Step 2 - Output

Anonymous

10/5/2020

```
load(file = "data/cis.Rdata")
```

Figure 1. Theoretical Model

Created in PowerPoint, load here.

```
knitr::include_graphics("results/Fig1.png")
```

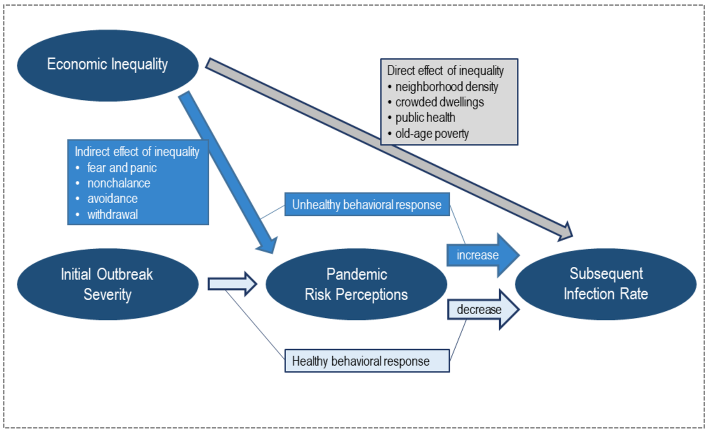


Figure 2

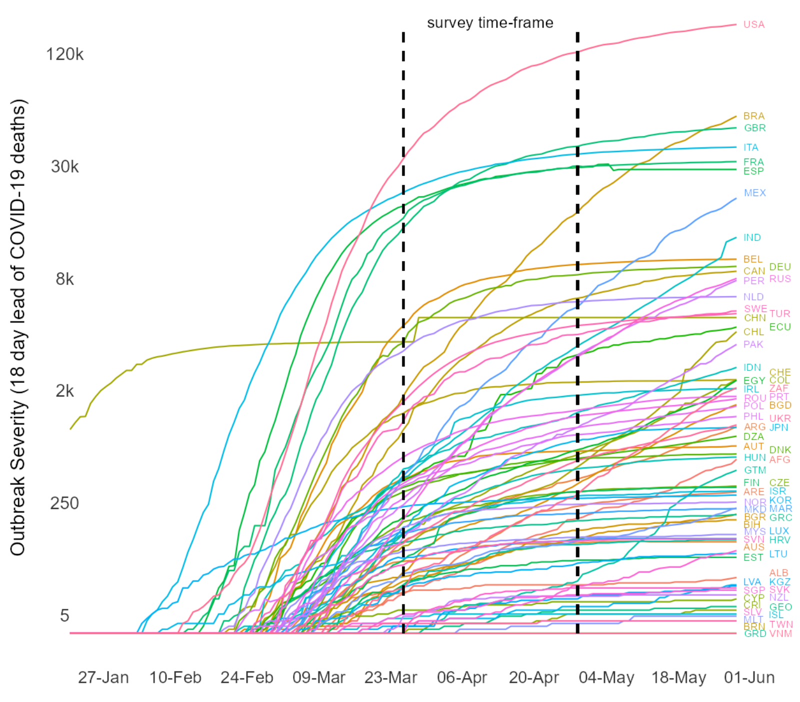
As testing rates vary dramatically by country, we measure the number of deaths with an 18-day lead as a better indicator of the severity of the outbreak by country. Those who died were inevitably sick or showing symptoms 18 days prior.

```
# this means the series ends at 06-01

deaths_long <- subset(deaths_long, date < as.Date("2020-06-02"))

# keep two extra days for plotting empty space

deaths_long$dead_lead <- ifelse(deaths_long$date > as.Date("2020-05-31"), NA, deaths_long$dead_lead)

# Using countries
deaths_longC <- subset(deaths_long, cow %in% use_countriesa)


# add Country name
deaths_longC$Country <- countrycode(deaths_longC$cow, "cown", "iso3c")
deaths_longC$Country <- ifelse(deaths_longC$cow ==  347, "KOS", deaths_longC$Country)




# log deaths
deaths_longC$dead_lead_log <- ifelse(deaths_longC$dead_lead > 3,log(deaths_longC$dead_lead),1)

# squared log deaths to accentuate differences
deaths_longC$dead_lead_log <- deaths_longC$dead_lead_log*deaths_longC$dead_lead_log

# create a label map so they do not overlap
deaths_longCL <- subset(deaths_longC, date == as.Date("2020-05-31"))
deaths_longCL <- deaths_longCL %>%
  mutate(date = ifelse(Country == "DEU" | Country == "RUS" | Country == "TUR" | Country == "ECU" | Country == "COL" | Countr
y == "ZAF" | Country == "PRT" | Country == "BGD" | Country == "CHE" | Country == "UKR" | Country == "JPN" | Country == "DNK"
| Country == "AFG" | Country == "CZE" | Country == "ISR" | Country == "KOR" | Country == "MAR" | Country == "GRC" | Country
 == "LUX" | Country == "HRV" | Country == "LTU" | Country == "ALB" | Country == "KGZ" | Country == "SVK" | Country == "NZL"
 | Country == "GEO" | Country == "ISL" | Country == "VNM" | Country == "TWN", "2020-06-06", "2020-06-01"),
         dead_lead_log = ifelse(Country == "CHE", 59.1, dead_lead_log),
         dead_lead_log = ifelse(Country == "DNK", 41.8, dead_lead_log),
         dead_lead_log = ifelse(Country == "UKR", 48.8, dead_lead_log),
         dead_lead_log = ifelse(Country == "CZE", 34.5, dead_lead_log),
         dead_lead_log = ifelse(Country == "KOR", 30.6, dead_lead_log),
         dead_lead_log = ifelse(Country == "MYS", 24.2, dead_lead_log),
         dead_lead_log = ifelse(Country == "AUS", 21.8, dead_lead_log),
         dead_lead_log = ifelse(Country == "LUX", 23.8, dead_lead_log),
         dead_lead_log = ifelse(Country == "FIN", 34.5, dead_lead_log),
         dead_lead_log = ifelse(Country == "KOS", 14, dead_lead_log),
         dead_lead_log = ifelse(Country == "ALB", 14.5, dead_lead_log),
         dead_lead_log = ifelse(Country == "LVA", 12.5, dead_lead_log),
         dead_lead_log = ifelse(Country == "GRC", 26.8, dead_lead_log),
         dead_lead_log = ifelse(Country == "KGZ", 12.5, dead_lead_log),
         dead_lead_log = ifelse(Country == "NZL", 9, dead_lead_log),
         dead_lead_log = ifelse(Country == "SVK", 10.5, dead_lead_log),
         dead_lead_log = ifelse(Country == "MEX", 99, dead_lead_log),
         dead_lead_log = ifelse(Country == "ESP", 103.8, dead_lead_log),
         dead_lead_log = ifelse(Country == "CRI", 7.35, dead_lead_log),
         dead_lead_log = ifelse(Country == "SLV", 5.78, dead_lead_log),
         dead_lead_log = ifelse(Country == "BRN", 2.6, dead_lead_log),
         dead_lead_log = ifelse(Country == "TWN", 3, dead_lead_log),
         dead_lead_log = ifelse(Country == "MLT", 4.16, dead_lead_log),
         dead_lead_log = ifelse(Country == "SWE", 73.2, dead_lead_log),
         dead_lead_log = ifelse(Country == "ZAF", 55.5, dead_lead_log),
          dead_lead_log = ifelse(Country == "ROU", 53.4, dead_lead_log),
         dead_lead_log = ifelse(Country == "ISL", 5.15, dead_lead_log),
         dead_lead_log = ifelse(Country == "CYP", 8.95, dead_lead_log),
         dead_lead_log = ifelse(Country == "SVK", 10.8, dead_lead_log))

# second go at labels
deaths_longCLa <- deaths_longCL


# labels need adjustment
deaths_longCLa <- deaths_longCLa %>%
  mutate(dead_lead_log = ifelse(Country == "AUS", 20.2, dead_lead_log),
         dead_lead_log = ifelse(Country == "BIH", 25.3, dead_lead_log),
         dead_lead_log = ifelse(Country == "BGR", 26.9, dead_lead_log),
         dead_lead_log = ifelse(Country == "MKD", 28.6, dead_lead_log),
         dead_lead_log = ifelse(Country == "MYS", 23.6, dead_lead_log))


fig2 <- ggplot(data=deaths_longC, aes(x=date , y=dead_lead_log, group=Country, color=Country)) +
    geom_line() +
  labs(x= "", y = "Outbreak Severity (18 day lead of COVID-19 deaths)") +
  geom_segment(aes(x = as.Date("2020-03-27"), y = 1, xend = as.Date("2020-03-27"), yend = 135), linetype = "dashed", color =
"black", size = 0.8) +
  geom_segment(aes(x = as.Date("2020-04-30"), y = 1, xend = as.Date("2020-04-30"), yend = 135), linetype = "dashed", color =
```

```
"black", size = 0.8) +
  annotate("text", x= as.Date("2020-01-21"), y= 5,
           label="5", size=4.5, color = "gray20") +
  annotate("text", x= as.Date("2020-01-21"), y= 30,
           label="250", size=4.5, color = "gray20") +
  annotate("text", x= as.Date("2020-01-21"), y= 55,
           label="2k", size=4.5, color = "gray20") +
  annotate("text", x= as.Date("2020-01-21"), y= 80,
           label="8k", size=4.5, color = "gray20") +
  annotate("text", x= as.Date("2020-01-21"), y= 105,
           label="30k", size=4.5, color = "gray20") +
  annotate("text", x= as.Date("2020-01-21"), y= 130,
           label="120k", size=4.5, color = "gray20") +
  annotate("text", x= as.Date("2020-04-13"), y=137, label="survey time-frame", size = 4, color="black") +
  scale_x_date(date_breaks = "2 weeks" , date_labels = "%d-%b") +
  geom_text(data = deaths_longCLa, aes(label = Country, colour = Country, x = as.Date(date), y = dead_lead_log, hjust = -.1
), size = 2.62) +
  theme(legend.position = "none",
        panel.background = element_blank(),
        axis.text.y = element_blank(),
        axis.ticks = element_blank(),
        axis.title.y = element_text(size=14, vjust=-0.5),
        axis.text.x = element_text(vjust=1, hjust=0.35, color = "gray20", size = 12),
        plot.margin = margin(0, 1, 0, 0, "cm"))

# log conversion
# 30 = 250
# 55 = 2,000
# 80 = 8,000
# 105 = 28,000
# 130 = 120,000
agg_png(file = "results/Fig2.png", width = 1200, height = 1020,  res = 144)
print(fig2)
dev.off()
```

```
## png
##   2
```

```
knitr::include_graphics("results/Fig2.png")
```




```
rm(deaths_longCL, deaths_longCLa)
```

Final Data Adjustments

Set up 18-Day lead

```
infect_merge <- as.data.frame(matrix(nrow = 74, ncol = 1))
infect_merge[1:74,1] <- as.numeric(use_countriesa)
colnames(infect_merge) <- c("cow")

d1 <- subset(deaths_longC, date == "2020-05-01", select = c(cow, dead_lead))
d2 <- subset(deaths_longC, date == "2020-05-31", select = c(cow, dead_lead, dead_1st_date))

infect_merge <- left_join(infect_merge, d1, by = "cow")
infect_merge <- left_join(infect_merge, d2, by = "cow")

colnames(infect_merge) <- c("cow","dead_lead_may1","dead_lead_may31","dead_1st_date")

rm(d1,d2)

df <- left_join(finaldf_C, infect_merge, by = "cow")

# fix Argentina and Indonesia (last top1 observation was 2004)
df$top1 <- ifelse(df$cow == 160, .168, ifelse(df$cow == 850, .085, df$top1))
```

Infection Increase (Ratio May 1-31)

```
# There was a change in reporting in Spain and the deaths dropped suddenly on May 6th, adjust for this here.

df$dead_lead_may1 <- ifelse(df$cow == 230, 27000, df$dead_lead_may1)
df <- df %>%
  mutate(rate_2 = ifelse(dead_lead_may31 == 0, 0, (dead_lead_may31 - dead_lead_may1) / (dead_lead_may1)),
         rate_2 = ifelse(rate_2 > 3, 3, rate_2) # trim outliers
  )

# deaths per capita May 1-31
df$newdthpc <- (df$dead_lead_may31 - df$dead_lead_may1)/df$pop
#make per 10,000 instead of per 1,000
df$newdthpc10 <- df$newdthpc*10
df$rate_3 <- df$newdthpc10
```

Squared Terms

```
df <- df %>%
  mutate(gini_dispR = gini_disp/100, # make gini smaller to keep boundaries reasonable in SEM
         gini_disp2 = gini_dispR^2,
         concern_self2 = concern_self^2,
         gini_disp2C = gini_disp2 - mean(gini_disp2),
         concern_self2C = concern_self2 - mean(concern_self2))
```

Main Models

Statistics

Set up first models (baseline models)

M1 Timing + severity of outbreak should predict risk perceptions. M22 Risk perceptions + risk perceptions-curve should predict deaths.

```
# adjust risk for severity of outbreak
m1 <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp, data = df)

# predicted values
df$m1p <- predict.lm(m1, df)

# residuals
df$m1r <- df$concern_self - df$m1p

m1a <- summary(m1)
m1a <- paste0("Adjusted r-square = ", round(m1a[["r.squared"]],3))

m23 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + concern_self + I(concern_self^2), data = df)
```

Descriptives

```
cor <- select(df, concern_self, concern_self_se, days_since_peak, conf_delta, gov_resp, rate_2, rate_3, gini_disp, top1, soc
policy, gdp)

corm <- cor %>%
  mutate(concern_selfsd = sd(concern_self, na.rm = T),
         concern_self_sesd = sd(concern_self_se, na.rm = T),
         days_since_peaksd = sd(days_since_peak, na.rm = T),
         conf_deltasd = sd(conf_delta, na.rm = T),
         gov_respsd = sd(gov_resp, na.rm = T),
         rate_2sd = sd(rate_2, na.rm = T),
         rate_3sd = sd(rate_3, na.rm = T),
         gini_dispsd = sd(gini_disp, na.rm = T),
         top1sd = sd(top1, na.rm = T),
         socpolicysd = sd(socpolicy, na.rm = T),
         gdpsd = sd(gdp, na.rm = T),concern_self_min = min(concern_self, na.rm = T),
         concern_self_se_min = min(concern_self_se, na.rm = T),
         days_since_peak_min = min(days_since_peak, na.rm = T),
         conf_delta_min = min(conf_delta, na.rm = T),
         gov_resp_min = min(gov_resp, na.rm = T),
         rate_2min = min(rate_2, na.rm = T),
         rate_3min = min(rate_3, na.rm = T),
         gini_disp_min = min(gini_disp, na.rm = T),
         top1_min = min(top1, na.rm = T),
         socpolicy_min = min(socpolicy, na.rm = T),
         gdp_min = min(gdp, na.rm = T),
         concern_self_max = max(concern_self, na.rm = T),
         concern_self_se_max = max(concern_self_se, na.rm = T),
         days_since_peak_max = max(days_since_peak, na.rm = T),
         conf_delta_max = max(conf_delta, na.rm = T),
         gov_resp_max = max(gov_resp, na.rm = T),
         rate_2max = max(rate_2, na.rm = T),
         rate_3max = max(rate_3, na.rm = T),
         gini_disp_max = max(gini_disp, na.rm = T),
         top1_max = max(top1, na.rm = T),
         socpolicy_max = max(socpolicy, na.rm = T),
         gdp_max = max(gdp, na.rm = T),
         n = ifelse(!is.na(top1), 74, 57),
         concern_self = mean(concern_self, na.rm = T),
         concern_self_se = mean(concern_self_se, na.rm = T),
         days_since_peak = mean(days_since_peak, na.rm = T),
         conf_delta = mean(conf_delta, na.rm = T),
         gov_resp = mean(gov_resp, na.rm = T),
         rate_2 = mean(rate_2, na.rm = T),
         rate_3 = mean(rate_3, na.rm = T),
         gini_disp = mean(gini_disp, na.rm = T),
         top1 = mean(top1, na.rm = T),
         socpolicy = mean(socpolicy, na.rm = T),
         gdp = mean(gdp, na.rm = T))

cor2 <- round(corm[1,1:11], 2)
cor2[2,] <- round(corm[1,12:21], 2)
cor2[3,] <- round(corm[1,22:31], 2)
cor2[4,] <- round(corm[1,32:41], 2)
cor2[5,1:11] <- 74
cor2[5,9] <- 57



colnames(cor2) <- c("Risk Perception", "SE of Risk Perception by Country", "Days Since Curve Inflection", "New Cases Past We
ek","Strength of Gov Intervention", "Increase in Infection", "Increase in Infection per capita", "Disposable Income Gini",
"Top 1% Income Concentration", "Welfare State", "GDP, per capita")

cor2 <- t(cor2)

colnames(cor2) <- c("Mean", "SD", "Min", "Max", "N")

kable_styling(kable(cor2, col.names = c("Mean", "SD", "Min", "Max", "N")))
```

	Mean	SD	Min	Max	N
Risk Perception	4.50	0.36	17.32	-0.94	74
SE of Risk Perception by Country	0.10	0.09	3.65	1.96	74
Days Since Curve Inflection	31.28	19.85	0.01	5.20	74
New Cases Past Week	0.37	0.60	0.00	0.34	74
Strength of Gov Intervention	0.00	1.00	-1.00	62.00	74
Increase in Infection	0.65	0.89	-2.20	1.00	74

	Mean	SD	Min	Max	N
Increase in Infection per capita	0.23	0.40	0.00	2.52	74
Disposable Income Gini	34.85	6.81	0.00	3.00	74
Top 1% Income Concentration	0.12	0.05	23.50	1.85	57
Welfare State	0.62	1.08	0.05	49.00	74
GDP, per capita	26.17	0.36	17.32	-0.94	74

```
tbl <- as.data.frame(c("3-item survey scale (COVIDiStress)", "Standard Error of individual level data","\"Outbreak Severity, actual\"; zero, or days since infection rate week-over-week started decreasing (Johns Hopkins, 18-day lead in COVID-19 deaths)", "\"Outbreak Severity, perceived\"; confirmned cases (Johns Hopkins)", "Severity of lockdown scale (Oxford/Blavatnik)", "Ratio of infection, May 31st to 1","Same as above, divided by population","Average of all available data (Solt)","Top 1% share (WID)", "Labor Market Coverage (ILO) & Social Spending (OECD) averaged", "In thousands, (Maddison)"))

colnames(tbl) <- c("Measurement")
tbl1 <- cbind(tbl, cor2)

write.csv(tbl1, file = "results/Tbl1.csv")
```

Correlations

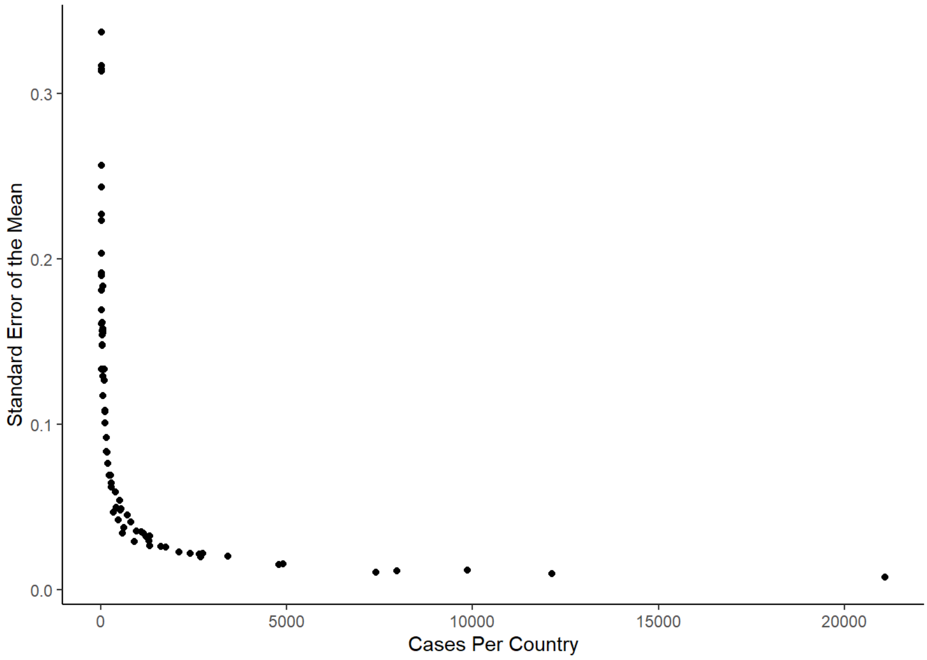
```
f1 <- cor(cor, use = "pairwise.complete.obs")


cor1 <- kable(f1, digits = 2, col.names = c("Risk Perception", "SE of Risk Perception by Country", "Days Since Curve Inflection", "New Cases Past Week","Strength of Gov Intervention", "Increase in Infection (ratio May 1-31)", "Increase in Infection (per capita May 1-31)", "Disposable Income Gini", "Top 1% Income Concentration", "Welfare State", "GDP, per capita (k)"))

kable_styling(cor1)
```

	Risk Perception	SE of Risk Perception by Country	Days Since Curve Inflection	New Cases Past Week	Strength of Gov Intervention	Increase in Infection (ratio May 1-31)	Increase in Infection (per capita May 1-31)	Disposable Income Gini	Top 1% Income Concentration	Welfare State
concern_self	1.00	0.02	-0.38	0.41	-0.19	0.48	0.27	0.55	0.51	-0.35
concern_self_se	0.02	1.00	-0.19	0.04	0.16	0.08	-0.11	0.13	0.00	-0.30
days_since_peak	-0.38	-0.19	1.00	-0.39	0.18	-0.77	-0.27	-0.51	-0.47	0.51
conf_delta	0.41	0.04	-0.39	1.00	-0.12	0.25	0.29	0.16	0.39	-0.20
gov_resp	-0.19	0.16	0.18	-0.12	1.00	-0.30	-0.19	-0.08	-0.31	-0.04
rate_2	0.48	0.08	-0.77	0.25	-0.30	1.00	0.36	0.55	0.64	-0.46
rate_3	0.27	-0.11	-0.27	0.29	-0.19	0.36	1.00	0.23	0.41	0.07
gini_disp	0.55	0.13	-0.51	0.16	-0.08	0.55	0.23	1.00	0.67	-0.69
top1	0.51	0.00	-0.47	0.39	-0.31	0.64	0.41	0.67	1.00	-0.38
socpolicy	-0.35	-0.30	0.51	-0.20	-0.04	-0.46	0.07	-0.69	-0.38	1.00
gdp	-0.32	-0.24	0.53	-0.26	-0.09	-0.53	-0.01	-0.53	-0.18	0.61

Additional Fig - CiS Cases per country

The COVIDiSTRESS (CiS) survey has a huge variance in cases per country

```
ggplot(df, aes(y = concern_self_se, x = cases)) +
  geom_point() +
  xlab("Cases Per Country") +
  ylab("Standard Error of the Mean") +
  theme_classic()
```




```
# ggplot(df, aes(x = reorder(iso, concern_self), y = concern_self)) +
#  geom_bar(stat = "identity") +
#  geom_errorbar(aes(ymin = ymin, ymax = ymax))
```

Additional Figs - Residuals

This visualizes the relationship between observed and predicted values of risk perceptions and the regression results used for this prediction.

This introduces the difference between ‘over’ and ‘under’ concern with the Coronavirus on average in a population.

```
# plot fitted v observed
ggplot(df, aes(y=m1p, x=concern_self)) +
  geom_point() +
  geom_text_repel(aes(label=iso), vjust = 1.5) +
  geom_abline(slope=1) +

  xlab("Observed Risk Perceptions") +
  ylab("Predicted Risk Perceptions") +
  theme_classic()
```

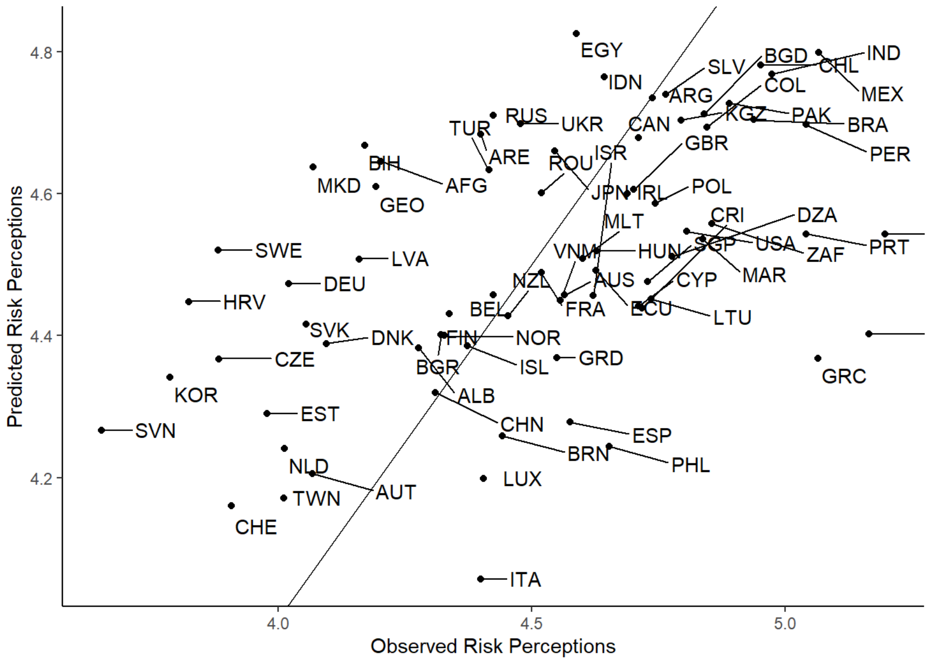
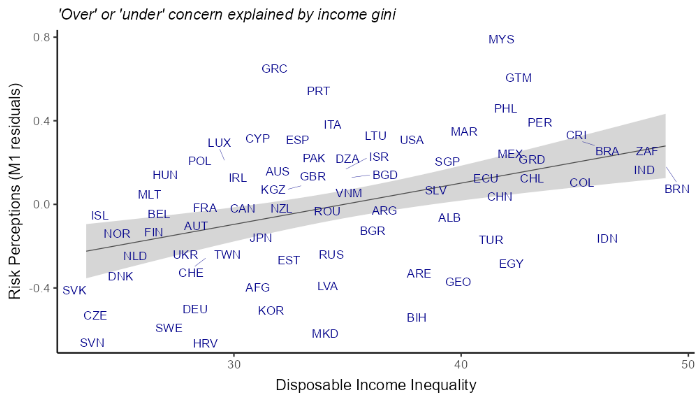


Figure 3

```
agg_png(file = "results/Fig3.png", width = 1000, height = 600, res = 144)
ggplot(df, aes(y=m1r, x=gini_disp)) +
  geom_smooth(method=lm, se=T, size = 0.3, color = "gray30") +
  geom_text_repel(aes(label=iso), size = 3, color = "blue4", segment.size = 0.1) +
  xlab("Disposable Income Inequality") +
  ylab("Risk Perceptions (M1 residuals)") +
  labs(title = "", subtitle = "\'Over\' or \'under\' concern explained by income gini") +
  theme_classic() +
  theme(
  plot.title = element_text(),
  plot.subtitle = element_text(face = "italic"),
  plot.caption = element_text(size = 9, color = "grey30", vjust = -2.5),
  axis.title.x = element_text(vjust = -0.8),
  axis.title.y = element_text(vjust = 2),
  )
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
invisible(dev.off())
knitr::include_graphics("results/Fig3.png")
```

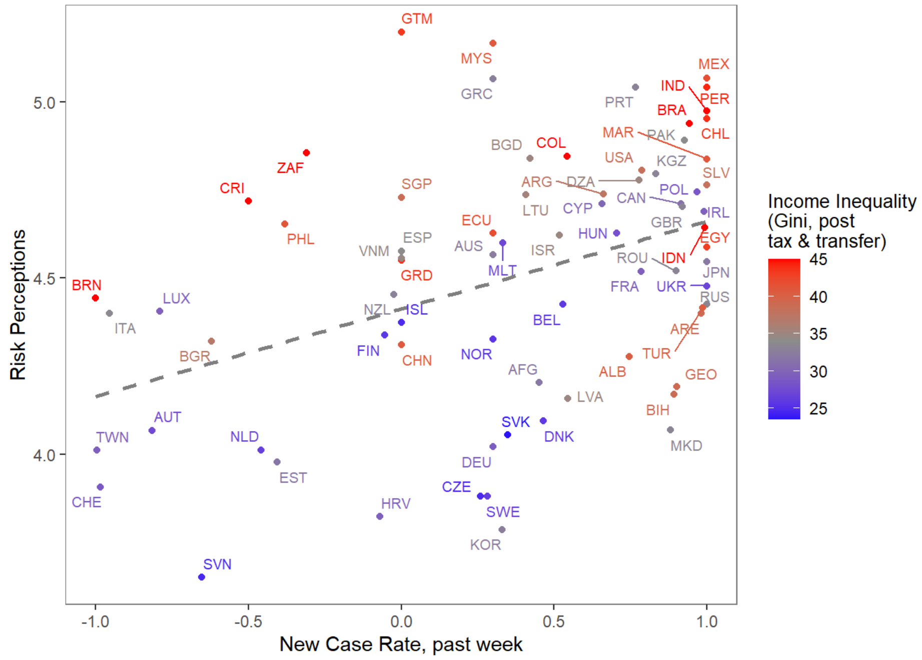



```
#
```

Additional Fig - 3 Way Plot

```
mid <- 34
# trim gini to get better color display
df$gini_color <- ifelse(df$gini_disp<45, df$gini_disp, 45)

ggplot(data=df, aes(x=conf_delta, y=concern_self, color = gini_color)) +
  geom_point() +
  geom_smooth(method=lm, se=FALSE, color = "gray50", linetype = "dashed") +
  geom_text_repel(aes(label = iso), size = 2.8) +
  scale_color_gradient2(midpoint=mid, low="blue", mid="gray55", high="red", space="Lab") +
  labs(x= "New Case Rate, past week", y = "Risk Perceptions", color = "Income Inequality\n(Gini, post\ntax & transfer)") +
      theme(panel.background = element_rect(fill = "white", colour = "grey50"),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        legend.title = element_text(size = 10))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Main Analyses

Predicting Risk Perceptions

Table 1. M1 through M5

So far this works to convert html to png https://cloudconvert.com/html-to-png (https://cloudconvert.com/html-to-png), but we should play around with htmltools package to automate this in the code

```
#m1x <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp*gov_resp_avg, data = df)

m2 <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp + gini_disp, data = df)

# predicted values for sem
df$m2p <- predict.lm(m2, df)

# residuals
df$m2r <- df$concern_self - df$m2p

m3 <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp + socpolicy, data = df)

m4 <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp + gdp, data = df)

m5 <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp + gini_disp + socpolicy + gdp, data = df)


tab_model(m1, m2, m3, m4, m5, p.style = "stars", p.threshold = c(0.10, 0.05, 0.01), show.ci = F, rm.terms = c("(Intercept)"
), show.loglik = T, show.aic = T, dv.labels = c("M1", "M2","M3", "M4","M5"), pred.labels = c("Days Since Curve Inflection",
"New Case Rate", "Government Intervention", "Disposable Income Inequality", "Welfare State Strength", "GDP Per Capita"), fil
e = "results/Tbl1.html")
```

	M1	M2	M3	M4	M5
Predictors	*Estimates*	*Estimates*	*Estimates*	*Estimates*	*Estimates*
Days Since Curve Inflection	-0.00 **	0.00	-0.00	-0.00	0.00
New Case Rate	0.18 ***	0.20 ***	0.18 ***	0.18 **	0.20 ***
Government Intervention	-0.04	-0.04	-0.05	-0.05	-0.04
Disposable Income Inequality		0.03 ***			0.03 ***
Welfare State Strength			-0.08 *		0.04
GDP Per Capita				-0.00	-0.00
Observations	74	74	74	74	74
R^2 / R^2 adjusted	0.238 / 0.206	0.427 / 0.393	0.277 / 0.235	0.262 / 0.219	0.433 / 0.382
AIC	47.802	28.783	45.915	47.492	31.982
log-Likelihood	-18.901	-8.392	-16.957	-17.746	-7.991

- *p<0.1 ** p<0.05 *** p<0.01*

Standardized Coefficients for Table 1

```
tab_model(m1, m2, m3, m4, m5, p.style = "stars", p.threshold = c(0.10, 0.05, 0.01), show.ci = F, rm.terms = c("(Intercept)"
), show.std = T, dv.labels = c("M1_Z", "M2_Z","M3_Z", "M4_Z","M5_Z"), pred.labels = c("Days Since Curve Inflection", "New Ca
se Rate", "Government Intervention", "Disposable Income Inequality", "Welfare State Strength", "GDP Per Capita"))
```

	M1_Z		M2_Z		M3_Z		M4_Z		M5_Z	
Predictors	*Estimates*	*std. Beta*	*Estimates*	*std. Beta*	*Estimates*	*std. Beta*	*Estimates*	*std. Beta*	*Estimates*	*std. Beta*
Days Since Curve Inflection	-0.00 **	-0.24	0.00	0.03	-0.00	-0.12	-0.00	-0.14	0.00	0.01
New Case Rate	0.18 ***	0.31	0.20 ***	0.33	0.18 ***	0.30	0.18 **	0.29	0.20 ***	0.34
Government Intervention	-0.04	-0.11	-0.04	-0.11	-0.05	-0.14	-0.05	-0.14	-0.04	-0.10
Disposable Income Inequality			0.03 ***	0.50					0.03 ***	0.57
Welfare State Strength					-0.08 *	-0.23			0.04	0.12
GDP Per Capita							-0.00	-0.19	-0.00	-0.03
Observations	74		74		74		74		74	
R^2 / R^2 adjusted	0.238 / 0.206		0.427 / 0.393		0.277 / 0.235		0.262 / 0.219		0.433 / 0.382	

- *p<0.1 ** p<0.05 *** p<0.01*

Additional Table "M2_resid"- The Regression behind Fig 3

```
m3r <- lm(m1r ~ days_since_peak + conf_delta + gov_resp + gini_disp, data = df)

tab_model(m3r, p.style = "stars", p.threshold = c(0.10, 0.05, 0.01), show.ci = F, rm.terms = c("(Intercept)"), show.std = T,
dv.labels = c("M2_resid"), pred.labels = c("Days Since Curve Inflection", "New Case Rate", "Intervention Severity", "Disposa
ble Income Inequality"))
```

	M2_resid	
Predictors	*Estimates*	*std. Beta*
Days Since Curve Inflection	0.00 **	0.30
New Case Rate	0.01	0.03
Intervention Severity	-0.00	-0.00
Disposable Income Inequality	0.03 ***	0.58
Observations	74	
R^2 / R^2 adjusted	0.247 / 0.204	

- *p<0.1 ** p<0.05 *** p<0.01*

Additional Table - Top 1% instead of Gini

```
# create dataset with top1 data cases only

dft <- df[(!is.na(df$top1)),]

m1t <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp, data = dft)

m2t <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp + top1, data = dft)

m3t <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp + socpolicy, data = dft)

m4t <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp + gdp, data = dft)

m5t <- lm(concern_self ~ days_since_peak + conf_delta + gov_resp + top1 + socpolicy + gdp, data = dft)

tab_model(m1t, m2t, m3t, m4t, m5t, p.style = "stars", p.threshold = c(0.10, 0.05, 0.01), show.ci = F, rm.terms = c("(Interce
pt)"), show.loglik = T, show.aic = T, dv.labels = c("M11", "M12","M13", "M14","M15"), pred.labels = c("Days Since Curve Infl
ection", "New Case Rate", "Government Intervention", "Top 1% Income Concentration", "Welfare State Strength", "GDP Per Capit
a"))
```

	M11	M12	M13	M14	M15
Predictors	*Estimates*	*Estimates*	*Estimates*	*Estimates*	*Estimates*
Days Since Curve Inflection	-0.00	0.00	-0.00	-0.00	0.00
New Case Rate	0.27 ***	0.22 ***	0.25 ***	0.25 ***	0.20 **
Government Intervention	-0.06	-0.03	-0.06	-0.07	-0.04
Top 1% Income Concentration		2.78 ***			2.75 **

Welfare State Strength			-0.05		-0.01
GDP Per Capita				-0.00	-0.00
Observations	57	57	57	57	57
R^2 / R^2 adjusted	0.270 / 0.229	0.363 / 0.314	0.286 / 0.231	0.278 / 0.222	0.374 / 0.298
AIC	38.338	32.563	39.074	39.732	35.628
log-Likelihood	-14.169	-10.282	-13.537	-13.866	-9.814

- *p<0.1 ** p<0.05 *** p<0.01*

Additional Table - Standardized Results for above

```
tab_model(m1t, m2t, m3t, m4t, m5t, p.style = "stars", p.threshold = c(0.10, 0.05, 0.01), show.ci = F, rm.terms = c("(Interce
pt)"), show.std = T, dv.labels = c("M11_Z", "M12_Z","M13_Z", "M14_Z","M15_Z"), pred.labels = c("Days Since Curve Inflection"
, "New Case Rate", "Intervention Severity", "Top 1% Income Concentration", "Welfare State Strength", "GDP Per Capita"))
```

	M11_Z		M12_Z		M13_Z		M14_Z		M15_Z	
Predictors	*Estimates*	*std. Beta*	*Estimates*	*std. Beta*	*Estimates*	*std. Beta*	*Estimates*	*std. Beta*	*Estimates*	*std. Beta*
Days Since Curve Inflection	-0.00	-0.07	0.00	0.05	-0.00	-0.02	-0.00	-0.04	0.00	0.10
New Case Rate	0.27 ***	0.44	0.22 ***	0.36	0.25 ***	0.42	0.25 ***	0.42	0.20 **	0.33
Intervention Severity	-0.06	-0.16	-0.03	-0.08	-0.06	-0.17	-0.07	-0.18	-0.04	-0.10
Top 1% Income Concentration			2.78 ***	0.37					2.75 **	0.36
Welfare State Strength					-0.05	-0.14			-0.01	-0.03
GDP Per Capita							-0.00	-0.10	-0.00	-0.10
Observations	57		57		57		57		57	
R^2 / R^2 adjusted	0.270 / 0.229		0.363 / 0.314		0.286 / 0.231		0.278 / 0.222		0.374 / 0.298	

- *p<0.1 ** p<0.05 *** p<0.01*

Predicting Infection Increase (ratio), May 1 - May 31

Conf_delta (new cases) measures a type of information the public and media consumes, it is actually a cause of a lower increase ratio of infection, probably because having awareness of a high infection increase leads to behavioral and policy changes.

Additional Table. Infection Increase as DV

These are the OLS analyses that mirror those in Table 2, but without ML simultaneous estimation. Here OLS is our 'first run', but it is biased because it cannot estimate mediation effects and therefore M40-M43 (and M50-M53 with top 1%) are the preferred models due to maximum-likelihood estimation.

```
m21 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp , data = df)

m22 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + concern_self, data = df)

m23 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + concern_self + I(concern_self^2), data = df)

m24 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp +  gini_disp, data = df)

m25 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + concern_self + gini_disp, data = df)

m26 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + gini_disp + I(gini_disp^2), data = df)

m27 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp +  concern_self + I(concern_self^2) + gini_disp + I(gini_disp^2),
data = df)



tab_model(m21, m22, m23, m24, m25, m26, m27, p.style = "stars", p.threshold = c(0.10, 0.05, 0.01), show.ci = F, rm.terms = c
("(Intercept)"), show.loglik = T, show.aic = T, dv.labels = c("M21", "M22","M23", "M24","M25","M26","M27"), pred.labels = c(
"Days Since Curve Inflection", "New Case Rate", "Government Intervention", "Risk Perceptions", "Risk Perceptions^2", "Dispos
able Income Inequality", "Disposable Income Inequality^2"))
```

	M21	M22	M23	M24	M25	M26	M27
Predictors	*Estimates*	*Estimates*	*Estimates*	*Estimates*	*Estimates*	*Estimates*	*Estimates*
Days Since Curve Inflection	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***
New Case Rate	-0.10	-0.21 *	-0.18	-0.09	-0.18	-0.03	-0.10
Government Intervention	-0.15 **	-0.13 **	-0.09	-0.16 **	-0.14 **	-0.15 **	-0.10 *
Risk Perceptions		0.60 ***	-10.87 ***		0.45 **		-10.07 ***

Risk Perceptions^2			1.29 ***				1.18 ***
Disposable Income Inequality				0.03 ***	0.02	-0.13	-0.12
Disposable Income Inequality^2						0.00	0.00
Observations	74	74	74	74	74	74	74
R^2 / R^2 adjusted	0.631 / 0.615	0.675 / 0.656	0.718 / 0.698	0.665 / 0.645	0.684 / 0.660	0.677 / 0.654	0.735 / 0.707
AIC	128.460	121.002	112.397	123.274	121.030	122.469	111.901
log-Likelihood	-59.230	-54.501	-49.198	-55.637	-53.515	-54.235	-46.951

- *p<0.1 ** p<0.05 *** p<0.01*

```
m27_beta <- lm.beta(m27)
m27_beta_concern <- m27_beta[["standardized.coefficients"]][["concern_self"]] + m27_beta[["standardized.coefficients"]][["I
(concern_self^2)"]]

m27_beta_gini <- m27_beta[["standardized.coefficients"]][["gini_disp"]] + m27_beta[["standardized.coefficients"]][["I(gini_d
isp^2)"]]
```

Standardized Coefficients M27

Standardized combined effect of term + term^2

Risk Perceptions = Disposable Income Inequality =

Additional Table. Infection Increase as DV Using Top 1% instead.

```
m31 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp , data = dft)

m32 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + concern_self, data = dft)

m33 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + concern_self + I(concern_self^2), data = dft)

m34 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp +  top1, data = dft)

m35 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + concern_self + top1, data = df)

m36 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp + top1 + I(top1^2), data = dft)

m37 <- lm(rate_2 ~ days_since_peak + conf_delta + gov_resp +  concern_self + I(concern_self^2) + top1 + I(top1^2), data = df
t)

tab_model(m31, m32, m33, m34, m35, m36, m37, p.style = "stars", p.threshold = c(0.10, 0.05, 0.01), show.ci = F, rm.terms = c
("(Intercept)"), show.loglik = T, show.aic = T, dv.labels = c("M31", "M32","M33", "M34","M35","M36", "M37"), pred.labels = c
("Days Since Curve Inflection", "New Case Rate", "Government Intervention", "Risk Perceptions", "Risk Perceptions^2", "Dispo
sable Income Inequality", "Disposable Income Inequality^2"))
```

	M31	**M32**	**M33**	**M34**	**M35**	**M36**	**M37**
Predictors	*Estimates*	*Estimates*	*Estimates*	*Estimates*	*Estimates*	*Estimates*	*Estimates*
Days Since Curve Inflection	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***
New Case Rate	-0.03	-0.17	-0.14	-0.14	-0.20 *	-0.13	-0.18
Government Intervention	-0.11	-0.07	-0.05	-0.04	-0.03	-0.05	-0.02
Risk Perceptions		0.51 **	-8.48 **		0.29		-6.52 *
Risk Perceptions^2			1.02 **				0.78 *
Disposable Income Inequality				5.67 ***	4.87 ***	0.96	-0.87
Disposable Income Inequality^2						15.40	16.88
Observations	57	57	57	57	57	57	57
R^2 / R^2 adjusted	0.622 / 0.601	0.666 / 0.640	0.699 / 0.670	0.710 / 0.688	0.722 / 0.695	0.713 / 0.685	0.747 / 0.711
AIC	85.389	80.449	76.373	72.373	71.958	73.723	70.580
log-Likelihood	-37.695	-34.224	-31.186	-30.186	-28.979	-29.861	-26.290

- *p<0.1 ** p<0.05 *** p<0.01*

Standardized Coefficients M37

```
m37_beta <- lm.beta(m37)
m37_beta_concern <- m37_beta[["standardized.coefficients"]][["concern_self"]] + m37_beta[["standardized.coefficients"]][["I
(concern_self^2)"]]

m37_beta_gini <- m37_beta[["standardized.coefficients"]][["top1"]] + m37_beta[["standardized.coefficients"]][["I(top1^2)"]]
```

Standardized combined effect of term + term^2

Risk Perceptions = Disposable Income Inequality =

Mediation Analysis

SEM squared variables must be constructed by hand. To keep estimates ‘under control’, it is useful to center the variables.

M40 Baseline Model

This is a maximum-likelihood estimated combination of M1 and M23, i.e., a structural equation model.

```
m40 <- '     rate_2 ~ days_since_peak + conf_delta + gov_resp + concern_self + b2*concern_self2  + c1*gini_dispR + c2*gini_d
isp2
             concern_self ~ days_since_peak + conf_delta + gov_resp + a1*gini_dispR + a2*gini_disp2
           # this is critical because we constructed one out of the other
             concern_self2 ~ concern_self
           # covariances

           # intercepts
             rate_2 ~ 1
             concern_self ~ 1
           # constraints
             c1 == 0
             c2 == 0
             b2 == 0
             a1 == 0
             a2 == 0
         '
m40fit <- sem(m40, data = df)
summary(m40fit, fit.measures = T)
```

```
## lavaan 0.6-7 ended normally after 76 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         19
##   Number of equality constraints                     5
##
##   Number of observations                            74
##
## Model Test User Model:
##
##   Test statistic                                41.957
##   Degrees of freedom                                10
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                               741.545
##   Degrees of freedom                                18
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.956
##   Tucker-Lewis Index (TLI)                       0.920
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)                -36.505
##   Loglikelihood unrestricted model (H1)        -15.526
##
##   Akaike (AIC)                                 101.010
##   Bayesian (BIC)                               133.267
##   Sample-size adjusted Bayesian (BIC)           89.147
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.208
##   90 Percent confidence interval - lower         0.145
##   90 Percent confidence interval - upper         0.275
##   P-value RMSEA <= 0.05                          0.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.116
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   rate_2 ~
##     dys_snc_p        -0.032    0.003   -9.526    0.000
##     conf_delt        -0.215    0.112   -1.912    0.056
##     gov_resp         -0.130    0.061   -2.149    0.032
##     cncrn_slf         0.597    0.188    3.176    0.001
##     cncrn_sl2 (b2)    0.000
##     gini_dspR (c1)    0.000       NA
##     gini_dsp2 (c2)    0.000
##   concern_self ~
##     dys_snc_p        -0.004    0.002   -2.156    0.031
##     conf_delt         0.185    0.066    2.799    0.005
##     gov_resp         -0.039    0.037   -1.043    0.297
##     gini_dspR (a1)    0.000       NA
##     gini_dsp2 (a2)    0.000       NA
##   concern_self2 ~
##     cncrn_slf         8.900    0.048  186.460    0.000
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .rate_2           -0.955    0.870   -1.097    0.273
##    .concern_self      4.572    0.084   54.532    0.000
##    .concern_self2   -19.670    0.216  -91.217    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .rate_2            0.255    0.042    6.083    0.000
##    .concern_self      0.098    0.016    6.083    0.000
##    .concern_self2     0.022    0.004    6.083    0.000
```

```
##
## Constraints:
##                                               |Slack|
##     c1 - 0                                     0.000
##     c2 - 0                                     0.000
##     b2 - 0                                     0.000
##     a1 - 0                                     0.000
##     a2 - 0                                     0.000
```

```
standardizedsolution(m40fit)
```

```
##               lhs op              rhs est.std    se        z pvalue ci.lower
## 1          rate_2  ~ days_since_peak  -0.713 0.057  -12.591  0.000   -0.824
## 2          rate_2  ~       conf_delta  -0.145 0.075   -1.929  0.054   -0.292
## 3          rate_2  ~         gov_resp  -0.146 0.067   -2.173  0.030   -0.278
## 4          rate_2  ~     concern_self   0.241 0.076    3.180  0.001    0.093
## 5          rate_2  ~    concern_self2   0.000 0.003    0.000  1.000   -0.006
## 6          rate_2  ~       gini_dispR   0.000 0.000    0.000  1.000    0.000
## 7          rate_2  ~       gini_disp2   0.000 0.000    0.000  1.000    0.000
## 8    concern_self  ~ days_since_peak  -0.240 0.107   -2.230  0.026   -0.450
## 9    concern_self  ~       conf_delta   0.308 0.104    2.969  0.003    0.105
## 10   concern_self  ~         gov_resp  -0.108 0.102   -1.051  0.293   -0.309
## 11   concern_self  ~       gini_dispR   0.000 0.000    0.000  1.000    0.000
## 12   concern_self  ~       gini_disp2   0.000 0.000    0.000  1.000    0.000
## 13  concern_self2  ~     concern_self   0.999 0.000 4104.510  0.000    0.998
## 14         rate_2 ~1                   -1.077 0.975   -1.104  0.270   -2.989
## 15   concern_self ~1                   12.772 1.022   12.498  0.000   10.769
## 16         rate_2 ~~           rate_2   0.325 0.052    6.232  0.000    0.223
## 17   concern_self ~~     concern_self   0.762 0.081    9.387  0.000    0.603
## 18  concern_self2 ~~    concern_self2   0.002 0.000    4.368  0.000    0.001
## 19 days_since_peak ~~ days_since_peak   1.000 0.000       NA     NA    1.000
## 20 days_since_peak ~~      conf_delta  -0.386 0.000       NA     NA   -0.386
## 21 days_since_peak ~~        gov_resp   0.178 0.000       NA     NA    0.178
## 22 days_since_peak ~~      gini_dispR  -0.508 0.000       NA     NA   -0.508
## 23 days_since_peak ~~      gini_disp2  -0.514 0.000       NA     NA   -0.514
## 24     conf_delta ~~       conf_delta   1.000 0.000       NA     NA    1.000
## 25     conf_delta ~~         gov_resp  -0.123 0.000       NA     NA   -0.123
## 26     conf_delta ~~       gini_dispR   0.155 0.000       NA     NA    0.155
## 27     conf_delta ~~       gini_disp2   0.133 0.000       NA     NA    0.133
## 28       gov_resp ~~         gov_resp   1.000 0.000       NA     NA    1.000
## 29       gov_resp ~~       gini_dispR  -0.080 0.000       NA     NA   -0.080
## 30       gov_resp ~~       gini_disp2  -0.081 0.000       NA     NA   -0.081
## 31     gini_dispR ~~       gini_dispR   1.000 0.000       NA     NA    1.000
## 32     gini_dispR ~~       gini_disp2   0.995 0.000       NA     NA    0.995
## 33     gini_disp2 ~~       gini_disp2   1.000 0.000       NA     NA    1.000
## 34  concern_self2 ~1                   -6.168 0.493  -12.522  0.000   -7.133
## 35 days_since_peak ~1                   1.586 0.000       NA     NA    1.586
## 36     conf_delta ~1                    0.615 0.000       NA     NA    0.615
## 37       gov_resp ~1                    0.000 0.000       NA     NA    0.000
## 38     gini_dispR ~1                    5.149 0.000       NA     NA    5.149
## 39     gini_disp2 ~1                    2.577 0.000       NA     NA    2.577
##    ci.upper
## 1    -0.602
## 2     0.002
## 3    -0.014
## 4     0.390
## 5     0.006
## 6     0.000
## 7     0.000
## 8    -0.029
## 9     0.512
## 10    0.093
## 11    0.000
## 12    0.000
## 13    0.999
## 14    0.835
## 15   14.775
## 16    0.427
## 17    0.921
## 18    0.003
## 19    1.000
## 20   -0.386
## 21    0.178
## 22   -0.508
## 23   -0.514
## 24    1.000
## 25   -0.123
## 26    0.155
## 27    0.133
## 28    1.000
## 29   -0.080
## 30   -0.081
## 31    1.000
## 32    0.995
## 33    1.000
## 34   -5.203
## 35    1.586
## 36    0.615
## 37    0.000
## 38    5.149
## 39    2.577
```

M41

This is a model that is M40 plus linear mediation of inequality by risk perceptions

```
m41 <- '   # direct effect
             rate_2 ~ c1*gini_dispR + c2*gini_disp2 + days_since_peak + conf_delta + gov_resp
           # mediator
             concern_self ~ a1*gini_dispR + a2*gini_disp2 + days_since_peak + conf_delta + gov_resp
             rate_2 ~ b1*concern_self + b2*concern_self2
           # this is critical because we constructed one out of the other
             concern_self2 ~ concern_self
           # covariances

           # intercepts
             rate_2 ~ 1
             concern_self ~ 1
           # constraints
             c2 == 0
             a2 == 0
             b2 == 0
           # indirect effect
             ab := a1*b1
           # total effect
             total := c1 + a1*b1
         '
m41fit <- sem(m41, data = df)

Tbl4 <- semTable(m41fit, type = "html", print.results = F)
```

```
semTable(m41fit, type = "html", print.results = F)
```

```
save_html(Tbl4, "Tbl4.html", background = "white", libdir = "results/")

file.move("Tbl4.html", "results/Tbl4.html")
```

```
## 0 files moved. 1 failed.
```

```
## Some files failed to move because it would have caused files to be overwritten.
##     * To allow overwriting, use `overwrite = TRUE`.
```

```
knit_print.html(Tbl4)
```

	Model			
	Estimate	Std. Err.	z	p
	Regression Slopes			
rate_2				
gini.dispR	1.62	1.15	1.41	.157
gini.disp2	0.00			
days.since.peak	-0.03	0.00	-8.06	.000
conf.delta	-0.18	0.11	-1.57	.117
gov.resp	-0.14	0.06	-2.28	.022
concern.self	0.45	0.21	2.09	.037
concern.self2	0.00			
concern_self				
gini.dispR	2.67	0.54	4.93	.000
gini.disp2	0.00			
days.since.peak	0.00	0.00	0.24	.811
conf.delta	0.20	0.06	3.47	.001
gov.resp	-0.04	0.03	-1.25	.213
concern_self2				
concern.self	8.90	0.05	186.46	.000
	Intercepts			
rate.2	-0.93	0.86	-1.08	.280
concern.self	3.49	0.23	15.03	.000
concern.self2	-19.67	0.22	-91.22	.000
gini.dispR	0.35+			
gini.disp2	0.13+			
days.since.peak	31.28+			
conf.delta	0.37+			
gov.resp	-0.00+			
	Residual Variances			
rate.2	0.25	0.04	6.08	.000
concern.self	0.07	0.01	6.08	.000
concern.self2	0.02	0.00	6.08	.000
gini.dispR	0.00+			
gini.disp2	0.00+			
days.since.peak	388.85+			

conf.delta	0.36+			
gov.resp	0.99+			
	Residual Covariances			
gini.dispR w/gini.disp2	0.00+			
gini.dispR w/days.since.peak	-0.68+			
gini.dispR w/conf.delta	0.01+			
gini.dispR w/gov.resp	-0.01+			
gini.disp2 w/days.since.peak	-0.50+			
gini.disp2 w/conf.delta	0.00+			
gini.disp2 w/gov.resp	-0.00+			
days.since.peak w/conf.delta	-4.54+			
days.since.peak w/gov.resp	3.49+			
conf.delta w/gov.resp	-0.07+			
	Constructed			
ab	1.19	0.62	1.92	.054
total	2.81	1.02	2.75	.006
	Fit Indices			
χ^2	18.97(8)			.015
CFI	0.98			
TLI	0.97			
RMSEA	0.14			

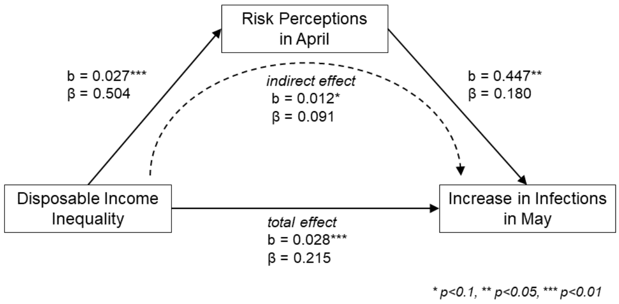
+Fixed parameter

```
standardizedsolution(m41fit)
```

```
##                lhs op            rhs est.std    se        z pvalue ci.lower
## 1           rate_2  ~      gini_dispR   0.124 0.087    1.420  0.155   -0.047
## 2           rate_2  ~      gini_disp2   0.000 0.000    0.000  1.000    0.000
## 3           rate_2  ~ days_since_peak  -0.662 0.068   -9.696  0.000   -0.796
## 4           rate_2  ~      conf_delta  -0.120 0.076   -1.576  0.115   -0.270
## 5           rate_2  ~        gov_resp  -0.154 0.066   -2.311  0.021   -0.284
## 6     concern_self  ~      gini_dispR   0.504 0.089    5.640  0.000    0.329
## 7     concern_self  ~      gini_disp2   0.000 0.000    0.000  1.000    0.000
## 8     concern_self  ~ days_since_peak   0.026 0.119    0.223  0.824   -0.206
## 9     concern_self  ~      conf_delta   0.332 0.090    3.692  0.000    0.156
## 10    concern_self  ~        gov_resp  -0.112 0.089   -1.256  0.209   -0.286
## 11          rate_2  ~    concern_self   0.180 0.086    2.091  0.036    0.011
## 12          rate_2  ~   concern_self2   0.000 0.004    0.000  1.000   -0.008
## 13   concern_self2  ~    concern_self   0.999 0.000 4198.761  0.000    0.998
## 14          rate_2 ~1                  -1.047 0.963   -1.087  0.277   -2.934
## 15    concern_self ~1                   9.739 1.182    8.237  0.000    7.421
## 16          rate_2 ~~          rate_2   0.316 0.050    6.304  0.000    0.218
## 17    concern_self ~~    concern_self   0.573 0.079    7.218  0.000    0.418
## 18   concern_self2 ~~   concern_self2   0.002 0.000    4.468  0.000    0.001
## 19      gini_dispR ~~      gini_dispR   1.000 0.000       NA     NA    1.000
## 20      gini_dispR ~~      gini_disp2   0.995 0.000       NA     NA    0.995
## 21      gini_dispR ~~ days_since_peak  -0.508 0.000       NA     NA   -0.508
## 22      gini_dispR ~~      conf_delta   0.155 0.000       NA     NA    0.155
## 23      gini_dispR ~~        gov_resp  -0.080 0.000       NA     NA   -0.080
## 24      gini_disp2 ~~      gini_disp2   1.000 0.000       NA     NA    1.000
## 25      gini_disp2 ~~ days_since_peak  -0.514 0.000       NA     NA   -0.514
## 26      gini_disp2 ~~      conf_delta   0.133 0.000       NA     NA    0.133
## 27      gini_disp2 ~~        gov_resp  -0.081 0.000       NA     NA   -0.081
## 28 days_since_peak ~~ days_since_peak   1.000 0.000       NA     NA    1.000
## 29 days_since_peak ~~      conf_delta  -0.386 0.000       NA     NA   -0.386
## 30 days_since_peak ~~        gov_resp   0.178 0.000       NA     NA    0.178
## 31      conf_delta ~~      conf_delta   1.000 0.000       NA     NA    1.000
## 32      conf_delta ~~        gov_resp  -0.123 0.000       NA     NA   -0.123
## 33        gov_resp ~~        gov_resp   1.000 0.000       NA     NA    1.000
## 34   concern_self2 ~1                  -6.168 0.470  -13.135  0.000   -7.088
## 35      gini_dispR ~1                   5.149 0.000       NA     NA    5.149
## 36      gini_disp2 ~1                   2.577 0.000       NA     NA    2.577
## 37 days_since_peak ~1                   1.586 0.000       NA     NA    1.586
## 38      conf_delta ~1                   0.615 0.000       NA     NA    0.615
## 39        gov_resp ~1                   0.000 0.000       NA     NA    0.000
## 43              ab :=           a1*b1   0.091 0.047    1.946  0.052   -0.001
## 44           total :=        c1+a1*b1   0.215 0.077    2.803  0.005    0.065
##    ci.upper
## 1     0.295
## 2     0.000
## 3    -0.528
## 4     0.029
## 5    -0.023
## 6     0.680
## 7     0.000
## 8     0.259
## 9     0.508
## 10    0.063
## 11    0.350
## 12    0.008
## 13    0.999
## 14    0.841
## 15   12.056
## 16    0.415
## 17    0.729
## 18    0.003
## 19    1.000
## 20    0.995
## 21   -0.508
## 22    0.155
## 23   -0.080
## 24    1.000
## 25   -0.514
## 26    0.133
## 27   -0.081
## 28    1.000
## 29   -0.386
## 30    0.178
## 31    1.000
## 32   -0.123
## 33    1.000
## 34   -5.248
## 35    5.149
## 36    2.577
## 37    1.586
## 38    0.615
## 39    0.000
```

```
## 43    0.183
## 44    0.365
```

```
knitr::include_graphics("results/Fig4.png")
```



M42

This adds the squared term for risk perceptions.

```
m42 <- '   # direct effect
             rate_2 ~ c1*gini_dispR + c2*gini_disp2 + days_since_peak + conf_delta + gov_resp
           # mediator
             concern_self ~ a1*gini_dispR + a2*gini_disp2 + days_since_peak + conf_delta + gov_resp
             rate_2 ~ b1*concern_self + b2*concern_self2
           # this is critical because we constructed one out of the other
             concern_self2 ~ concern_self
           # covariances

           # intercepts
             rate_2 ~ 1
             concern_self ~ 1
           # constraints
             c2 == 0
             a2 == 0

          '
m42fit <- sem(m42, data = df, meanstructure = T)
summary(m42fit, fit.measures = T)
```

```
## lavaan 0.6-7 ended normally after 107 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         19
##   Number of equality constraints                     2
##
##   Number of observations                            74
##
## Model Test User Model:
##
##   Test statistic                                 8.340
##   Degrees of freedom                                 7
##   P-value (Chi-square)                           0.304
##
## Model Test Baseline Model:
##
##   Test statistic                               741.545
##   Degrees of freedom                                18
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.998
##   Tucker-Lewis Index (TLI)                       0.995
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)                -19.696
##   Loglikelihood unrestricted model (H1)        -15.526
##
##   Akaike (AIC)                                  73.393
##   Bayesian (BIC)                               112.562
##   Sample-size adjusted Bayesian (BIC)           58.988
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.051
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.158
##   P-value RMSEA <= 0.05                          0.427
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.007
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   rate_2 ~
##     gini_dspR (c1)    1.519    1.068    1.422    0.155
##     gini_dsp2 (c2)    0.000
##     dys_snc_p        -0.030    0.003   -8.803    0.000
##     conf_delt        -0.144    0.106   -1.355    0.175
##     gov_resp         -0.100    0.056   -1.798    0.072
##   concern_self ~
##     gini_dspR (a1)    2.667    0.541    4.930    0.000
##     gini_dsp2 (a2)    0.000       NA
##     dys_snc_p         0.000    0.002    0.239    0.811
##     conf_delt         0.199    0.057    3.470    0.001
##     gov_resp         -0.040    0.032   -1.246    0.213
##   rate_2 ~
##     cncrn_slf (b1)  -10.874    3.273   -3.322    0.001
##     cncrn_sl2 (b2)    1.272    0.367    3.464    0.001
##   concern_self2 ~
##     cncrn_slf         8.900    0.048  186.460    0.000
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .rate_2           24.142    7.265    3.323    0.001
##    .concern_self      3.486    0.232   15.028    0.000
##    .concern_self2   -19.670    0.216  -91.217    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .rate_2            0.215    0.035    6.083    0.000
##    .concern_self      0.073    0.012    6.083    0.000
```

```
##    .concern_self2     0.022    0.004    6.083    0.000
##
## Constraints:
##                                        |Slack|
##     c2 - 0                               0.000
##     a2 - 0                               0.000
```

```
standardizedsolution(m42fit)
```

```
##              lhs op            rhs est.std    se        z pvalue ci.lower
## 1         rate_2  ~      gini_dispR   0.116 0.083    1.398  0.162   -0.047
## 2         rate_2  ~      gini_disp2   0.000 0.000    0.000  1.000   -0.001
## 3         rate_2  ~ days_since_peak  -0.674 0.112   -6.035  0.000   -0.892
## 4         rate_2  ~      conf_delta  -0.097 0.073   -1.334  0.182   -0.239
## 5         rate_2  ~        gov_resp  -0.113 0.064   -1.751  0.080   -0.239
## 6   concern_self  ~      gini_dispR   0.504 0.122    4.139  0.000    0.266
## 7   concern_self  ~      gini_disp2   0.000 0.001    0.000  1.000   -0.002
## 8   concern_self  ~ days_since_peak   0.026 0.455    0.058  0.954   -0.865
## 9   concern_self  ~      conf_delta   0.332 0.105    3.157  0.002    0.126
## 10  concern_self  ~        gov_resp  -0.112 0.091   -1.230  0.219   -0.290
## 11        rate_2  ~    concern_self  -4.396 1.516   -2.899  0.004   -7.368
## 12        rate_2  ~   concern_self2   4.580 1.530    2.993  0.003    1.581
## 13 concern_self2  ~    concern_self   0.999 0.001 1821.250  0.000    0.998
## 14        rate_2 ~1                  27.267 8.951    3.046  0.002    9.722
## 15  concern_self ~1                   9.739 1.988    4.897  0.000    5.841
## 16        rate_2 ~~          rate_2   0.275 0.091    3.016  0.003    0.096
## 17  concern_self ~~    concern_self   0.573 0.204    2.807  0.005    0.173
## 18 concern_self2 ~~   concern_self2   0.002 0.001    1.938  0.053    0.000
## 19    gini_dispR ~~      gini_dispR   1.000 0.000       NA     NA    1.000
## 20    gini_dispR ~~      gini_disp2   0.995 0.000       NA     NA    0.995
## 21    gini_dispR ~~ days_since_peak  -0.508 0.000       NA     NA   -0.508
## 22    gini_dispR ~~      conf_delta   0.155 0.000       NA     NA    0.155
## 23    gini_dispR ~~        gov_resp  -0.080 0.000       NA     NA   -0.080
## 24    gini_disp2 ~~      gini_disp2   1.000 0.000       NA     NA    1.000
## 25    gini_disp2 ~~ days_since_peak  -0.514 0.000       NA     NA   -0.514
## 26    gini_disp2 ~~      conf_delta   0.133 0.000       NA     NA    0.133
## 27    gini_disp2 ~~        gov_resp  -0.081 0.000       NA     NA   -0.081
## 28 days_since_peak ~~ days_since_peak   1.000 0.000       NA     NA    1.000
## 29 days_since_peak ~~      conf_delta  -0.386 0.000       NA     NA   -0.386
## 30 days_since_peak ~~        gov_resp   0.178 0.000       NA     NA    0.178
## 31    conf_delta ~~      conf_delta   1.000 0.000       NA     NA    1.000
## 32    conf_delta ~~        gov_resp  -0.123 0.000       NA     NA   -0.123
## 33      gov_resp ~~        gov_resp   1.000 0.000       NA     NA    1.000
## 34 concern_self2 ~1                  -6.168 1.114   -5.537  0.000   -8.351
## 35    gini_dispR ~1                   5.149 0.000       NA     NA    5.149
## 36    gini_disp2 ~1                   2.577 0.000       NA     NA    2.577
## 37 days_since_peak ~1                 1.586 0.000       NA     NA    1.586
## 38    conf_delta ~1                   0.615 0.000       NA     NA    0.615
## 39      gov_resp ~1                   0.000 0.000       NA     NA    0.000
##    ci.upper
## 1     0.279
## 2     0.001
## 3    -0.455
## 4     0.045
## 5     0.013
## 6     0.743
## 7     0.002
## 8     0.918
## 9     0.538
## 10    0.066
## 11   -1.424
## 12    7.580
## 13    1.000
## 14   44.811
## 15   13.636
## 16    0.453
## 17    0.974
## 18    0.004
## 19    1.000
## 20    0.995
## 21   -0.508
## 22    0.155
## 23   -0.080
## 24    1.000
## 25   -0.514
## 26    0.133
## 27   -0.081
## 28    1.000
## 29   -0.386
## 30    0.178
## 31    1.000
## 32   -0.123
## 33    1.000
## 34   -3.985
## 35    5.149
## 36    2.577
## 37    1.586
## 38    0.615
## 39    0.000
```

M43

M43 raw estimates

```
m43 <- '   # direct effect
             rate_2 ~ c1*gini_dispR + c2*gini_disp2 + w11*days_since_peak + w12*conf_delta + w13*gov_resp
           # mediator
             concern_self ~ a1*gini_dispR + a2*gini_disp2 + w1*days_since_peak + w2*conf_delta + w3*gov_resp
             rate_2 ~ b1*concern_self + b2*concern_self2
           # this is critical because we constructed one out of the other
             concern_self2 ~ concern_self
           # intercept naming
             concern_self ~ i1*1
             rate_2 ~ i2*1
           # constraint
             a2 == 0

         '
m43fit <- sem(m43, data = df, meanstructure = T)

summary(m43fit, fit.measures = T)
```

```
## lavaan 0.6-7 ended normally after 117 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of free parameters                         19
##   Number of equality constraints                     1
##
##   Number of observations                            74
##
## Model Test User Model:
##
##   Test statistic                                 5.838
##   Degrees of freedom                                 6
##   P-value (Chi-square)                           0.442
##
## Model Test Baseline Model:
##
##   Test statistic                               741.545
##   Degrees of freedom                                18
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    1.000
##   Tucker-Lewis Index (TLI)                       1.001
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)                -18.445
##   Loglikelihood unrestricted model (H1)        -15.526
##
##   Akaike (AIC)                                  72.890
##   Bayesian (BIC)                               114.364
##   Sample-size adjusted Bayesian (BIC)           57.639
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.000
##   90 Percent confidence interval - lower         0.000
##   90 Percent confidence interval - upper         0.149
##   P-value RMSEA <= 0.05                          0.556
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.007
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Regressions:
##                    Estimate  Std.Err  z-value  P(>|z|)
##   rate_2 ~
##     gin_dspR  (c1)  -12.445    8.643   -1.440    0.150
##     gin_dsp2  (c2)   19.457   12.014    1.620    0.105
##     dys_snc_ (w11)   -0.029    0.003   -8.435    0.000
##     conf_dlt (w12)   -0.104    0.108   -0.960    0.337
##     gov_resp (w13)   -0.101    0.055   -1.836    0.066
##   concern_self ~
##     gin_dspR  (a1)    2.667    0.541    4.930    0.000
##     gin_dsp2  (a2)    0.000       NA
##     dys_snc_  (w1)    0.000    0.002    0.239    0.811
##     conf_dlt  (w2)    0.199    0.057    3.470    0.001
##     gov_resp  (w3)   -0.040    0.032   -1.246    0.213
##   rate_2 ~
##     cncrn_sl  (b1)  -10.065    3.218   -3.127    0.002
##     cncrn_s2  (b2)    1.184    0.361    3.280    0.001
##   concern_self2 ~
##     cncrn_sl          8.900    0.048  186.460    0.000
##
## Intercepts:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .cncrn_slf (i1)    3.486    0.232   15.028    0.000
##    .rate_2    (i2)   24.651    7.288    3.382    0.001
##    .cncrn_sl2       -19.670    0.216  -91.217    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)
##    .rate_2            0.208    0.034    6.083    0.000
##    .concern_self      0.073    0.012    6.083    0.000
```

```
##    .concern_self2     0.022    0.004    6.083    0.000
##
## Constraints:
##                                               |Slack|
##     a2 - 0                                      0.000
```

```
standardizedsolution(m43fit)
```

```
##                lhs op            rhs est.std     se        z pvalue ci.lower
## 1           rate_2  ~      gini_dispR  -0.953  0.713   -1.337  0.181   -2.350
## 2           rate_2  ~      gini_disp2   1.076  0.729    1.477  0.140   -0.352
## 3           rate_2  ~ days_since_peak  -0.649  0.192   -3.371  0.001   -1.026
## 4           rate_2  ~      conf_delta  -0.070  0.076   -0.927  0.354   -0.219
## 5           rate_2  ~        gov_resp  -0.113  0.069   -1.637  0.102   -0.249
## 6     concern_self  ~      gini_dispR   0.504  0.180    2.799  0.005    0.151
## 7     concern_self  ~      gini_disp2   0.000  0.002    0.000  1.000   -0.004
## 8     concern_self  ~ days_since_peak   0.026  0.833    0.032  0.975   -1.606
## 9     concern_self  ~      conf_delta   0.332  0.137    2.427  0.015    0.064
## 10    concern_self  ~        gov_resp  -0.112  0.095   -1.171  0.242   -0.299
## 11          rate_2  ~    concern_self  -4.075  1.843   -2.211  0.027   -7.688
## 12          rate_2  ~   concern_self2   4.271  1.884    2.267  0.023    0.578
## 13   concern_self2  ~    concern_self   0.999  0.001 1002.766  0.000    0.997
## 14    concern_self ~1                   9.739  3.244    3.002  0.003    3.380
## 15          rate_2 ~1                  27.883 11.322    2.463  0.014    5.692
## 16          rate_2 ~~          rate_2   0.266  0.158    1.689  0.091   -0.043
## 17    concern_self ~~    concern_self   0.573  0.363    1.577  0.115   -0.139
## 18   concern_self2 ~~   concern_self2   0.002  0.002    1.067  0.286   -0.002
## 19      gini_dispR ~~      gini_dispR   1.000  0.000       NA     NA    1.000
## 20      gini_dispR ~~      gini_disp2   0.995  0.000       NA     NA    0.995
## 21      gini_dispR ~~ days_since_peak  -0.508  0.000       NA     NA   -0.508
## 22      gini_dispR ~~      conf_delta   0.155  0.000       NA     NA    0.155
## 23      gini_dispR ~~        gov_resp  -0.080  0.000       NA     NA   -0.080
## 24      gini_disp2 ~~      gini_disp2   1.000  0.000       NA     NA    1.000
## 25      gini_disp2 ~~ days_since_peak  -0.514  0.000       NA     NA   -0.514
## 26      gini_disp2 ~~      conf_delta   0.133  0.000       NA     NA    0.133
## 27      gini_disp2 ~~        gov_resp  -0.081  0.000       NA     NA   -0.081
## 28 days_since_peak ~~ days_since_peak   1.000  0.000       NA     NA    1.000
## 29 days_since_peak ~~      conf_delta  -0.386  0.000       NA     NA   -0.386
## 30 days_since_peak ~~        gov_resp   0.178  0.000       NA     NA    0.178
## 31      conf_delta ~~      conf_delta   1.000  0.000       NA     NA    1.000
## 32      conf_delta ~~        gov_resp  -0.123  0.000       NA     NA   -0.123
## 33        gov_resp ~~        gov_resp   1.000  0.000       NA     NA    1.000
## 34   concern_self2 ~1                  -6.168  1.966   -3.138  0.002  -10.021
## 35      gini_dispR ~1                   5.149  0.000       NA     NA    5.149
## 36      gini_disp2 ~1                   2.577  0.000       NA     NA    2.577
## 37 days_since_peak ~1                   1.586  0.000       NA     NA    1.586
## 38      conf_delta ~1                   0.615  0.000       NA     NA    0.615
## 39        gov_resp ~1                   0.000  0.000       NA     NA    0.000
##    ci.upper
## 1     0.444
## 2     2.505
## 3    -0.272
## 4     0.078
## 5     0.022
## 6     0.858
## 7     0.004
## 8     1.659
## 9     0.600
## 10    0.075
## 11   -0.463
## 12    7.964
## 13    1.001
## 14   16.098
## 15   50.074
## 16    0.576
## 17    1.286
## 18    0.006
## 19    1.000
## 20    0.995
## 21   -0.508
## 22    0.155
## 23   -0.080
## 24    1.000
## 25   -0.514
## 26    0.133
## 27   -0.081
## 28    1.000
## 29   -0.386
## 30    0.178
## 31    1.000
## 32   -0.123
## 33    1.000
## 34   -2.315
## 35    5.149
## 36    2.577
## 37    1.586
## 38    0.615
## 39    0.000
```

M43 for prediction/calculation

This strategy follows Hayes and Preacher (2010). The effect is non-linear meaning that it is heterogeneous. There is no single value of the income inequality effect. Hayes and Preacher (2010) suggest estimating an **instantaneous indirect effect** which is actually based on derivatives and then can be plotted as different effects at different levels.

Table 1 in Hayes and Preacher (2010) offers the derivation of the formulas:

Mediation Formulas

$\hat{Y} = i_2 + c_1' X + c_2' X^2 + b_1 M + b_2 M^2$

$\hat{M} = i_1 + aX$

Instantaneous Indirect Effects of X on Y through M are derived as (Table 1):

$a(b_1 + 2b_2\hat{M}))$

The only problem is that the effect of X includes a squared term, this squared term must be treated as a covariate (like W in their paper) and then the model should work, changing the squared term's fixed value for each.

Also, to get predicted instantaneous indirect effect we need to remove the covariance of concern_self and gini_disp2 from contaminating the estimates. We fixed gini_disp2 to zero in the equation for concern_self, thus the algorithm find every other possible way to explain this residual correlation before letting it fall into the residual. This likely skews the results, therefore, for prediction purposes we allow all of this correlation to simply fall into the residual thus aligning the instantaneous indirect effect with the predicted values of rate_2

```
m43p <- '   # direct effect
             rate_2 ~ c1*gini_dispR + c2*gini_disp2 + w11*days_since_peak + w12*conf_delta + w13*gov_resp
           # mediator
             concern_self ~ a1*gini_dispR + a2*gini_disp2 + w1*days_since_peak + w2*conf_delta + w3*gov_resp
             rate_2 ~ b1*concern_self + b2*concern_self2
           # this is critical because we constructed one out of the other
             concern_self2 ~ concern_self
           # to fix the residual variance out of the predictions of the model this is necessary
             concern_self ~~ gini_disp2
           # intercept naming
             concern_self ~ i1*1
             rate_2 ~ i2*1
           # constraint
             a2 == 0
           # instantaneous indir effect calc at values
             x1 := .225
             x2 := .25
             x3 := .275
             x4 := .30
             x5 := .325
             x6 := .35
             x7 := .375
             x8 := .40
             x9 := .425
             x10 := .45
             x11 := .475
             x12 := .50
           # requires fixing covariates and gini_disp2 = 0
             predm1 := i1+(a1*x1)+w1*.30+w2*0.37+0*w3
             predm2 := i1+(a1*x2)+w1*.30+w2*0.37+0*w3
             predm3 := i1+(a1*x3)+w1*.30+w2*0.37+0*w3
             predm4 := i1+(a1*x4)+w1*.30+w2*0.37+0*w3
             predm5 := i1+(a1*x5)+w1*.30+w2*0.37+0*w3
             predm6 := i1+(a1*x6)+w1*.30+w2*0.37+0*w3
             predm7 := i1+(a1*x7)+w1*.30+w2*0.37+0*w3
             predm8 := i1+(a1*x8)+w1*.30+w2*0.37+0*w3
             predm9 := i1+(a1*x9)+w1*.30+w2*0.37+0*w3
             predm10 := i1+(a1*x10)+w1*.30+w2*0.37+0*w3
             predm11 := i1+(a1*x11)+w1*.30+w2*0.37+0*w3
             predm12 := i1+(a1*x12)+w1*.30+w2*0.37+0*w3
           # instantaneous indirect effects (gini_disp2 must be held constant here)
             theta1 := (b1+2*b2*predm1)*a1
             theta2 := (b1+2*b2*predm2)*a1
             theta3 := (b1+2*b2*predm3)*a1
             theta4 := (b1+2*b2*predm4)*a1
             theta5 := (b1+2*b2*predm5)*a1
             theta6 := (b1+2*b2*predm6)*a1
             theta7 := (b1+2*b2*predm7)*a1
             theta8 := (b1+2*b2*predm8)*a1
             theta9 := (b1+2*b2*predm9)*a1
             theta10 := (b1+2*b2*predm10)*a1
             theta11 := (b1+2*b2*predm11)*a1
             theta12 := (b1+2*b2*predm12)*a1
           # pred values
             predy1 := i2 + c1*x1 + c2*x1*x1 + b1*predm1 + b2*predm1*predm1 + w11*.30+w12*0.37+0*w13
             predy2 := i2 + c1*x2 + c2*x2*x2 + b1*predm2 + b2*predm2*predm2 + w11*.30+w12*0.37+0*w13
             predy3 := i2 + c1*x3 + c2*x3*x3 + b1*predm3 + b2*predm3*predm3 + w11*.30+w12*0.37+0*w13
             predy4 := i2 + c1*x4 + c2*x4*x4 + b1*predm4 + b2*predm4*predm4 + w11*.30+w12*0.37+0*w13
             predy5 := i2 + c1*x5 + c2*x5*x5 + b1*predm5 + b2*predm5*predm5 + w11*.30+w12*0.37+0*w13
             predy6 := i2 + c1*x6 + c2*x6*x6 + b1*predm6 + b2*predm6*predm6 + w11*.30+w12*0.37+0*w13
             predy7 := i2 + c1*x7 + c2*x7*x7 + b1*predm7 + b2*predm7*predm7 + w11*.30+w12*0.37+0*w13
             predy8 := i2 + c1*x8 + c2*x8*x8 + b1*predm8 + b2*predm8*predm8 + w11*.30+w12*0.37+0*w13
             predy9 := i2 + c1*x9 + c2*x9*x9 + b1*predm9 + b2*predm9*predm9 + w11*.30+w12*0.37+0*w13
             predy10 := i2 + c1*x10 + c2*x10*x10 + b1*predm10 + b2*predm10*predm10 + w11*.30+w12*0.37+0*w13
             predy11 := i2 + c1*x11 + c2*x11*x11 + b1*predm11 + b2*predm11*predm11 + w11*.30+w12*0.37+0*w13
             predy12 := i2 + c1*x12 + c2*x12*x12 + b1*predm12 + b2*predm12*predm12 + w11*.30+w12*0.37+0*w13

          '
m43pfit <- sem(m43p, data = df, meanstructure = T)
```

Table 2. Main Models

We used an html template from tab_model command and hand edited the values to produce a table that was visually identical to Table 1. Automation would be ideal here, but we double checked the scores.

```
# set up frames

m40tab <- matrix(summary(m40fit, fit.measures = T))
m40fits <- as.data.frame(m40tab[[1]])
m40parm <- as.data.frame(m40tab[[2]])
m40parm <- select(m40parm, -c(label, exo))
m41tab <- matrix(summary(m41fit, fit.measures = T))
m41fits <- as.data.frame(m41tab[[1]])
m41parm <- as.data.frame(m41tab[[2]])
m41parm <- select(m41parm, -c(label, exo))
m42tab <- matrix(summary(m42fit, fit.measures = T))
m42fits <- as.data.frame(m42tab[[1]])
m42parm <- as.data.frame(m42tab[[2]])
m42parm <- select(m42parm, -c(label,exo))
m43tab <- matrix(summary(m43fit, fit.measures = T))
m43fits <- as.data.frame(m43tab[[1]])
m43parm <- as.data.frame(m43tab[[2]])
m43parm <- select(m43parm, -c(label, exo))

# combine and name which results go with which models

semtab <- rbind(m40parm,m41parm,m42parm,m43parm)
semtab_labels <- as.data.frame(matrix(nrow = length(semtab[,1]), ncol = 1))
semtab_labels$V1 <- as.list(unlist(strsplit(paste(paste(replicate(length(m40parm[,1]), "m40"), collapse = ","), paste(replic
ate(length(m41parm[,1]), "m41"), collapse = ","), paste(replicate(length(m42parm[,1]), "m42"), collapse = ","), paste(replic
ate(length(m43parm[,1]), "m43"), collapse = ","), sep = ","), ",")))

semtab <- cbind(semtab_labels, semtab)
semtab <- semtab %>%
  mutate(est = round(est,3),
         se = round(se,3),
         pvalue = round(pvalue,3),
         stars = ifelse(pvalue>0.1,"", ifelse(pvalue>0.05,"*", ifelse(pvalue>0.01,"**","***"))),
         est = ifelse(rhs == "gini_dispR" | rhs == "gini_disp2", est/100, est),
         se = ifelse(rhs == "gini_dispR" | rhs == "gini_disp2", se/100, se)
  )

semtab <- subset(semtab, op == "~" & lhs!= "concern_self2")


semfits <- round(cbind(m40fits,m41fits,m42fits,m43fits),3)
colnames(semfits) <- c("m40","m41","m42","m43")

# get r-squared
m40r2 <- round(as.data.frame(inspect(m40fit,'r2')),3)
m41r2 <- round(as.data.frame(inspect(m41fit,'r2')),3)
m42r2 <- round(as.data.frame(inspect(m42fit,'r2')),3)
m43r2 <- round(as.data.frame(inspect(m43fit,'r2')),3)

semr2s <- cbind(m40r2,m41r2,m42r2,m43r2)
semr2s <- semr2s[1:2,]
colnames(semr2s) <- c("m40","m41","m42","m43")

semfits <- rbind(semfits,semr2s)

semfits <- semfits[c("rate_2", "concern_self", "aic", "bic", "logl"),]

rm(m40tab, m40fits, m40parm, m41tab, m41fits, m41parm, m42tab, m42fits, m42parm, m43tab, m43fits, m43parm, m40r2,m41r2,m42r
2,m43r2, semr2s, semtab_labels)

kable(semtab)
kable(semfits)
```

To create a table identical to the tab_model table we adjust the html code by hand, but hopefully we can automate this in the future.

```
knitr::include_graphics("results/Tbl2.png")
```

Infection Increase in May (paths predicting Y)	**M40** *Estimates*	**M41** *Estimates*	**M42** *Estimates*	**M43** *Estimates*
Days Since Curve Inflection	-0.03 ***	-0.03 ***	-0.03 ***	-0.03 ***
New Case Rate	-0.22 *	-0.18 +	-0.14	-0.10
Government Intervention	-0.13 **	-0.14 **	-0.10 *	-0.10 *
Risk Perceptions	0.60 ***	0.45 **	-10.87 ***	-10.07 ***
Risk Perceptions^2			1.27 ***	1.18 ***
Disposable Income Inequality		0.02	0.02	-0.12 +
Disposable Income Inequality^2				0.19 +
Risk Perceptions in April (paths predicting M)	**M40** *Estimates*	**M41** *Estimates*	**M42** *Estimates*	**M43** *Estimates*
Days Since Curve Inflection	-0.00 **	0.00	0.00	0.00
New Case Rate	0.19 ***	0.20 ***	0.20 ***	0.20 ***
Government Intervention	-0.04	-0.04	-0.04	-0.04
Disposable Income Inequality		0.03 ***	0.03 ***	0.03 ***
Observations	74	74	74	74
R^2 Y	0.675	0.684	0.725	0.734
R^2 M	0.238	0.427	0.427	0.427
AIC	101.010	82.019	73.393	72.890
BIC	133.267	118.884	112.562	114.364
log-Likelihood	-36.505	-25.010	-19.696	-18.445
	+ p<0.15	** p<0.1*	*** p<0.05*	**** p<0.01*

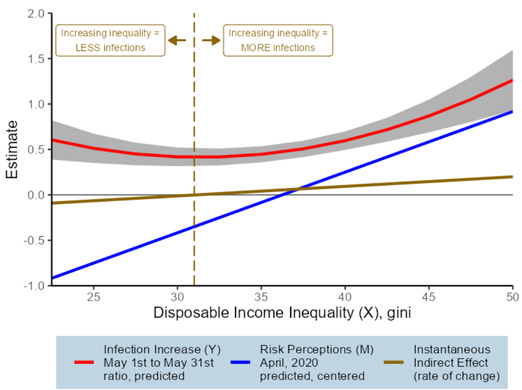
Figure 5

```
m43mat <- summary(m43pfit)
m43mat <- as.data.frame(m43mat[["PE"]])
a43 <- m43mat[38:49,]
a43 <- select(a43, rhs)
a43$rhs <- as.numeric(a43$rhs)
colnames(a43) <- c("X_gini")
b43 <- m43mat[50:61,]
b43 <- select(b43, label, est)
colnames(b43) <- c("moderator","pred_M")
c43 <- m43mat[62:73,]
c43 <- select(c43, est, se, z, pvalue)
colnames(c43) <- c("inst_indr_effect","iie_se","iie_z","iie_p")
d43 <- m43mat[74:85,]
d43 <- select(d43, est, se, z, pvalue)
colnames(d43) <- c("pred_Y","Y_se","Y_z","Y_p")

m43mat <- cbind(a43,b43,c43,d43)

# Now transform so that all values are relative to an 'average', axis cross

# 1 - rescale Gini back into its 100-point original scale
m43mat <- m43mat %>%
  mutate(X_gini = round(X_gini*100,1),
         pred_MC = pred_M - mean(pred_M),
         iie_orig = inst_indr_effect/100,
         iie_origC = iie_orig - mean(iie_orig),
         pred_YavgC = 0.33820*pred_Y, # make avg-Y_hat mean = avg-Y mean
         ymin = pred_YavgC - 1.96*Y_se*0.33820,
         ymax = pred_YavgC + 1.96*Y_se*0.33820)# add 95% confidence intervals)



rm(a43,b43,c43,d43)
```

```
agg_png(file = "results/Fig5.png", width = 800, height = 600, res = 144)
ggplot(data = m43mat, aes(x = X_gini)) +
  geom_abline(intercept = 0, slope = 0, size = 0.25, color = "grey2") +
  geom_ribbon(aes(ymin=ymin,ymax=ymax), fill = "grey70") +
  geom_line(aes(y = pred_YavgC, color = "a"), size = 1) +
  geom_line(aes(y = pred_MC, color = "b"), size = 1) +
  geom_line(aes(y = iie_orig, color = "c"), size = 1) +
  geom_vline(xintercept = 31, linetype = 5, color = "darkgoldenrod4") +
  annotate(geom="label", x=26, y=1.7, label="Increasing inequality =\nLESS infections",
            color="darkgoldenrod4", size = 2.5) +
  annotate(geom="label", x=36, y=1.7, label="Increasing inequality =\nMORE infections",
            color="darkgoldenrod4", size = 2.5) +
  geom_segment(aes(x = 30.5, y = 1.7, xend = 29.5, yend = 1.7),
                arrow = arrow(length = unit(0.2, "cm"), type = "closed"), colour = "darkgoldenrod4") +
  geom_segment(aes(x = 31.5, y = 1.7, xend = 32.5, yend = 1.7),
                arrow = arrow(length = unit(0.2, "cm"), type = "closed"), colour = "darkgoldenrod4") +
  scale_color_manual(name = "", labels = c("a" = "Infection Increase (Y)\nMay 1st to May 31st\nratio, predicted", "c" = "Ins
tantaneous\nIndirect Effect\n(rate of change)" , "b" = "Risk Perceptions (M)\nApril, 2020\npredicted, centered"), values = c
("a" = "red", "b" = "blue", "c" = "darkgoldenrod4")) +
  scale_x_continuous(breaks = c(25,30,35,40,45,50), expand = c(0,0)) +
  scale_y_continuous(breaks = c(-1,-0.5,0,0.5,1,1.5,2), limits = c(-1,2), expand = c(0,0)) +
  labs(x = "Disposable Income Inequality (X), gini",y = "Estimate") +
  theme(panel.background = element_blank(),
        panel.grid = element_blank(),
        axis.line = element_line(),
        legend.position = "bottom",
        legend.background = element_rect(fill = "#BFD5E3"),
        legend.key = element_rect(fill = "#BFD5E3"),
        plot.margin=unit(c(1,1,0.1,0.3),"cm"))
dev.off()
```

```
## png
##   2
```

```
knitr::include_graphics("results/Fig5.png")
```



We know that there is a risk of random sampling variation causing disturbance to the results. Therefore we again simulate the variability in the mean risk perceptions by country.

Get rid of earlier sampling robustness exercise and just do it here for both.

Robustness of Mediation Effect to Sampling. Simulating Plausible Alternative Values for Country-Means

The confidence interval of our regression estimates is based on a sampling distribution across countries, but we have a potentially large source of uncertainty within countries due to the use of an online survey and some very small samples. The online survey problem cannot be solved directly through bootstrapping, but we can asses the robustness of our estimates using the within-country uncertainty.

We 'bootstrap' the estimates by generating random data that follow a normal distribution for each country based on the standard error. Then we run the analysis on each dataset to generate a confidence interval for our estimates that incorporates the within-country standard error of the mean.

Designate the model and calculate the slope of the indirect effect

```
r43 <- '    rate_2 ~ c1*gini_dispR + c2*gini_disp2 + w11*days_since_peak + w12*conf_delta + w13*gov_resp
             concern_self ~ a1*gini_dispR + a2*gini_disp2 + w1*days_since_peak + w2*conf_delta + w3*gov_resp
             rate_2 ~ b1*concern_self + b2*concern_selfsq
             concern_selfsq ~ concern_self
             concern_self ~~ gini_disp2
             concern_self ~ i1*1
             rate_2 ~ i2*1
             a2 == 0
             x1 := .225
             x12 := .50
             predm1 := i1+(a1*x1)+w1*.30+w2*0.37+0*w3
             predm12 := i1+(a1*x12)+w1*.30+w2*0.37+0*w3
             theta1 := (b1+2*b2*predm1)*a1
             theta12 := (b1+2*b2*predm12)*a1
             slope := (theta12-theta1)/(x12-x1)
             yint := theta12 - (slope*x12)
             xint := -yint/slope
         '
```

note with limited RAM I can only do 750 at a time

first 750

```
#
for (i in 1:750) {
  assign(paste0("r43_",i), str_replace_all(r43, "concern_self", test[i]))
}

# fit first 750 models

# get list
model.list <- mget(grep("r43_[0-9]+$", ls(),value=T))

# remove values (clear up RAM)
rm(list = ls(pattern = "r43_"))

# fitting loop
for (m in 1:length(model.list)) {
assign(paste0("fit_", names(model.list[m])), sem(paste0(model.list[m]), data = finaldf_Ca_sim, meanstructure = T, check.grad
ient = F))
}

# get fit list
fit.list <- mget(grep("fit_r43_[0-9]+$", ls(),value=T))
rm(list = ls(pattern = "fit_r43_"))

# extract iie
iie_fit <- matrix(nrow = 2000, ncol = 2)
i <- 1
for (e in 1:750) {
iie_fit[i,1] <- fit.list[[e]]@ParTable[["est"]][44]
iie_fit[i,2] <- fit.list[[e]]@ParTable[["est"]][46]
i <- i + 1
}

rm(fit.list, model.list)
```

751-1500

```
#
for (i in 751:1500) {
  assign(paste0("r43_",i), str_replace_all(r43, "concern_self", test[i]))
}

# fit first 750 models

# get list
model.list <- mget(grep("r43_[0-9]+$", ls(),value=T))

# remove values (clear up RAM)
rm(list = ls(pattern = "r43_"))

# fitting loop
for (m in 1:length(model.list)) {
assign(paste0("fit_", names(model.list[m])), sem(paste0(model.list[m]), data = finaldf_Ca_sim, meanstructure = T, check.grad
ient = F))
}

# get fit list
fit.list <- mget(grep("fit_r43_[0-9]+$", ls(),value=T))
rm(list = ls(pattern = "fit_r43_"))

# extract iie
i <- 751
for (e in 1:750) {
iie_fit[i,1] <- fit.list[[e]]@ParTable[["est"]][44]
iie_fit[i,2] <- fit.list[[e]]@ParTable[["est"]][46]
i <- i + 1
}
rm(fit.list, model.list)
```

1501-2000

```
#
for (i in 1501:2000) {
  assign(paste0("r43_",i), str_replace_all(r43, "concern_self", test[i]))
}

# fit first 750 models

# get list
model.list <- mget(grep("r43_[0-9]+$", ls(),value=T))

# remove values (clear up RAM)
rm(list = ls(pattern = "r43_"))

# fitting loop
for (m in 1:length(model.list)) {
assign(paste0("fit_", names(model.list[m])), sem(paste0(model.list[m]), data = finaldf_Ca_sim, meanstructure = T, check.grad
ient = F))
}

# get fit list
fit.list <- mget(grep("fit_r43_[0-9]+$", ls(),value=T))
rm(list = ls(pattern = "fit_r43_"))

# extract iie
i <- 1501
for (e in 1:500) {
iie_fit[i,1] <- fit.list[[e]]@ParTable[["est"]][44]
iie_fit[i,2] <- fit.list[[e]]@ParTable[["est"]][46]
i <- i + 1
}
rm(fit.list, model.list)

iie_fit <- as.data.frame(iie_fit)
colnames(iie_fit) <- c("slope","intercept")

# put into original metric
iie_fit$slope <- round(iie_fit$slope/100, 3)
iie_fit$intercept <- round(iie_fit$intercept*100, 3)

# get standard error
rob1 <- apply(iie_fit, 2, mean)
rob2 <- apply(iie_fit, 2, sd)
rob3 <- c(NA,NA)
rob <- rbind(rob1, rob2, rob3)

rob[3,1] <- rob[2,1]/sqrt(2000)
rob[3,2] <- rob[2,2]/sqrt(2000)
m43slope <- (m43pfit@ParTable[["est"]][73]-m43pfit@ParTable[["est"]][72])/(.5 - .475)
m43yint <- m43pfit@ParTable[["est"]][73] - m43slope*.5
m43xint <- (-m43yint/m43slope)*100
m43slope <- m43slope/100
```

For the instantaneous indirect effects, the robust mean slope is 0.711763 with a standard error of 0.0094763 and the robust mean intercept is 29.7752245 with a standard error of 0.1183316. Original slope is 1.0543015 and x-int 31.1193831

References

Hayes, Andrew F., and Kristopher J. Preacher. 2010. “Quantifying and Testing Indirect Effects in Simple Mediation Models When the Constituent Paths Are Nonlinear.” Multivariate Behavioral Research 45(4):627–60. Shared Copy (http://quantpsy.org/pubs/hayes_preacher_2010.pdf)