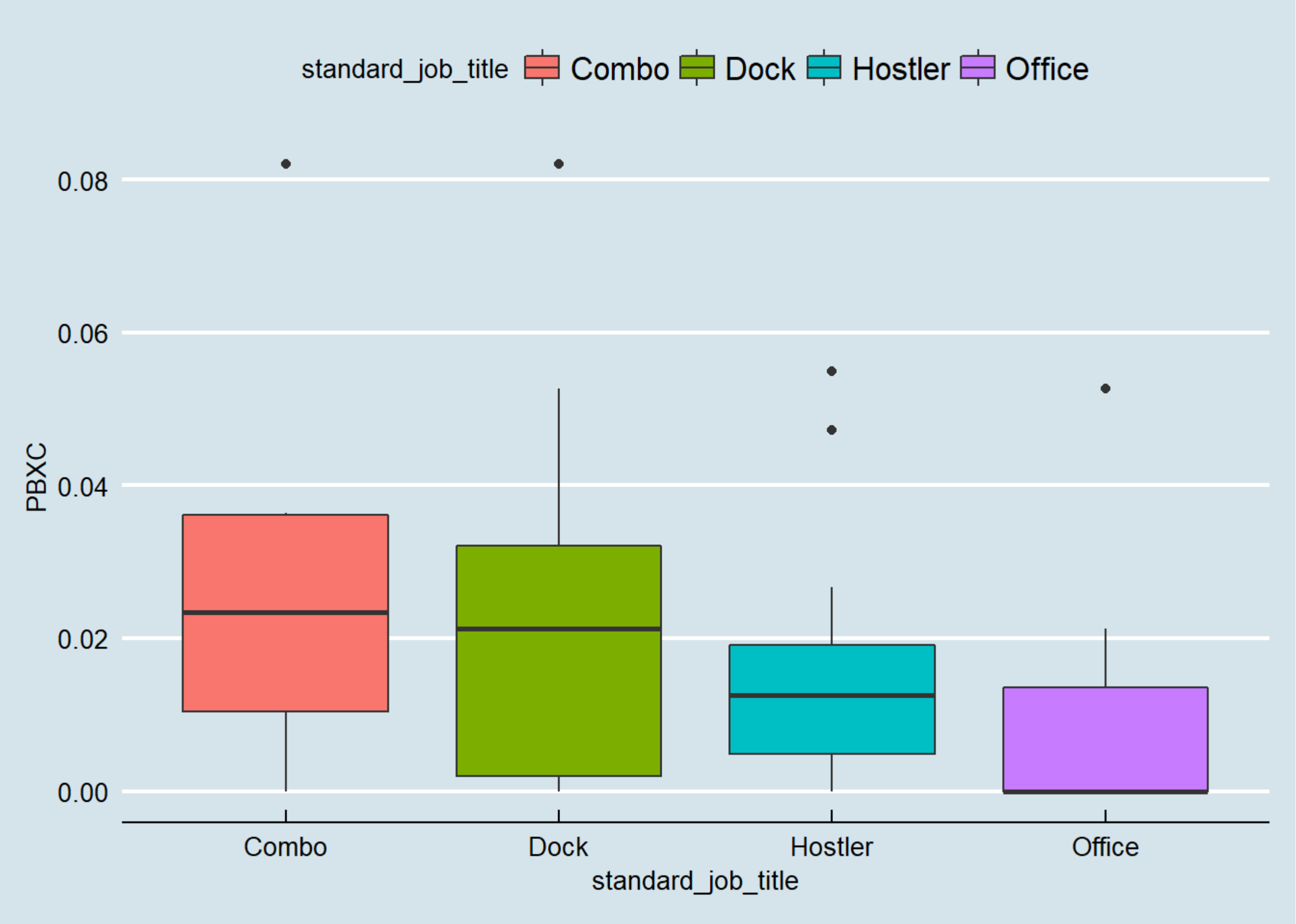
Final Project

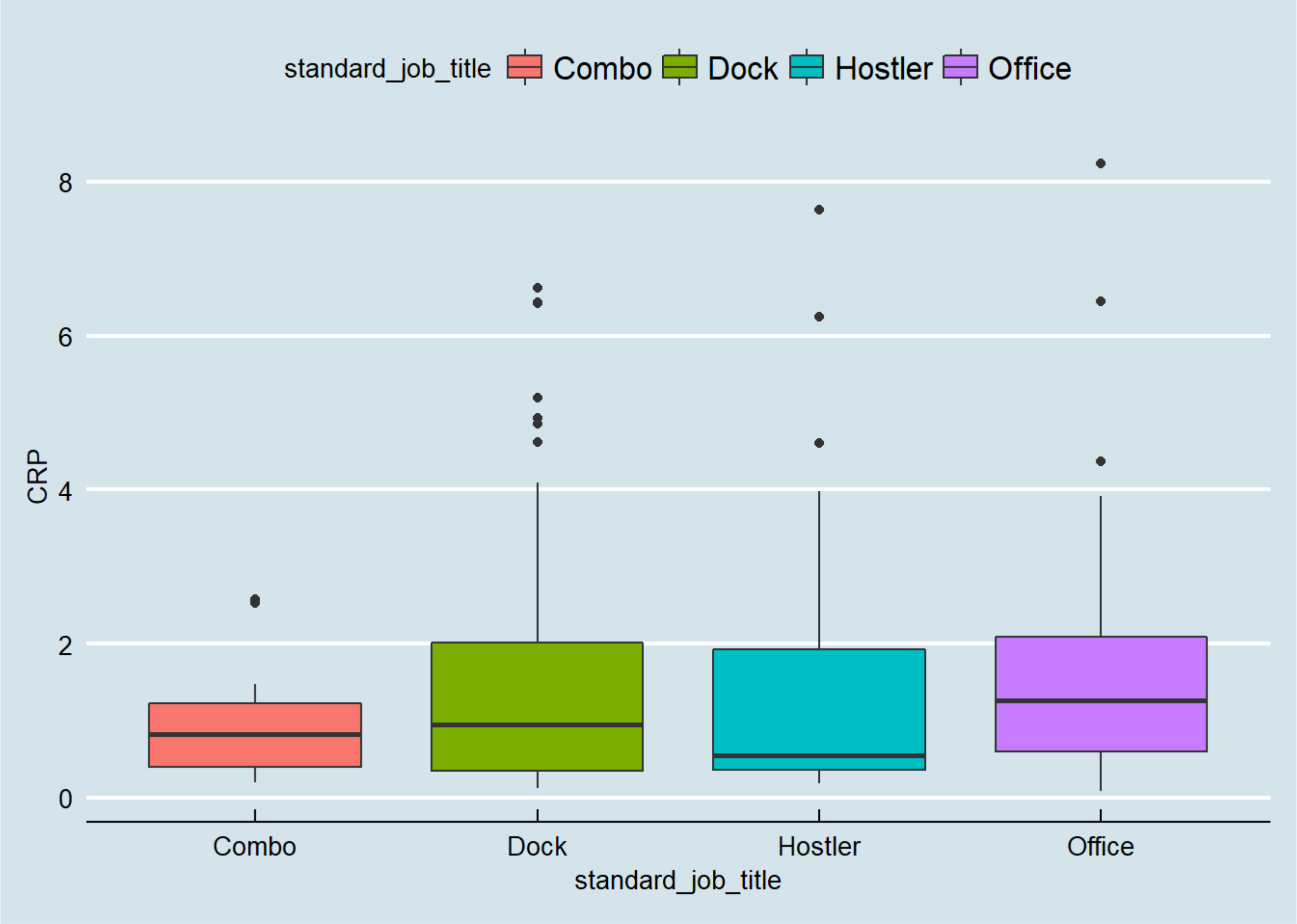
Andrew Shapero

```
biomarkers <- read_excel ("xfr_with_inflamm.xls")
metals <- read_excel ("XRF_results_HVELX63X.xls", sheet = 3)
data <- left_join (biomarkers, metals, by = "sampleid")
```

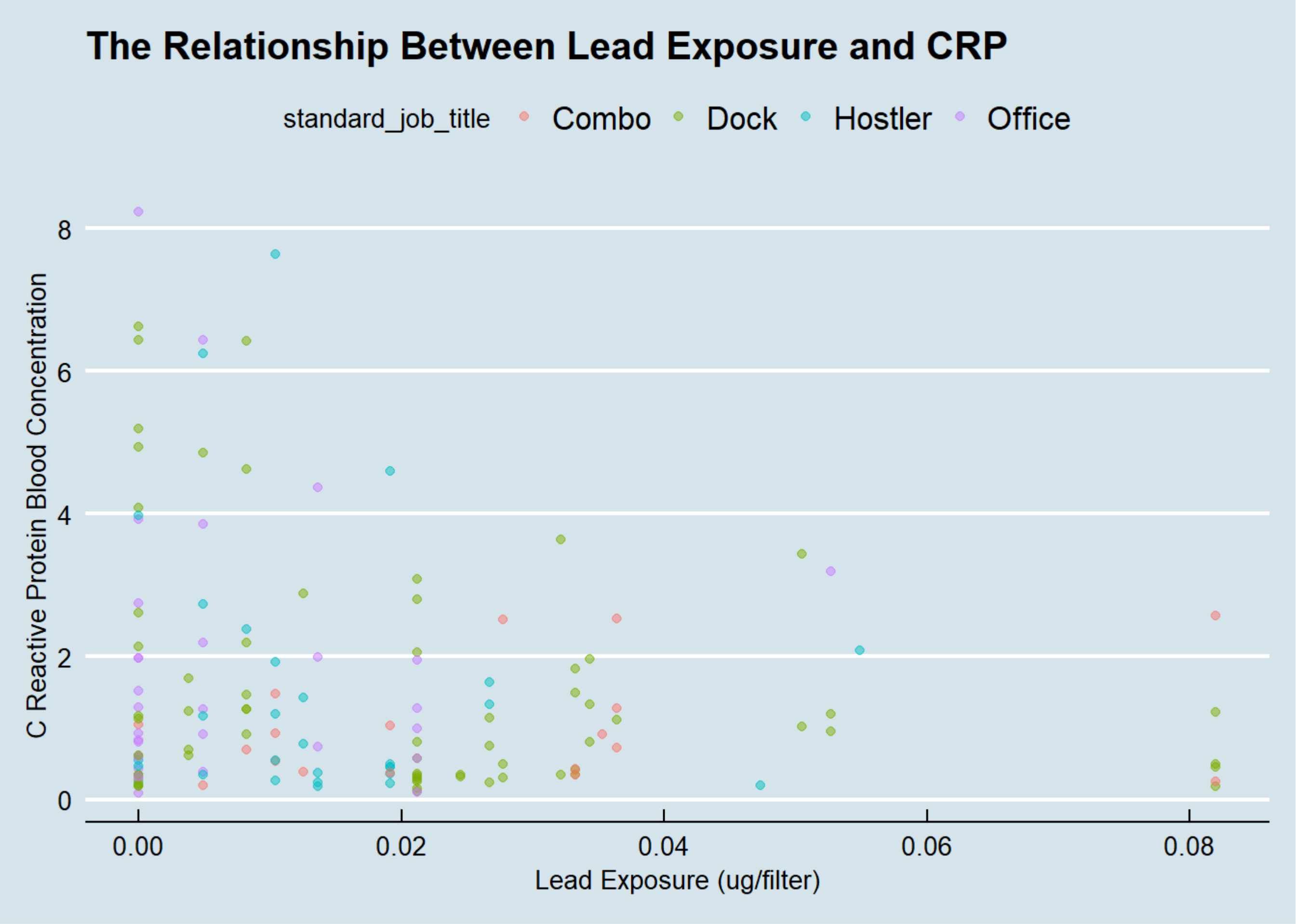
```
data %>% ggplot (aes (x = standard_job_title, y = PBXC, fill = standard_job_title)) +
          geom_boxplot () +
          theme_economist ()
```



```
data %>% ggplot (aes (x = standard_job_title, y = CRP, fill = standard_job_title)) +
          geom_boxplot () +
          theme_economist ()
```



```
p<- data %>% ggplot (aes (x = PBXC , y = CRP, col = standard_job_title)) +
          geom_point(alpha = 0.5) +
          xlab ("Lead Exposure (ug/filter)") +
          ylab ("C Reactive Protein Blood Concentration ") +
          ggtitle ("The Relationship Between Lead Exposure and CRP") +
          theme_economist()
p
```

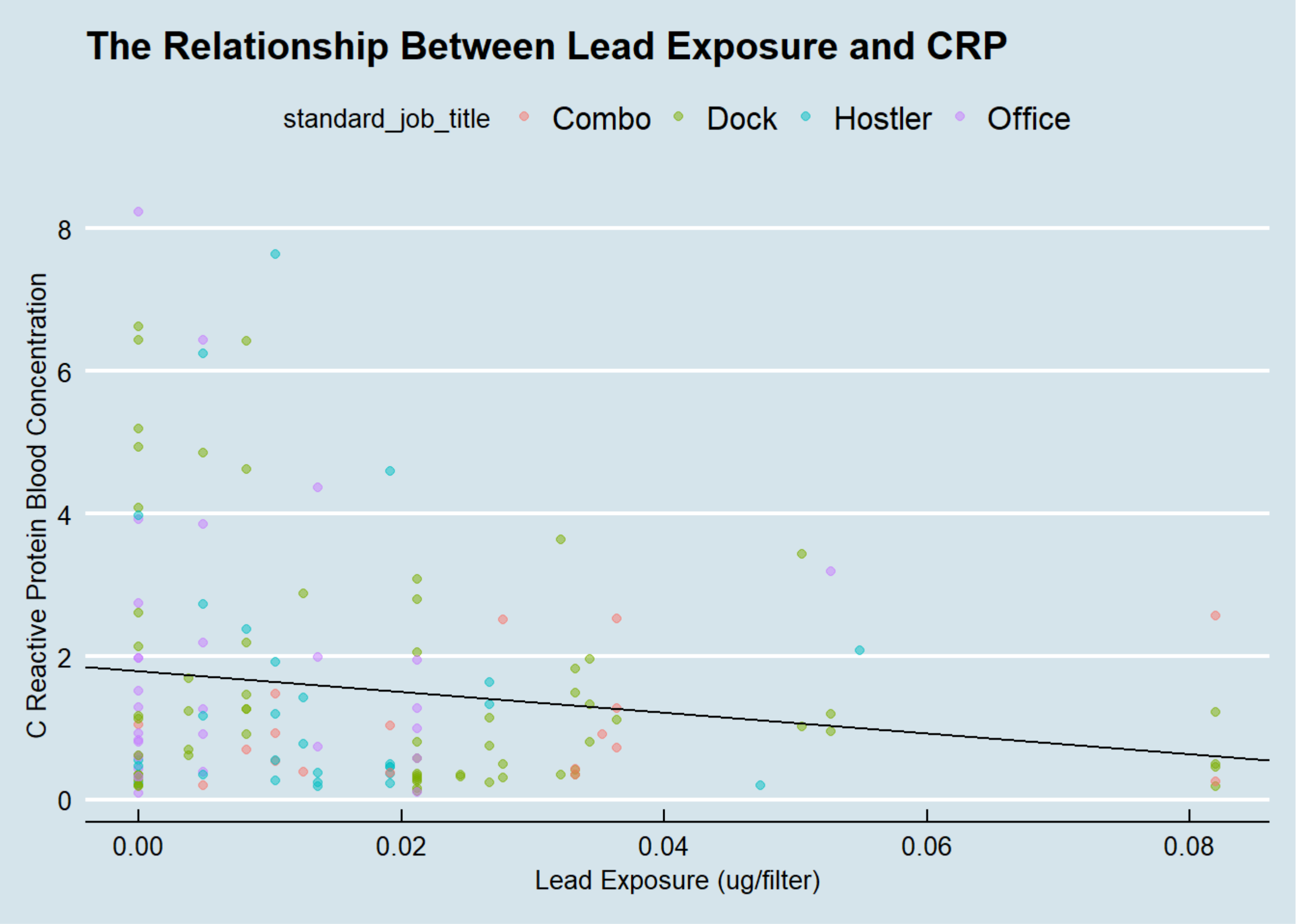


```
fit <- lm (CRP ~ PBXC, data = data, na.rm = TRUE)
fit <- tidy(fit)
fit
```

```
##          term   estimate std.error statistic      p.value
## 1 (Intercept)   1.795326 0.1871858  9.591147 3.972930e-17
## 2        PBXC -14.505816 7.1653893 -2.024428 4.478785e-02
```

```
int <- fit$estimate [1]
m <- fit$estimate [2]
```

```
p + geom_abline ( intercept = int, slope = m)
```



So there's the data for one metal. We also want to account for potential confounders. However, we also want to make sure we can look at other metals in the dataset. To do that, I'm going to make a **long** dataset instead of a **wide** dataset. In this case, we can look at all metals at the same time.

```
tidy_data <- data %>% gather (Code, concentration, `NAXC`:`URXU`)
```

Each metal reading has a concentration and an error estimate. Let's get rid of the error estimates for now, as the actual readings are our best estimates of exposure. Each of the estimates ends in "XC". These are the data points we want to keep in our data frame.

```
tidy_data <- tidy_data %>% filter (str_sub (Code, -2) == "XC")
```

Let's also rename the metals, so that they correspond to actual metal names. I made an Excel sheet to decipher each of the codes. Let's read that in and then translate the metal codes to the actual metal names.

```
metal_codes <- read_csv ("metal_codes.csv")
```

```
## Parsed with column specification:
## cols(
##   Code = col_character(),
##   Metal = col_character()
## )
```

```
tidy_data <- left_join(tidy_data, metal_codes, by = "Code")
```

Now let’s run a regression for every metal.

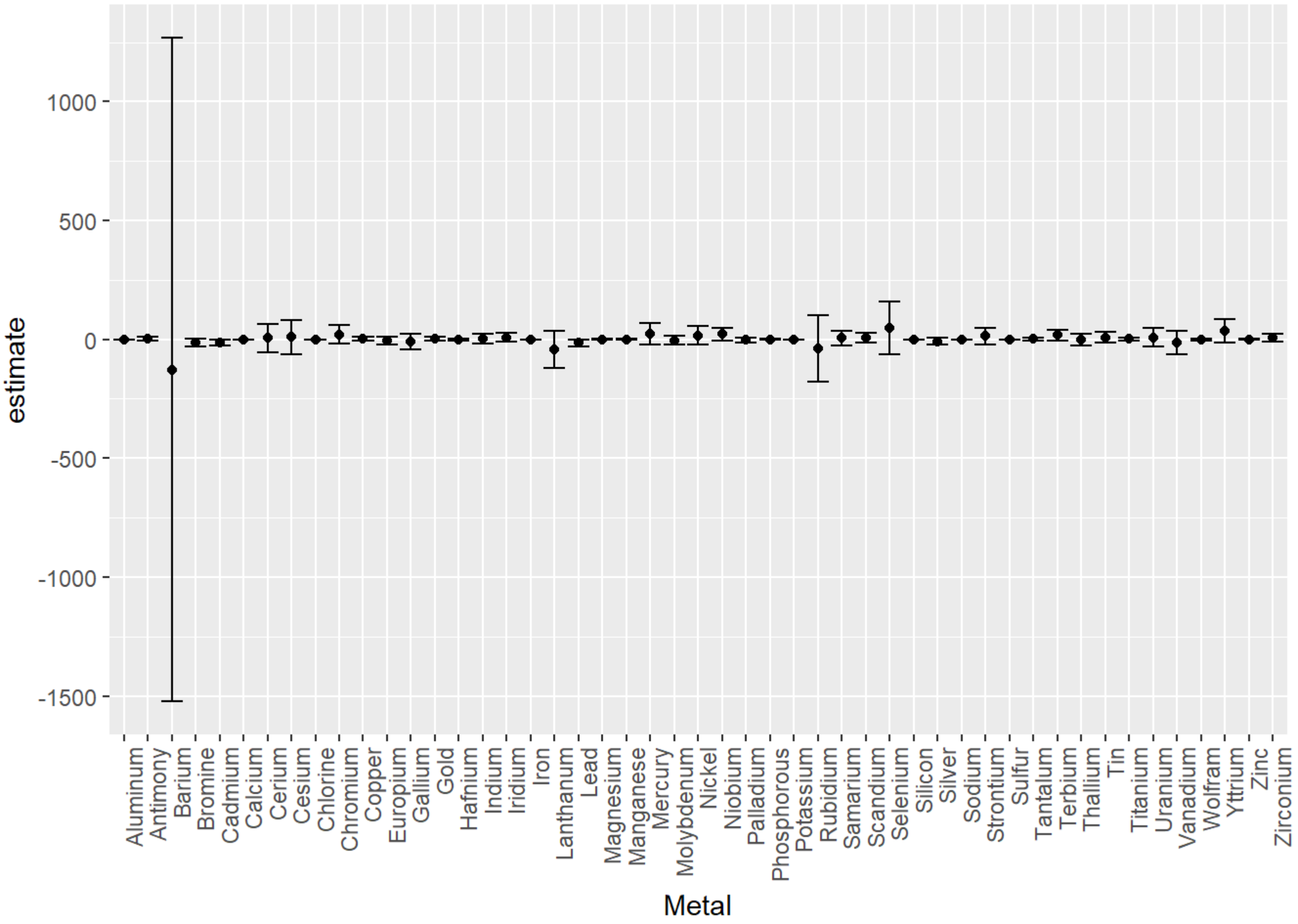
```
reg <- tidy_data %>% group_by (Metal) %>%
  do (tidy (lm (CRP ~ concentration, data = .), conf.int = TRUE))
```

Let’s now filter out all the intercepts. We’re not necessarily interested in those.

```
reg <- reg %>% filter (term != "(Intercept)")
```

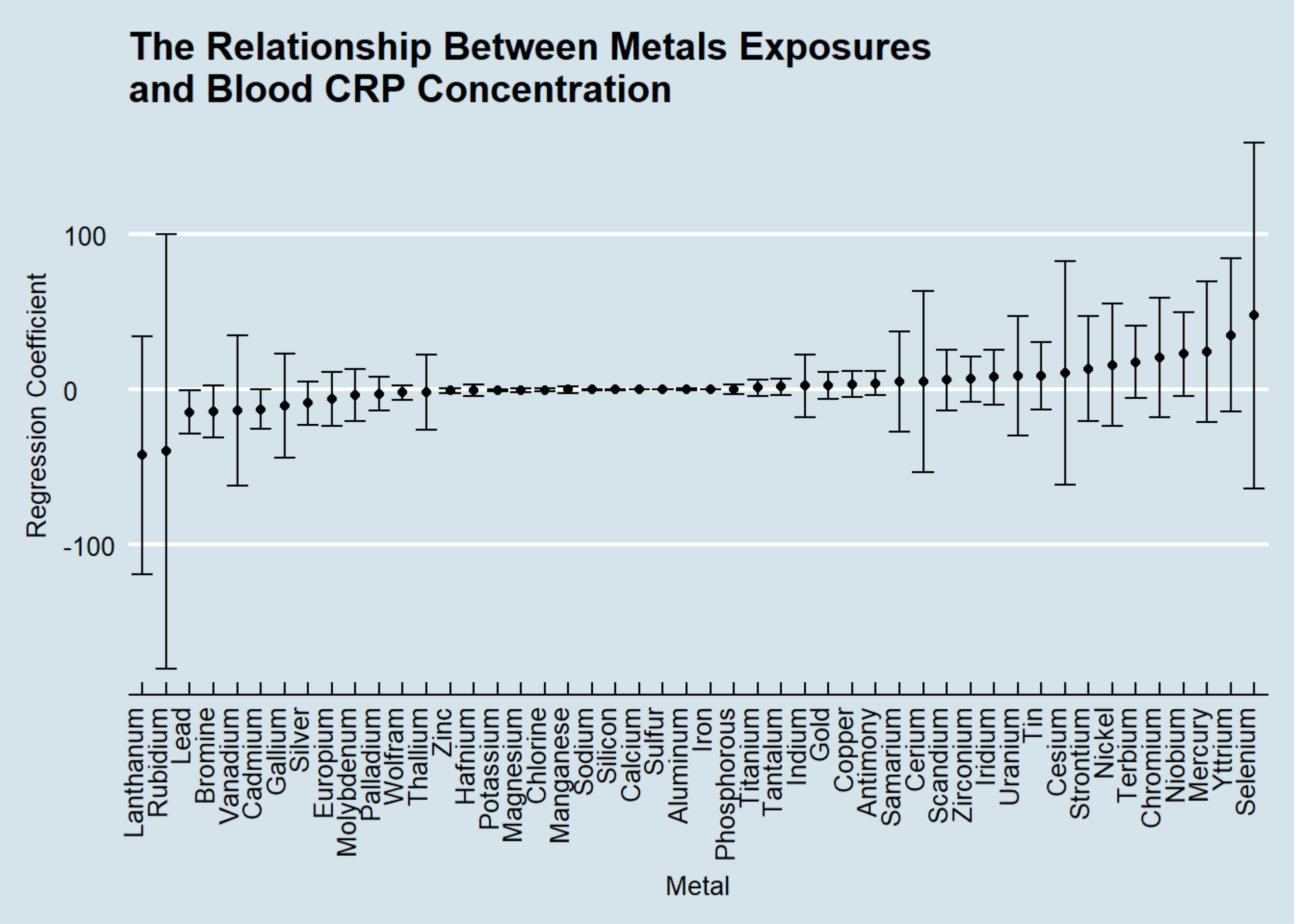
Now we can show the confidence intervals for each of the metals.

```
reg %>%  ggplot (aes (x = Metal, y = estimate, ymin = conf.low, ymax = conf.high)) +
    geom_errorbar () +
    geom_point () +
    theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



The confidence interval for barium is way too wide. Let’s filter that out, as it is obscuring the other confidence intervals.

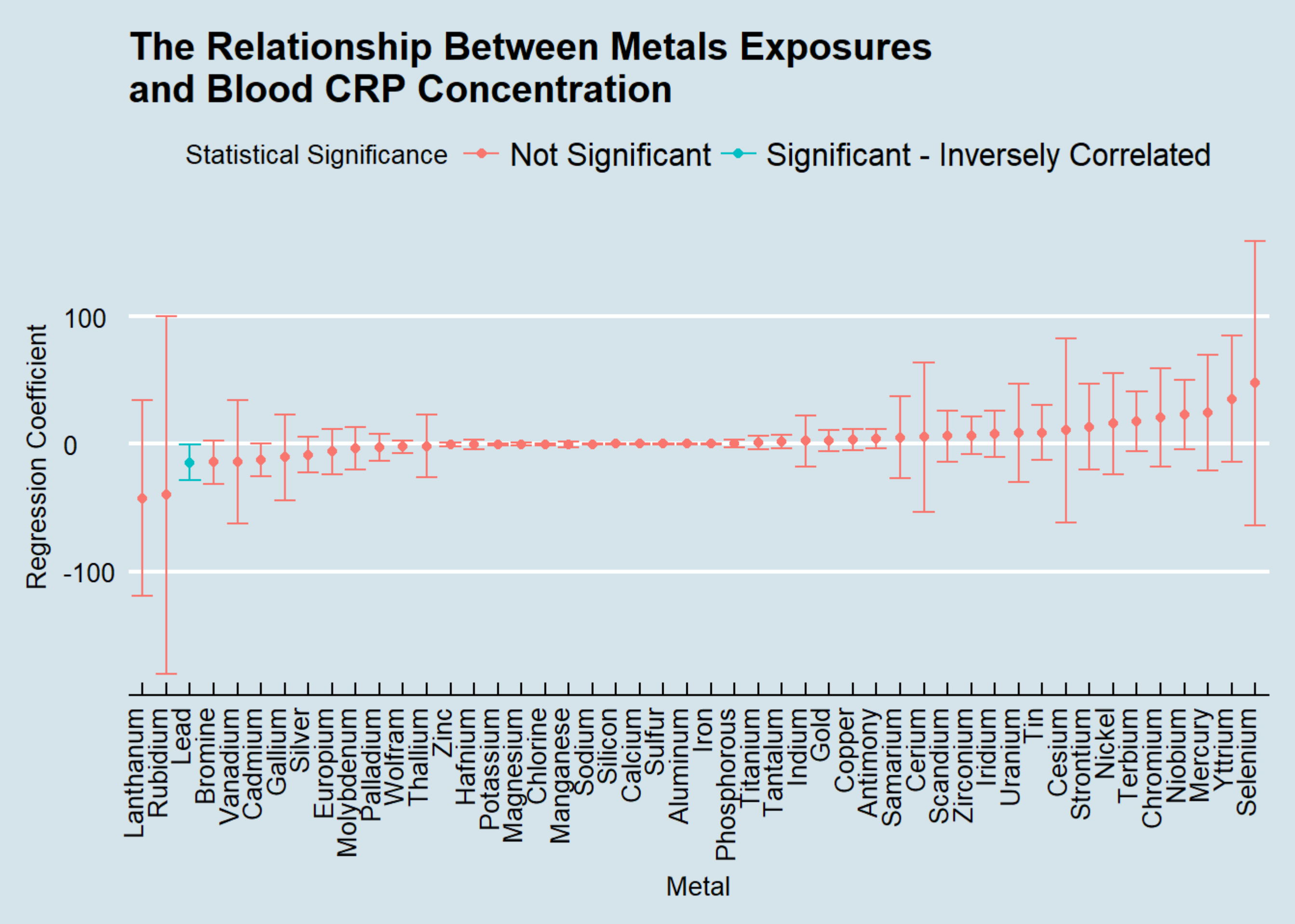
```
reg %>% filter (Metal != "Barium") %>%
  ggplot ( aes ( x = reorder (Metal, estimate), y = estimate, ymin = conf.low, ymax = conf.high)
) +
    geom_errorbar () +
    geom_point () +
    xlab ("Metal") +
    ylab ("Regression Coefficient") +
    ggtitle ("The Relationship Between Metals Exposures\nand Blood CRP Concentration") +
    theme_economist () +
    theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Let’s now create a color code so we can see if any of the relationships are statistically significant.

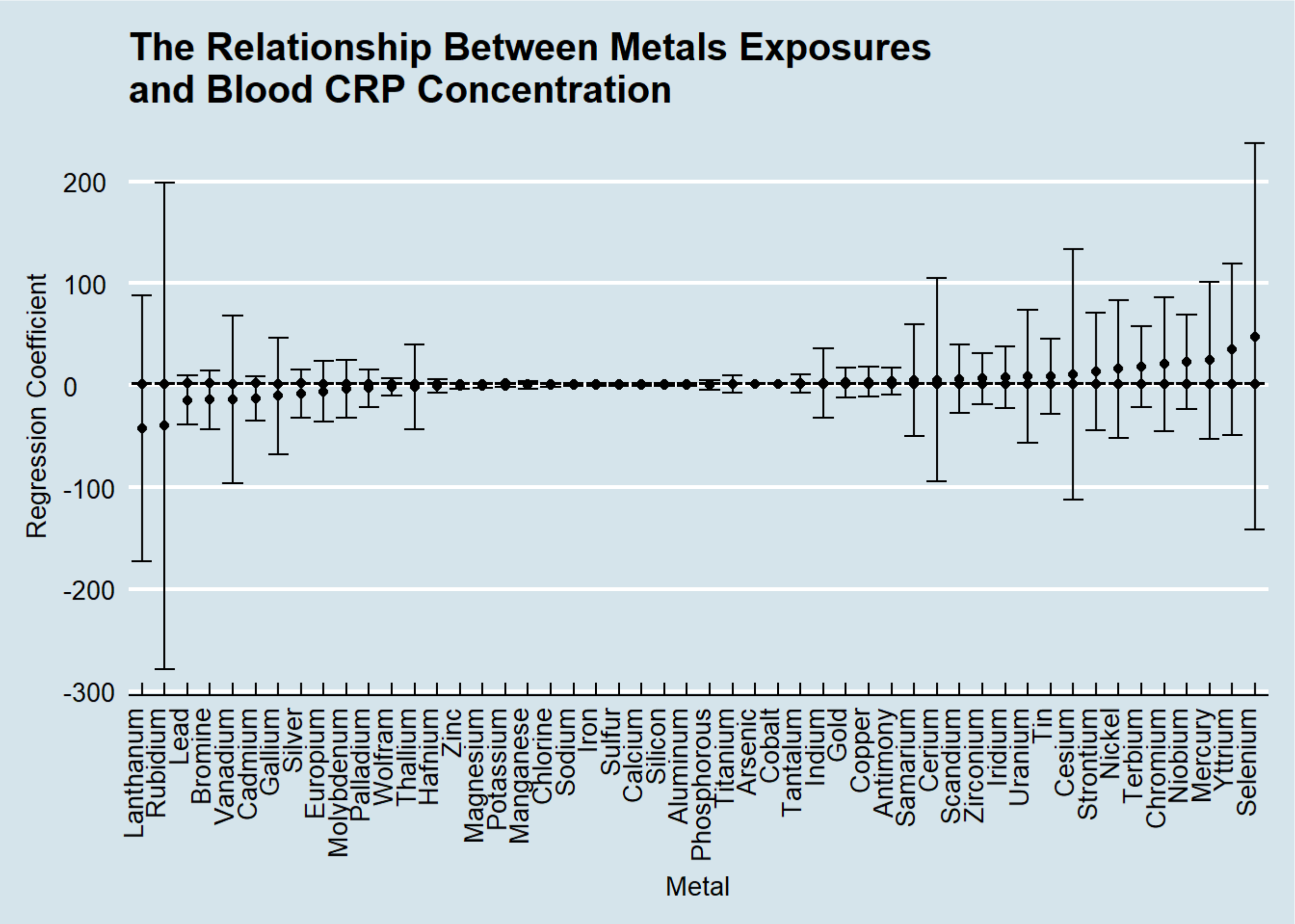
```
reg <- reg %>% mutate (
  sig = ifelse(conf.high < 0 , "Significant - Inversely Correlated",
             ifelse(conf.high >0 | conf.low <0, "Not Significant",
                    ifelse(conf.low > 0, "Significant - Positively Correlated" , NA)))
)
```

```
reg %>% filter (Metal != "Barium") %>%
  ggplot (aes (x = reorder (Metal, estimate), y = estimate, ymin = conf.low, ymax = conf.high, c
ol = sig)) +
    geom_errorbar () +
    geom_point () +
    xlab ("Metal") +
    ylab ("Regression Coefficient") +
    ggtitle ("The Relationship Between Metals Exposures\nand Blood CRP Concentration") +
    theme_economist () +
    theme (axis.text.x = element_text (angle = 90, hjust = 1)) +
    guides (col = guide_legend (title = "Statistical Significance"))
```



Now let’s apply a Bonferroni correction to account for multiple testing, given that we are looking at 51 different metals.

```
alpha <- 0.05 / 51
conf_level <- 1 - alpha
reg2 <- tidy_data %>% group_by (Metal) %>%
  do (tidy (lm (CRP ~ concentration, data = .), conf.int = TRUE, conf.level = conf_level))
reg <- reg %>% filter (term != "(Intercept)")
reg2 <- reg2 %>% mutate (
  sig = ifelse(conf.high < 0 , "Significant - Inversely Correlated",
               ifelse(conf.high >0 | conf.low <0, "Not Significant",
                      ifelse(conf.low > 0, "Significant - Positively Correlated" , NA)))
)
reg2 %>% filter (Metal != "Barium") %>%
  ggplot ( aes ( x = reorder (Metal, estimate), y = estimate, ymin = conf.low, ymax = conf.high)
) +
    geom_errorbar () +
    geom_point () +
    xlab ("Metal") +
    ylab ("Regression Coefficient") +
    ggtitle ("The Relationship Between Metals Exposures\nand Blood CRP Concentration") +
    theme_economist () +
    theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Now we need to adjust for covariates.