Association between policy and fair chance job posting rates

Zofia C. Stanley

2024-11-19

```r
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# Set parameter to overwrite plots
overwrite_plots = TRUE

# Read in the data
policy_data <- read.csv("../Controls/regression.csv")
employment_data <- read.csv("../Employment/Indeed/indeed-retail-foodservice-fairchance.csv")
validation_employment_data <- read.csv("../Employment/Indeed/indeed-complete-industry.csv")
control_data <- read.csv("../Controls/control_indices.csv")
employment_control_data <- read.csv("../Controls/control-vars-employment.csv")
```

```r
policy_data <- policy_data[c("state", "tot_pos_index", "neg_combo_index")] %>%
  rename(state = case_when("State" %in% names(.) ~ "State", TRUE ~ "state")) %>%
  rename(pos_pol_index = tot_pos_index, neg_pol_index = neg_combo_index)

employment_data <- employment_data[c("state", "fair_chance_rate")] %>%
  rename(state = case_when("State" %in% names(.) ~ "State", TRUE ~ "state")) %>%
  rename(original_fair_chance_rate = fair_chance_rate)

validation_employment_data <- validation_employment_data %>%
  group_by(state) %>%
  summarize(
    total_jobs = sum(totalJobs, na.rm = TRUE),
    total_fair_chance_jobs = sum(totalJobs_fairChance, na.rm = TRUE)
  ) %>%
  mutate(validation_fair_chance_rate = (total_fair_chance_jobs / total_jobs) * 1000) %>%
  rename(state = case_when("State" %in% names(.) ~ "State", TRUE ~ "state")) %>%
  select(state, validation_fair_chance_rate)

control_data <- control_data %>%
  rename(state = case_when("State" %in% names(.) ~ "State", TRUE ~ "state")) %>%
```

```r
  rename_with(
    .cols = -state, # Exclude the "state" column
    .fn = ~ paste0(.x, "_control") # Append "_control" to the column names
  )

# Add columns for original and validation
employment_control_data <- employment_control_data %>%
  mutate(
    original_employment_control = rtrade_employment + arts_employment,
    validation_employment_control = rowSums(select(., construct_employment:arts_employment), na.rm = TRU
  ) %>%
  rename(state = case_when("State" %in% names(.) ~ "State", TRUE ~ "state")) %>%
  select(state, original_employment_control, validation_employment_control)

# Sequentially merge all cleaned data frames by "state"
full_data <- policy_data %>%
  full_join(employment_data, by = "state") %>%
  full_join(validation_employment_data, by = "state") %>%
  full_join(control_data, by = "state") %>%
  full_join(employment_control_data, by = "state")

# Remove all objects except "full_data"
rm(list = setdiff(ls(), c("full_data", "overwrite_plots")))
```

Introduction

This analysis aims to explore the relationship between positive policies, collateral consequences, and fair chance hiring outcomes in retail and food service industries. We use scatter plots, linear regression models, and chi-square tests to understand these relationships.

Compute Residuals: Throughout this analysis, we will use residualized quantities to understand the relationship between variables, after controlling for confounding factors. We calculate all residuals up front here.

```r
# Fit the linear models and extract residuals for each variable

# Model for neg_pol_index with original controls
lm_original_neg <- lm(neg_pol_index ~ macroeconomy_control + socioeconomics_control +
               politics_control + diversity_control + original_employment_control,
             data = full_data)
residuals_original_neg <- resid(lm_original_neg)

# Model for pos_pol_index with original controls
lm_original_pos <- lm(pos_pol_index ~ macroeconomy_control + socioeconomics_control +
               politics_control + diversity_control + original_employment_control,
             data = full_data)
residuals_original_pos <- resid(lm_original_pos)
```

```r
# Model for original_fair_chance_rate
lm_original_fair <- lm(original_fair_chance_rate ~ macroeconomy_control + socioeconomics_control +
                         politics_control + diversity_control + original_employment_control,
                       data = full_data)
residuals_original_fair <- resid(lm_original_fair)

# Model for neg_pol_index with validation controls
lm_validation_neg <- lm(neg_pol_index ~ macroeconomy_control + socioeconomics_control +
               politics_control + diversity_control + validation_employment_control,
             data = full_data)
residuals_validation_neg <- resid(lm_validation_neg)

# Model for pos_pol_index with validation controls
lm_validation_pos <- lm(pos_pol_index ~ macroeconomy_control + socioeconomics_control +
               politics_control + diversity_control + validation_employment_control,
             data = full_data)
residuals_validation_pos <- resid(lm_validation_pos)

# Model for validation_fair_chance_rate
lm_validation_fair <- lm(validation_fair_chance_rate ~ macroeconomy_control + socioeconomics_control +
                           politics_control + diversity_control + validation_employment_control,
                         data = full_data)
residuals_validation_fair <- resid(lm_validation_fair)

# Create the residuals data frame
residuals <- data.frame(
  state = full_data$state,
  original_neg_pol_index = residuals_original_neg,
  original_pos_pol_index = residuals_original_pos,
  original_fair_chance_rate = residuals_original_fair,
  validation_neg_pol_index = residuals_validation_neg,
  validation_pos_pol_index = residuals_validation_pos,
  validation_fair_chance_rate = residuals_validation_fair
)

# Remove all objects except "full_data" and "residuals"
rm(list = setdiff(ls(), c("full_data", "residuals", "overwrite_plots")))
```

Scatter Plot: Positive Policies vs Collateral Consequences

We begin by visualizing the relationship between positive policies and collateral consequences.

```r
# Create a scatter plot of positive policy vs negative policy
plot_pos_vs_neg <- ggplot(full_data, aes(x = neg_pol_index, y = pos_pol_index)) +
  geom_point() +
  geom_smooth(method = "lm", formula = y ~ 0 + x, se = FALSE) +
  labs(
    title = "Positive vs Negative Policy Indices",
    x = "Negative Policy Index",
    y = "Positive Policy Index"
  ) +
  xlim(0, 900) +
```

```r
  ylim(0, 1) +
  theme_minimal()

if (overwrite_plots) {
  pdf("scatter_pos_neg.pdf")
  print(plot_pos_vs_neg)
  dev.off()
}
```

```
## pdf
##   2
```

```r
print(plot_pos_vs_neg)
```

Interpretation: The scatter plot shows the relationship between the weighted number of collateral consequences and the positive policy index. Visually, we see a strong positive association between the weighted number of collateral consequences and positive policy index.

```r
# Perform a correlation analysis
correlation_result <- cor.test(
  full_data$neg_pol_index,
  full_data$pos_pol_index,
  method = "pearson"
)

# Print the correlation result
print(correlation_result)
```

```
##
##  Pearson's product-moment correlation
##
## data:  full_data$neg_pol_index and full_data$pos_pol_index
## t = 2.8819, df = 48, p-value = 0.005896
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1183657 0.5984337
## sample estimates:
##       cor
## 0.3840599
```

Interpretation: There is a statistically significant moderate positive relationship between the Negative Policy Index and the Positive Policy Index ($r = 0.384$, $p = 0.0059$). This suggests that as the Negative Policy Index increases, the Positive Policy Index tends to increase as well, albeit moderately.

The confidence interval indicates that the strength of the correlation is likely between 0.118 and 0.598, suggesting some variability in the true relationship but consistently positive.

```r
custom_correlation_test <- function(x, y, df_reduction) {
  # Remove rows with missing values in x or y
  complete_cases <- complete.cases(x, y)
  x <- x[complete_cases]
  y <- y[complete_cases]

  # Number of observations
  n <- length(x)

  # Compute Pearson correlation coefficient
  r <- cor(x, y, method = "pearson")

  # Adjusted degrees of freedom
  df <- n - 2 - df_reduction

  # Compute the t-statistic
  t_stat <- r * sqrt(df / (1 - r^2))

  # Compute the p-value for the two-tailed test
  p_value <- 2 * pt(-abs(t_stat), df)

  # Compute 95% confidence intervals for r
  alpha <- 0.05
  z <- qt(1 - alpha / 2, df)
  ci_lower <- tanh(atanh(r) - z * sqrt(1 / (df - 1)))
  ci_upper <- tanh(atanh(r) + z * sqrt(1 / (df - 1)))

  # Return results as a list
  return(list(
    correlation = r,
    degrees_of_freedom = df,
    t_statistic = t_stat,
    p_value = p_value,
    confidence_interval = c(ci_lower, ci_upper)
  ))
}
```

```r
# Number of predictors used in the models
df_reduction <- 5

# Call the custom correlation test
result <- custom_correlation_test(
  residuals$original_neg_pol_index,
  residuals$original_pos_pol_index,
  df_reduction
)

# Print the results
cat("Custom Correlation Analysis of Residuals:\n")
```

```
## Custom Correlation Analysis of Residuals:
```

```r
cat("Correlation Coefficient (r):", result$correlation, "\n")
```

```
## Correlation Coefficient (r): 0.2043163
```

```r
cat("Adjusted Degrees of Freedom (df):", result$degrees_of_freedom, "\n")
```

```
## Adjusted Degrees of Freedom (df): 43
```

```r
cat("t-Statistic:", result$t_statistic, "\n")
```

```
## t-Statistic: 1.368664
```

```r
cat("p-value:", result$p_value, "\n")
```

```
## p-value: 0.1782128
```

```r
cat("95% Confidence Interval: [", result$confidence_interval[1], ",", result$confidence_interval[2], "]"
```

```
## 95% Confidence Interval: [ -0.1035768 , 0.4764759 ]
```

```r
# Perform a correlation analysis
correlation_result <- cor.test(
  residuals$original_neg_pol_index,
  residuals$original_pos_pol_index,
  method = "pearson"
)

# Print the correlation result
print(correlation_result)
```

```
##
##  Pearson's product-moment correlation
##
## data:  residuals$original_neg_pol_index and residuals$original_pos_pol_index
## t = 1.446, df = 48, p-value = 0.1547
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.07849554  0.45669152
## sample estimates:
##       cor
## 0.2043163
```

Fit Linear Models: Fair Chance Job Posting Rate vs Policy Indices

```r
# Fit linear models for original_fair_chance_rate
lm_original_fair_pos <- lm(original_fair_chance_rate ~ pos_pol_index, data = full_data)
lm_original_fair_neg <- lm(original_fair_chance_rate ~ neg_pol_index, data = full_data)

# Fit linear models for validation_fair_chance_rate
lm_validation_fair_pos <- lm(validation_fair_chance_rate ~ pos_pol_index, data = full_data)
lm_validation_fair_neg <- lm(validation_fair_chance_rate ~ neg_pol_index, data = full_data)

# Summarize results for original_fair_chance_rate
cat("Model: original_fair_chance_rate ~ pos_pol_index\n")
```

```
## Model: original_fair_chance_rate ~ pos_pol_index
```

```r
summary(lm_original_fair_pos)
```

```
##
## Call:
## lm(formula = original_fair_chance_rate ~ pos_pol_index, data = full_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.394  -9.182  -2.762   9.659  24.781
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     39.319      3.604   10.91 1.36e-14 ***
## pos_pol_index  -21.895      7.792   -2.81  0.00715 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.81 on 48 degrees of freedom
## Multiple R-squared:  0.1413, Adjusted R-squared:  0.1234
## F-statistic: 7.897 on 1 and 48 DF,  p-value: 0.007146
```

```r
cat("\nModel: original_fair_chance_rate ~ neg_pol_index\n")
```

```
##
## Model: original_fair_chance_rate ~ neg_pol_index
```

```r
summary(lm_original_fair_neg)
```

```
##
## Call:
## lm(formula = original_fair_chance_rate ~ neg_pol_index, data = full_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.3513 -10.5852  -0.6338  11.3975  21.7300
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   43.48701    4.75505   9.145 4.34e-12 ***
## neg_pol_index -0.03374    0.01144  -2.949  0.00491 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.73 on 48 degrees of freedom
## Multiple R-squared:  0.1534, Adjusted R-squared:  0.1358
## F-statistic: 8.698 on 1 and 48 DF,  p-value: 0.004911
```

```r
# Summarize results for validation_fair_chance_rate
cat("\nModel: validation_fair_chance_rate ~ pos_pol_index\n")
```

```
##
## Model: validation_fair_chance_rate ~ pos_pol_index
```

```r
summary(lm_validation_fair_pos)
```

```
##
## Call:
## lm(formula = validation_fair_chance_rate ~ pos_pol_index, data = full_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.916  -7.297  -4.014   7.095  27.214
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     34.189      3.359  10.178 1.42e-13 ***
## pos_pol_index  -19.800      7.261  -2.727  0.00891 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.01 on 48 degrees of freedom
## Multiple R-squared:  0.1341, Adjusted R-squared:  0.1161
## F-statistic: 7.435 on 1 and 48 DF,  p-value: 0.008906
```

```r
cat("\nModel: validation_fair_chance_rate ~ neg_pol_index\n")
```

```
##
## Model: validation_fair_chance_rate ~ neg_pol_index
```

```r
summary(lm_validation_fair_neg)
```

```
##
## Call:
## lm(formula = validation_fair_chance_rate ~ neg_pol_index, data = full_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.757  -7.114  -2.119   2.968  26.786
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   38.07318    4.42596   8.602 2.75e-11 ***
## neg_pol_index -0.03081    0.01065  -2.893  0.00572 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.92 on 48 degrees of freedom
## Multiple R-squared:  0.1485, Adjusted R-squared:  0.1307
```

```
## F-statistic: 8.369 on 1 and 48 DF,  p-value: 0.005722
```

```r
# Fit linear models for residuals$original_fair_chance_rate
lm_residuals_original_fair_pos <- lm(original_fair_chance_rate ~ original_pos_pol_index, data = residual
lm_residuals_original_fair_neg <- lm(original_fair_chance_rate ~ original_neg_pol_index, data = residual

# Fit linear models for residuals$validation_fair_chance_rate
lm_residuals_validation_fair_pos <- lm(validation_fair_chance_rate ~ original_pos_pol_index, data = resi
lm_residuals_validation_fair_neg <- lm(validation_fair_chance_rate ~ original_neg_pol_index, data = resi

# Summarize results for residuals$original_fair_chance_rate
cat("Model: residuals$original_fair_chance_rate ~ residuals$original_pos_pol_index\n")
```

```
## Model: residuals$original_fair_chance_rate ~ residuals$original_pos_pol_index
```

```r
summary(lm_residuals_original_fair_pos)
```

```
##
## Call:
## lm(formula = original_fair_chance_rate ~ original_pos_pol_index,
##     data = residuals)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.6711  -7.4332  -0.7706   8.1041  22.0614
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -7.979e-16  1.497e+00   0.000    1.000
## original_pos_pol_index -1.075e+01  7.833e+00  -1.373    0.176
##
## Residual standard error: 10.58 on 48 degrees of freedom
## Multiple R-squared:  0.03779,    Adjusted R-squared:  0.01775
## F-statistic: 1.885 on 1 and 48 DF,  p-value: 0.1761
```

```r
cat("\nModel: residuals$original_fair_chance_rate ~ residuals$original_neg_pol_index\n")
```

```
##
## Model: residuals$original_fair_chance_rate ~ residuals$original_neg_pol_index
```

```r
summary(lm_residuals_original_fair_neg)
```

```
##
## Call:
## lm(formula = original_fair_chance_rate ~ original_neg_pol_index,
##     data = residuals)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.732  -7.393  -1.221   8.590  25.260
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -4.355e-16  1.513e+00   0.000    1.000
## original_neg_pol_index -1.331e-02  1.494e-02  -0.891    0.377
##
## Residual standard error: 10.7 on 48 degrees of freedom
```

```
## Multiple R-squared:  0.01628,    Adjusted R-squared:  -0.004212
## F-statistic: 0.7945 on 1 and 48 DF,  p-value: 0.3772
```

```r
# Summarize results for residuals$validation_fair_chance_rate
cat("\nModel: residuals$validation_fair_chance_rate ~ residuals$original_pos_pol_index\n")
```

```
##
## Model: residuals$validation_fair_chance_rate ~ residuals$original_pos_pol_index
```

```r
summary(lm_residuals_validation_fair_pos)
```

```
##
## Call:
## lm(formula = validation_fair_chance_rate ~ original_pos_pol_index,
##     data = residuals)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.794  -8.097  -1.773   8.395  22.677
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -5.449e-16  1.422e+00   0.000   1.0000
## original_pos_pol_index -1.297e+01  7.439e+00  -1.744   0.0876 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.05 on 48 degrees of freedom
## Multiple R-squared:  0.05958,    Adjusted R-squared:  0.03999
## F-statistic: 3.041 on 1 and 48 DF,  p-value: 0.08759
```

```r
cat("\nModel: residuals$validation_fair_chance_rate ~ residuals$original_neg_pol_index\n")
```

```
##
## Model: residuals$validation_fair_chance_rate ~ residuals$original_neg_pol_index
```

```r
summary(lm_residuals_validation_fair_neg)
```

```
##
## Call:
## lm(formula = validation_fair_chance_rate ~ original_neg_pol_index,
##     data = residuals)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.061  -6.076  -2.571   7.771  20.588
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -3.585e-17  1.397e+00   0.000   1.0000
## original_neg_pol_index -3.036e-02  1.379e-02  -2.202   0.0325 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.878 on 48 degrees of freedom
## Multiple R-squared:  0.09176,    Adjusted R-squared:  0.07284
## F-statistic:  4.85 on 1 and 48 DF,  p-value: 0.03249
```

```r
# Calculate the median Positive Policy Index
median_pos_pol_index <- median(full_data$pos_pol_index, na.rm = TRUE)

# Find the state(s) closest to the median Positive Policy Index
closest_pos <- full_data %>%
  mutate(diff_from_median_pos = abs(pos_pol_index - median_pos_pol_index)) %>%
  filter(diff_from_median_pos == min(diff_from_median_pos, na.rm = TRUE)) %>%
  select(state, pos_pol_index)

# Calculate the median Negative Policy Index
median_neg_pol_index <- median(full_data$neg_pol_index, na.rm = TRUE)

# Find the state(s) closest to the median Negative Policy Index
closest_neg <- full_data %>%
  mutate(diff_from_median_neg = abs(neg_pol_index - median_neg_pol_index)) %>%
  filter(diff_from_median_neg == min(diff_from_median_neg, na.rm = TRUE)) %>%
  select(state, neg_pol_index)

# Print results
cat("Median Positive Policy Index:", median_pos_pol_index, "\n")
```

```
## Median Positive Policy Index: 0.3744444
```

```r
cat("State(s) closest to the median Positive Policy Index:\n")
```

```
## State(s) closest to the median Positive Policy Index:
```

```r
print(closest_pos)
```

```
##          state pos_pol_index
## 1       Hawaii     0.3825397
## 2 Rhode Island     0.3663492
```

```r
cat("\nMedian Negative Policy Index:", median_neg_pol_index, "\n")
```

```
##
## Median Negative Policy Index: 383.4417
```

```r
cat("State(s) closest to the median Negative Policy Index:\n")
```

```
## State(s) closest to the median Negative Policy Index:
```

```r
print(closest_neg)
```

```
##      state neg_pol_index
## 1 Virginia      394.2167
```

Scatter Plots: Fair Chance Job Posting Rate vs Policy Indices

We next look at how fair chance job posting rates relate to both positive and negative policy indices.

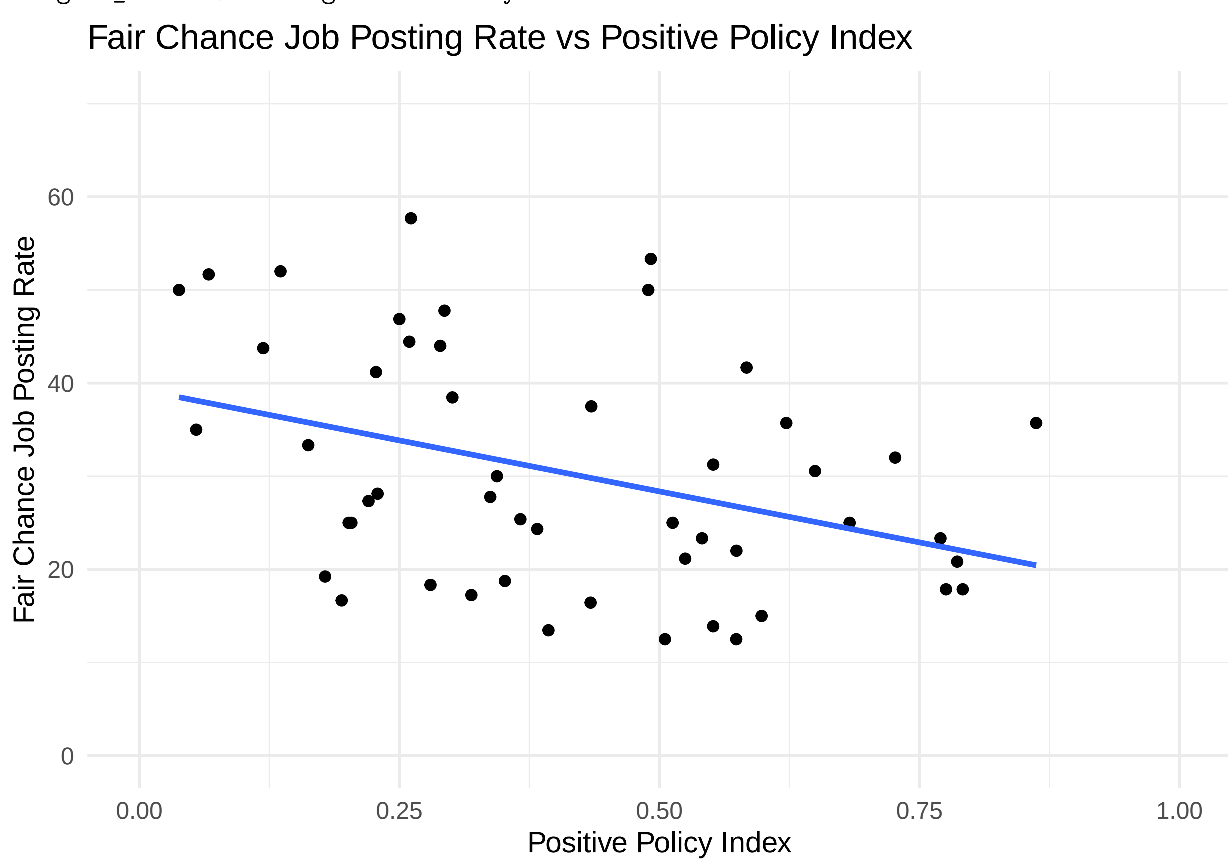
```r
# Create a scatter plot of positive policy index vs fair chance job posting rate
plot_pos_vs_fairchance_rate <- ggplot(full_data, aes(x = pos_pol_index, y = original_fair_chance_rate))
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(
```

```r
    title = "Fair Chance Job Posting Rate vs Positive Policy Index",
    x = "Positive Policy Index",
    y = "Fair Chance Job Posting Rate"
  ) +
  xlim(0, 1) +
  ylim(0, 70) +
  theme_minimal()

if (overwrite_plots) {
  # Save plot as PDF
  pdf("scatter_pos_fairchance.pdf")
  print(plot_pos_vs_fairchance_rate)
  dev.off()

  # Save plot as PNG
  ggsave("scatter_pos_fairchance.png", plot = plot_pos_vs_fairchance_rate, width = 8, height = 6)

}
```

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

```r
print(plot_pos_vs_fairchance_rate)
```

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



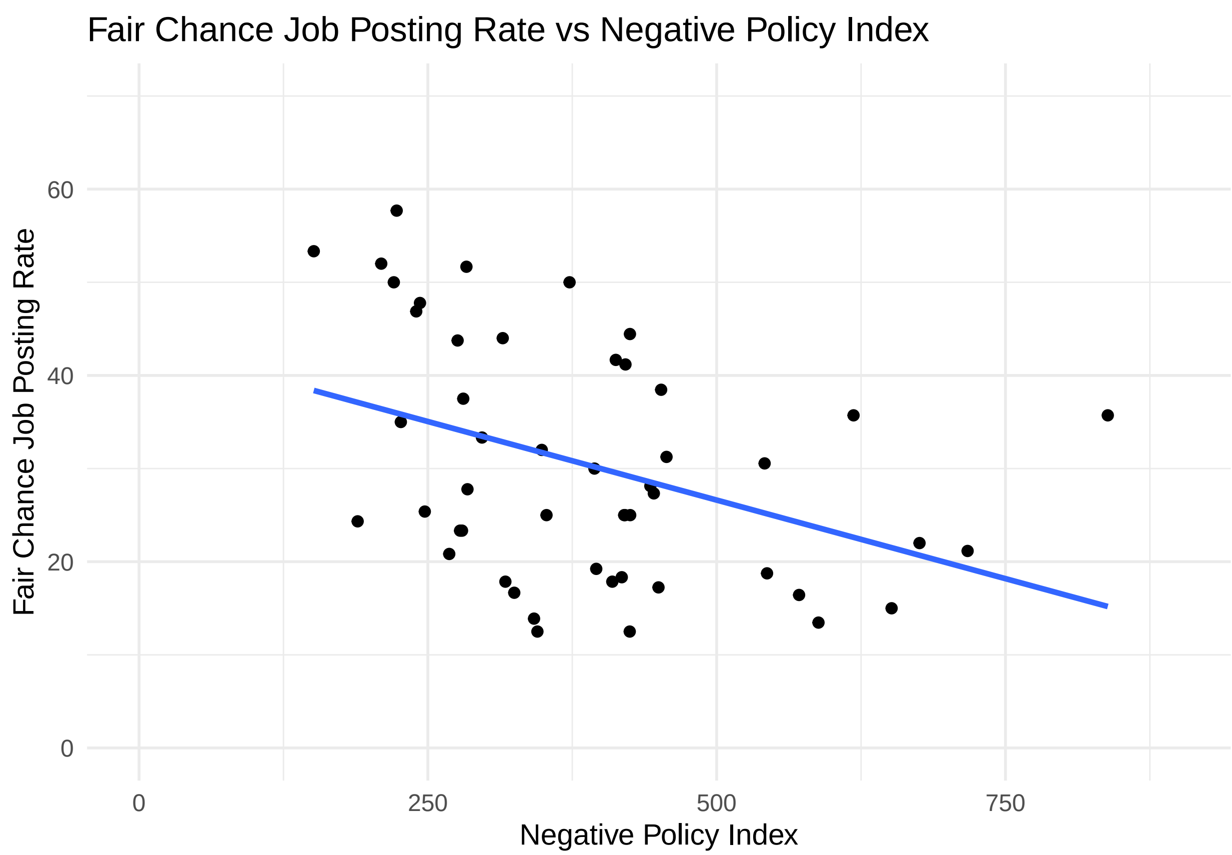

```r
# Create a scatter plot of negative policy index vs fair chance job posting rate
plot_neg_vs_fairchance_rate <- ggplot(full_data, aes(x = neg_pol_index, y = original_fair_chance_rate))
```

```r
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(
    title = "Fair Chance Job Posting Rate vs Negative Policy Index",
    x = "Negative Policy Index",
    y = "Fair Chance Job Posting Rate"
  ) +
  xlim(0, 900) +
  ylim(0, 70) +
  theme_minimal()

if (overwrite_plots) {
  # Save plot as PDF
  pdf("scatter_neg_fairchance.pdf")
  print(plot_neg_vs_fairchance_rate)
  dev.off()

  # Save plot as PNG
    ggsave("scatter_neg_fairchance.png", plot = plot_neg_vs_fairchance_rate, width = 8, height = 6)

}
```

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

```r
print(plot_neg_vs_fairchance_rate)
```

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




```r
# Linear model for original_fair_chance_rate
lm_original <- lm(
  original_fair_chance_rate ~ pos_pol_index + neg_pol_index +
    macroeconomy_control + socioeconomics_control +
    politics_control + diversity_control +
    original_employment_control,
  data = full_data
)

# Summary of the model for original_fair_chance_rate
cat("Model: original_fair_chance_rate\n")
```

```
## Model: original_fair_chance_rate
```

```r
summary(lm_original)
```

```
##
## Call:
## lm(formula = original_fair_chance_rate ~ pos_pol_index + neg_pol_index +
##     macroeconomy_control + socioeconomics_control + politics_control +
##     diversity_control + original_employment_control, data = full_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.870  -7.755  -1.823   7.822  22.256
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)                  49.074019  17.642579   2.782  0.00807 **
## pos_pol_index                -9.718043   8.517880  -1.141  0.26038
## neg_pol_index                -0.009568   0.016064  -0.596  0.55462
## macroeconomy_control         -0.574567   2.849435  -0.202  0.84117
## socioeconomics_control       -0.159502   2.813301  -0.057  0.95506
## politics_control             -1.346480   2.411237  -0.558  0.57952
## diversity_control           -17.590657   8.065079  -2.181  0.03483 *
## original_employment_control   0.312673   0.787169   0.397  0.69322
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.27 on 42 degrees of freedom
## Multiple R-squared:  0.3168, Adjusted R-squared:  0.2029
## F-statistic: 2.782 on 7 and 42 DF,  p-value: 0.01802
```

```r
# Linear model for validation_fair_chance_rate
lm_validation <- lm(
  validation_fair_chance_rate ~ pos_pol_index + neg_pol_index +
    macroeconomy_control + socioeconomics_control +
    politics_control + diversity_control +
    validation_employment_control,
  data = full_data
)

# Summary of the model for validation_fair_chance_rate
cat("\nModel: validation_fair_chance_rate\n")
```

```
##
## Model: validation_fair_chance_rate
```

```r
summary(lm_validation)
```

```
##
## Call:
## lm(formula = validation_fair_chance_rate ~ pos_pol_index + neg_pol_index +
##     macroeconomy_control + socioeconomics_control + politics_control +
##     diversity_control + validation_employment_control, data = full_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.815  -5.960  -2.231   7.314  20.398
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   137.69703   43.76875   3.146  0.00304 **
## pos_pol_index                 -11.25904    7.86508  -1.432  0.15968
## neg_pol_index                  -0.02724    0.01452  -1.876  0.06767 .
## macroeconomy_control           -5.84623    2.56380  -2.280  0.02773 *
## socioeconomics_control          2.95535    2.67791   1.104  0.27605
## politics_control                2.20763    2.15928   1.022  0.31245
## diversity_control             -20.06724    8.98931  -2.232  0.03098 *
## validation_employment_control  -1.25218    0.61925  -2.022  0.04957 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 10.21 on 42 degrees of freedom
## Multiple R-squared:  0.348,  Adjusted R-squared:  0.2393
## F-statistic: 3.203 on 7 and 42 DF,  p-value: 0.008257
```

Chi-Square Test of Independence

Finally, we perform chi-square tests to understand if having an above-average policy index is associated with above-average employment outcomes.

```r
# Create binary variables for "above or below average" for each measure (Positive Policy Index)
policy_above_avg_pos <- ifelse(residuals$original_pos_pol_index > 0, "Above Avg", "Below Avg")
employment_above_avg <- ifelse(residuals$original_fair_chance_rate > 0, "Above Avg", "Below Avg")

# Create a contingency table based on the two categorical variables (Positive Policy Index)
contingency_table_pos <- table(policy_above_avg_pos, employment_above_avg)

# Print the contingency table (Positive Policy Index)
print(contingency_table_pos)
```

```
##                     employment_above_avg
## policy_above_avg_pos Above Avg Below Avg
##            Above Avg         9        12
##            Below Avg        15        14
```

```r
# Run the chi-square test of independence (Positive Policy Index)
chi_squared_result_pos <- chisq.test(contingency_table_pos)

# Print the test result (Positive Policy Index)
print(chi_squared_result_pos)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  contingency_table_pos
## X-squared = 0.11065, df = 1, p-value = 0.7394
```

Interpretation: The chi-square test of independence for the positive policy index resulted in a p-value of 0.7394, which is much greater than the conventional significance level of 0.05. This means that we fail to reject the null hypothesis, suggesting that there is no statistically significant association between having a positive policy index above average and having a better-than-average employment outcome. In other words, there is no evidence to indicate that states with an above-average positive policy index are more likely to have above-average fair chance job posting rates.

```r
# Create binary variables for "above or below average" for each measure (Negative Policy Index)
policy_above_avg_neg <- ifelse(residuals$original_neg_pol_index > 0, "Above Avg", "Below Avg")
employment_above_avg <- ifelse(residuals$original_fair_chance_rate > 0, "Above Avg", "Below Avg")

# Create a contingency table based on the two categorical variables (Negative Policy Index)
contingency_table_neg <- table(policy_above_avg_neg, employment_above_avg)

# Print the contingency table (Negative Policy Index)
print(contingency_table_neg)
```

```
##                     employment_above_avg
## policy_above_avg_neg Above Avg Below Avg
```

```
##            Above Avg         8        14
##            Below Avg        16        12
```

```r
# Run the chi-square test of independence (Negative Policy Index)
chi_squared_result_neg <- chisq.test(contingency_table_neg)

# Print the test result (Negative Policy Index)
print(chi_squared_result_neg)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  contingency_table_neg
## X-squared = 1.38, df = 1, p-value = 0.2401
```

Interpretation: The chi-square test of independence for the negative combo index yielded a p-value of 0.2401, which is much greater than the conventional significance level of 0.05. This means that we fail to reject the null hypothesis, suggesting that there is no statistically significant association between having a negative policy index above average and having a better-than-average employment outcome. In other words, there is no evidence to indicate that states with an above-average negative policy index are more likely to have above-average fair chance job posting rates.

Conclusion

This report provides an exploratory analysis of the relationship between fair chance hiring policies and employment outcomes. We used visualizations, linear regression models, and chi-square tests to assess these relationships, which can provide valuable insights for policy evaluation and improvement.