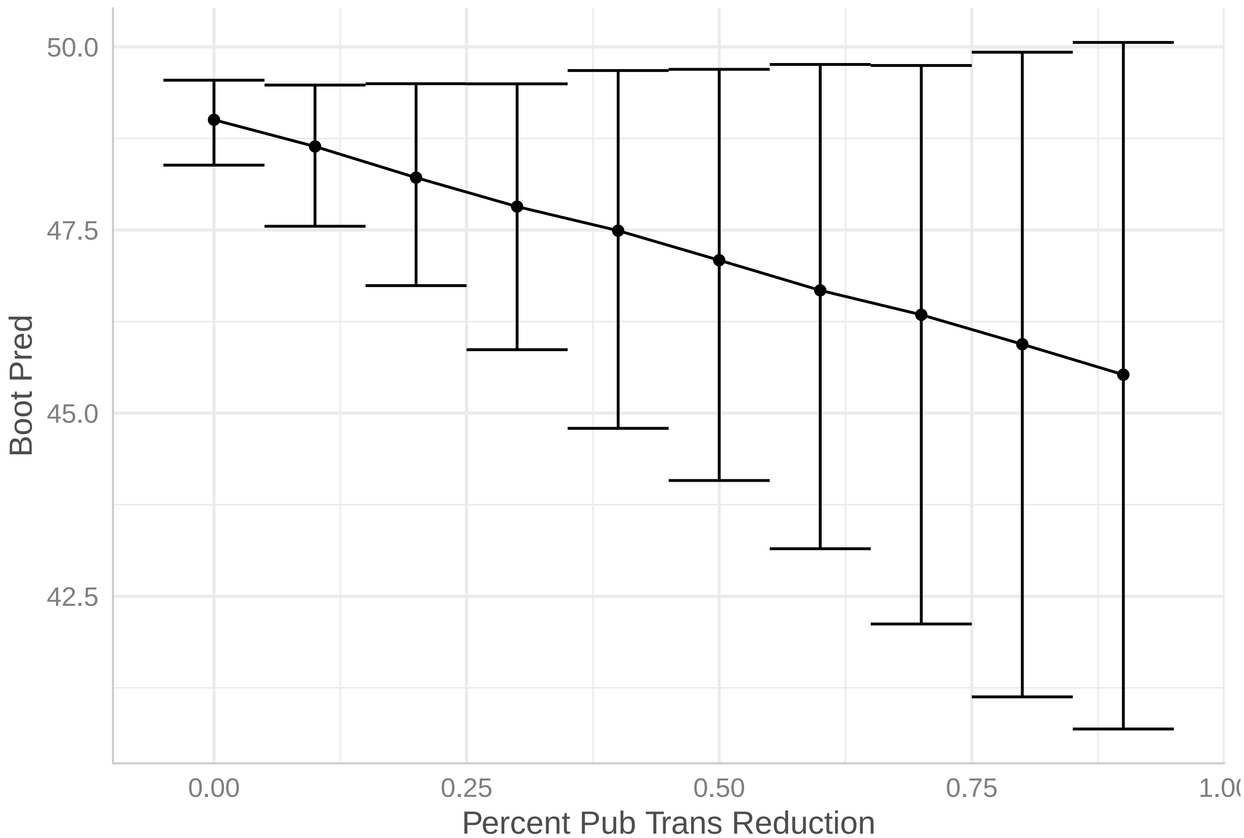



```
ggplot(results_day25, aes(x=`Percent Pub Trans Reduction`, y=`Init. Model Pred`)) +
    geom_errorbar(aes(ymin=`Boot Low`, ymax=`Boot High`), width=.1) +
    geom_line() +
    geom_point()
```

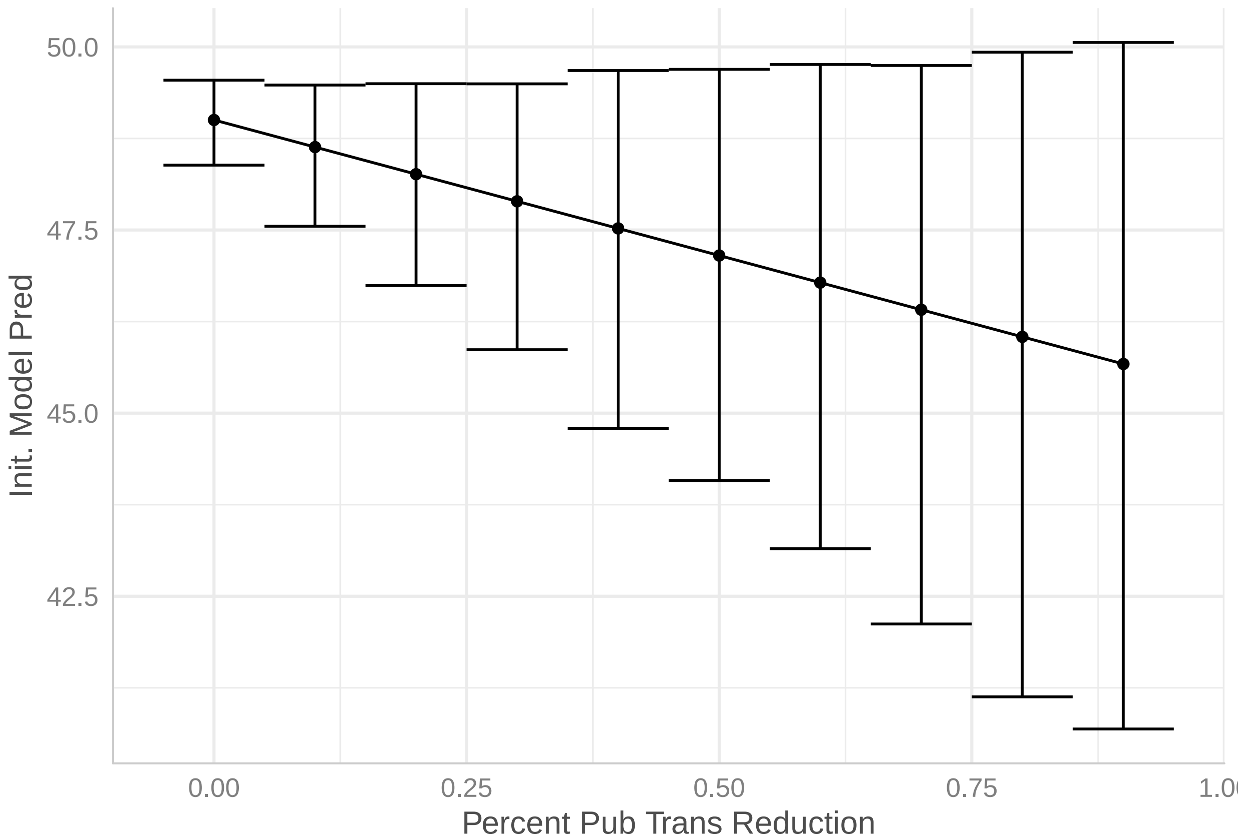


Plot average day since first case change due to public transit reduction, same method:

```
ggplot(results_day_first, aes(x=`Percent Pub Trans Reduction`, y=`Boot Pred`)) +
    geom_errorbar(aes(ymin=`Boot Low`, ymax=`Boot High`), width=.1) +
    geom_line() +
    geom_point()
```




```r
ggplot(results_day_first, aes(x=`Percent Pub Trans Reduction`, y=`Init. Model Pred`)) +
    geom_errorbar(aes(ymin=`Boot Low`, ymax=`Boot High`), width=.1) +
    geom_line() +
    geom_point()
```



Setting Up SuperLearner and TMLE

Can include more covariates and not assuming linearity of the model - sanity check to make sure parametric model is sound:

```
##set up node list

node_list <- list(
  W = c('popland', 'GDP',eco_vars_of_int[-c(1,2)]),
  A = "pub_trans_quantile",
  Y = "ConfirmedCasesDay25"
)

#process missing
processed <- process_missing(day25_covars, node_list)
COVID_data <- processed$data
node_list <- processed$node_list

ate_spec <- tmle_ATE(
  treatment_level = 10,
  control_level = 1
)

sl3_list_learners("continuous")
sl3_list_learners("binomial")
sl3_list_learners("categorical")

# choose y learners
```

```r
lrnr_mean <- make_learner(Lrnr_mean)
lrnr_xgboost <- make_learner(Lrnr_xgboost)
Lrnr_glm <- make_learner(Lrnr_glm)
Lrnr_hal9001 <- make_learner(Lrnr_hal9001)

# choose a learners
Lrnr_grf <- make_learner(Lrnr_grf)
#Lrnr_multivariate <- make_learner(Lrnr_multivariate)


# define metalearners appropriate to data types
ls_metalearner <- make_learner(Lrnr_nnls)
#mn_metalearner <- make_learner(Lrnr_solnp, metalearner_linear_multinomial,
                                #loss_loglik_multinomial)

sl_Y <- Lrnr_sl$new(learners = list(lrnr_mean, lrnr_xgboost, Lrnr_glm, Lrnr_hal9001),
                    metalearner = ls_metalearner)

sl_A <- Lrnr_sl$new(learners = list(Lrnr_grf))

learner_list <- list(A = sl_A, Y = sl_Y)
```

```r
tmle_fit <- tmle3(ate_spec, COVID_data, node_list, learner_list)
tmle_fit
estimates <- tmle_fit$summary$psi_transformed
estimates
```

Initial cross validated predictions: not used now

```r
States <- unique(day25$State)
CV.risk <- matrix(NA, nrow=length(States), ncol=6)

#estimates<- matrix(NA, nrow=500, ncol=4)
ObsData <- day25

for(i in States){

idx <- match(i,States)

validation_data <- ObsData %>% filter(State == i)
training_data <- ObsData %>% filter(State != i)

est1.model <- glm(ConfirmedCasesDay25 ~ popland + GDP + CommutingByPublicTransportation, family = "gaus

est2.model <- glm(ConfirmedCasesDay25 ~ popland + GDP + CommutingByPublicTransportation, family = "pois

est3.model <- glm(ConfirmedCasesDay25 ~ popland *GDP * CommutingByPublicTransportation, family = "gauss

est4.model <- glm(ConfirmedCasesDay25 ~ popland *GDP * CommutingByPublicTransportation, family = "poiss

est5.model <- glm(ConfirmedCasesDay25 ~ popland + GDP , family = "gaussian", data = training_data)

est6.model <- glm(ConfirmedCasesDay25 ~ popland + GDP , family = "poisson", data = training_data)
```

```r
predict.est1 <- predict(est1.model, newdata = validation_data)
predict.est2 <- predict(est2.model, newdata = validation_data)
predict.est3 <- predict(est3.model, newdata = validation_data)
predict.est4 <- predict(est4.model, newdata = validation_data)
predict.est5 <- predict(est5.model, newdata = validation_data)
predict.est6 <- predict(est6.model, newdata = validation_data)
#predict.est7 <- predict(est7.model, newdata = validation_data)

l2.hat1 <- mean((validation_data$ConfirmedCasesDay25 - predict.est1)^2)
l2.hat2 <- mean((validation_data$ConfirmedCasesDay25 - predict.est2)^2)
l2.hat3 <- mean((validation_data$ConfirmedCasesDay25 - predict.est3)^2)
l2.hat4 <- mean((validation_data$ConfirmedCasesDay25 - predict.est4)^2)
l2.hat5 <- mean((validation_data$ConfirmedCasesDay25 - predict.est5)^2)
l2.hat6 <- mean((validation_data$ConfirmedCasesDay25 - predict.est6)^2)
#l2.hat7 <- mean((validation_data$ConfirmedCasesDay25 - predict.est7)^2)

CV.risk[idx,] <- c(l2.hat1, l2.hat2, l2.hat3, l2.hat4,l2.hat5,l2.hat6)
}

colnames(CV.risk) <- c("l2.est1", "l2.est2", "l2.est3", "l2.est4", "l2.est5", "l2.est6")
CV.risk <- as.data.frame(CV.risk)

CV.risk
mses <- colMeans(CV.risk, na.rm = TRUE)

match(mses, min(mses))
```

```r
Y <- day25$ConfirmedCasesDay25
X <-  subset(day25, select= -ConfirmedCasesDay25)
X <- as.data.frame(X)

Q_lib <- c("SL.mean", "SL.glmnet", "SL.ranger", "SL.rpartPrune", "SL.bayesglm")
g_lib <- c("SL.mean", "SL.glmnet")


vim <- varimpact(Y = Y, data = X, Q.library = Q_lib, g.library = g_lib, family="gaussian" )
vim$results_all

plot_var("Population", vim)
```

Junk Drawer

```r
boot_log <- replicate(1000, bootstrapCI(model = est1.log_model, newdata = newdata))

day25$uci_log <- apply(boot_log, MARGIN = 1, FUN = quantile, probs = 0.925, na.rm=TRUE)
day25$lci_log <- apply(boot_log, MARGIN = 1, FUN = quantile, probs = 0.025, na.rm=TRUE)
day25$fit_log <- apply(boot_log, MARGIN = 1, FUN = quantile, probs = 0.5, na.rm=TRUE)

day25$uci <- apply(boot, MARGIN = 1, FUN = quantile, probs = 0.925, na.rm=TRUE)
day25$lci <- apply(boot, MARGIN = 1, FUN = quantile, probs = 0.025, na.rm=TRUE)
day25$fit <- apply(boot, MARGIN = 1, FUN = quantile, probs = 0.5, na.rm=TRUE)
```

```r
g3_log <- ggplot(day25, aes(x = day25$CommutingByPublicTransportation, y = ConfirmedCasesDay25)) +
  theme_bw() +
  geom_point() +
  geom_line(aes(y = fit)) +
  geom_ribbon(aes(ymin = lci, ymax = uci), alpha = 0.3)

g3_log

g3 <- ggplot(day25, aes(x = day25$CommutingByPublicTransportation, y = ConfirmedCasesDay25)) +
  theme_bw() +
  geom_point() +
  geom_line(aes(y = fit_log)) +
  geom_ribbon(aes(ymin = lci_log, ymax = uci_log), alpha = 0.3)

g3


#predictions for 10 and 90 quantiles log model
predict.log_est_10 <- predict(est1.log_model, newdata = marginal_data_10, type='response')
predict.log_est_90 <- predict(est1.log_model, newdata = marginal_data_90, type='response')

#predictions for 10 and 90 quantiles
predict.est_10 <- predict(est1.model, newdata = marginal_data_10, type='response')
predict.est_90 <- predict(est1.model, newdata = marginal_data_90, type='response')

log_marg_10_90_diff <- mean(predict.log_est_90 - predict.log_est_10, na.rm = TRUE)
marg_10_90_diff <- mean(predict.est_90 - predict.est_10, na.rm = TRUE)
```

The marginal differences for public transportation at 90 vs. 10 percentile

```r
marg_10_90_diff
log_marg_10_90_diff
```