Federal Reserve Bank of Minneapolis — Work Sample

Ben Rosenberg

2024-11-07

In this document, we'll be analyzing data from the Census Bureau's Current Population Survey (CPS). These files each describe a single month, so describing *all* months in 2023 required accessing and then consolidating 12 separate files. The consolidated table was then narrowed to only include the relevant variables and saved locally for faster processing.

```
# Loading Data

# Change to TRUE if calculations require other age ranges or variables
rebuild_data = F

if (rebuild_data) {
  cps_all23 = data.frame()

  for (i in 1:12) { #Combines each monthly dataset into one large table with a "month" value
    filename = paste0('CPS-Microdata/', month.abb[i], '23pub.csv')
    monthData = read.csv(filename) %>%
      rename_all(.funs=str_to_upper) %>% # cps variable names use inconsistent capitalization
      mutate(month=i)
    cps_all23 = cps_all23 %>% rbind(monthData) #loads in data
  }

  prime_age = cps_all23 %>% filter(PRTAGE > 24 & PRTAGE < 55) %>%
    select(PREMPNOT,   # labor status: 1. employed, 2. unemployed, 3&4. NILF
           PESEX,      # sex: 1. male, 2. female
           PRTAGE,     # age (max 85)
           GESTFIPS,   # FIPS codes for state-level analysis
           PTDTRACE,   # race. see race_coding for levels
           month)

  #Save filtered data to new file for faster knitting and processing
  write.csv(prime_age, "prime_age.csv")
}

prime_age = read.csv("prime_age.csv")
```

A few brief notes on my approach:

- `PREMPNOT`, the variable describing employment officially has four valid levels `1-4`. However, there is a fifth level, `-1`, seen in 3007 prime-age entries, or ~0.68% of the data being analyzed. A `-1` value is clearly defined for many other variables as "not in universe (NIU)" where no answer is applicable, but its interpretation is unclear to me for this variable and these analyses except as missing data. With that in mind, those entries are excluded from the population counts to which these statistics compare.

- Though I have used fairly descriptive titles and axis labels for these plots, I would generally offer more accompanying text to contextualize the statistic shown, depending on the audience. In this case, I am erring on the side of brevity.

- There are a few quirks to the syntax/style I use in R, namely using `=` instead of `<-` for variable assignment. Having worked in multiple programming languages and dealt with porting code from one to another, I prefer a more general syntax. That said, in the context of shared projects or work within an organization, I would be more than happy to adopt other style standards.

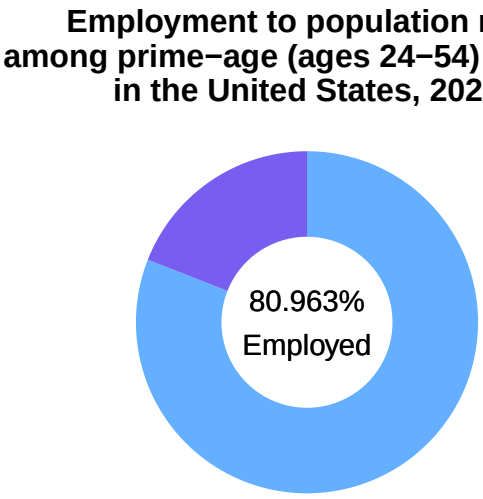
```
## CUSTOM FUNCTIONS & VARIABLES

factorize = function(tbl, groupby="month") { #Converts data into summable booleans
  return(
    tbl %>%
      mutate(inLF = PREMPNOT > 0 & PREMPNOT < 3,
           outLF = PREMPNOT > 2,
           employed = PREMPNOT == 1,
           unemployed = PREMPNOT == 2,
           other = PREMPNOT == -1,
           all = 1,
           complete = all-other) %>%
    group_by(tbl[groupby]) %>%
    summarize_at(.vars=vars(inLF, outLF, employed, unemployed, other, all, complete), .funs=sum)
  )
}

donutAB = function(rate, A, B, metric) { #ggplot2 formulation for a simple donut chart
  out = data.frame(
    r = c(rate, 1),
    category = c(A,B)
  ) %>% mutate(ymn = c(0, rate),
               ymx = c(rate, 1),
               labelY = (ymn+ymx)/2
               ) %>%
    ggplot(aes(xmin=3, xmax=4, ymin=ymn, ymax=ymx, fill=category)) +
    geom_rect() +
    geom_text(x=2, y=0, label=paste0(round(rate*100, 3), "%\n", metric), size=4) +
    coord_polar(theta="y") +
    xlim(2, 4) +
    theme_void() +
    scale_fill_manual(values=my_palette) +
    guides(fill="none") +
    theme(plot.title = element_text(hjust=0.5, size=12, face="bold"),
          plot.caption = element_text(hjust=1, face="italic", size=8))
  return(out)
}

propDiffSig = function(p1, n1, p2, n2, rSE=F) { #I'm sure there's a nifty library function for this
  p.hat = (p1*n1 + p2*n2)/(n1+n2)
  SE = sqrt(p.hat*(1-p.hat)*(1/n1 + 1/n2))
  z = (p1-p2)/SE
```

```
  if (rSE) {return(SE)}
  return(pnorm(abs(z), lower.tail=F))
}

pctStyle = function(n, dec=2, suf="%") {
  return(paste0(round(n*100, dec), suf))
}

my_theme = theme_bw() +
   theme(plot.title = element_text(hjust=0.5, size=12, face="bold"),
         plot.caption = element_text(hjust=0, face="italic", size=8))


monthscale = scale_x_continuous(breaks=c(1:12), labels=month.abb)

my_palette= c("#64afff","#785ef0","#dc267f", "#fe6100","#ffb000")

All = factorize(prime_age)
Male = factorize(prime_age %>% filter(PESEX==1))
Female = factorize(prime_age %>% filter(PESEX==2))

race_coding = c('White Only',
                'Black Only',
                'American Indian, Alaskan Native Only',
                'Asian Only',
                'Hawaiian/Pacific Islander Only',
                'White-Black',
                'White-AI',
                'White-Asian',
                'White-HP',
                'Black-AI',
                'Black-Asian',
                'Black-HP',
                'AI-Asian',
                'AI-HP',
                'Asian-HP',
                'W-B-AI',
                'W-B-A',
                'W-B-HP',
                'W-AI-A',
                'W-AI-HP',
                'W-A-HP',
                'B-AI-A',
                'W-B-AI-A',
                'W-AI-A-HP',
                'Other 3 Race Combinations',
                'Other 4 and 5 Race Combinations'
                )
```

Employment to population ratio

This is a fairly straightforward statistic, and knowing that the key information we are communicating is the proportion of a general population which is employed (and not in comparison to any other proportion), I think a very simple pie or donut chart is appropriate.

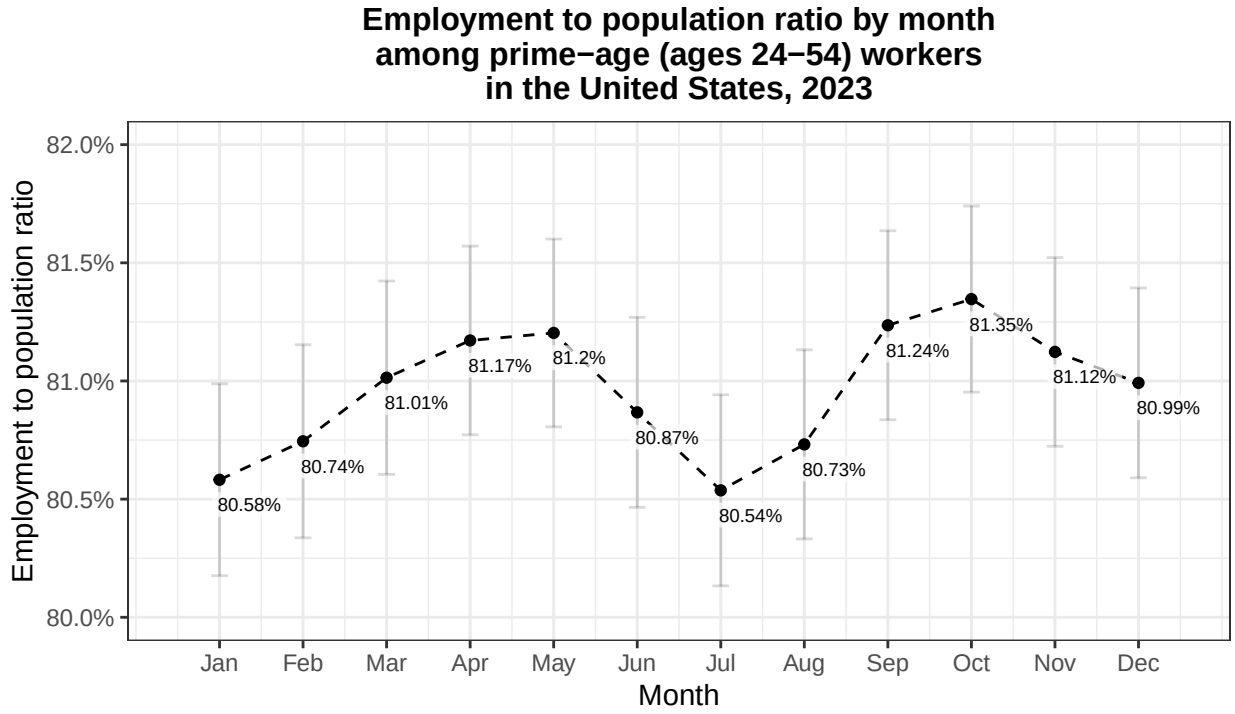
```
e2p = data.frame(month = All$month, #Employment to population by month including 95%CI error
                 E2P = c(All$employed / All$complete)
                 ) %>%
  mutate(SE = sqrt(E2P * (1-E2P) / All$complete),
         err = SE*1.96)

e2p_all = sum(All$employed) / sum(All$complete)

donutAB(e2p_all, "Employed", "Other", "Employed") +
  labs(title="Employment to population ratio\namong prime-age (ages 24-54) workers\nin the United States
       caption="Data Source: Census Bureau - Current Population Survey")
```



Though the prompt called for a single statistic characterizing *all months*, I also calculated the statistic for *each* month in 2023 and have included a visualization of those data below including error bars at a 95% confidence interval.

```
e2p %>%
  ggplot(aes(x=month, y=E2P,
             label=pctStyle(E2P),
             ymin = E2P-err, ymax = E2P+err)) +
  geom_point() +
  geom_line(linetype=2) +
  geom_errorbar(alpha=0.15, width=0.2) +
  geom_label(hjust=0.1, vjust=1.5, size=2.5, alpha=0.7, position=position_dodge(width=1), label.size=NA
  labs(title = "Employment to population ratio by month\namong prime-age (ages 24-54) workers\nin the U
       x="Month", y="Employment to population ratio",
       caption="Data Source: US Census Bureau - Current Population Survey") +
  scale_y_continuous(labels=scales::percent, limits=c(0.8, 0.82)) +
  monthscale +
```

```
my_theme
```



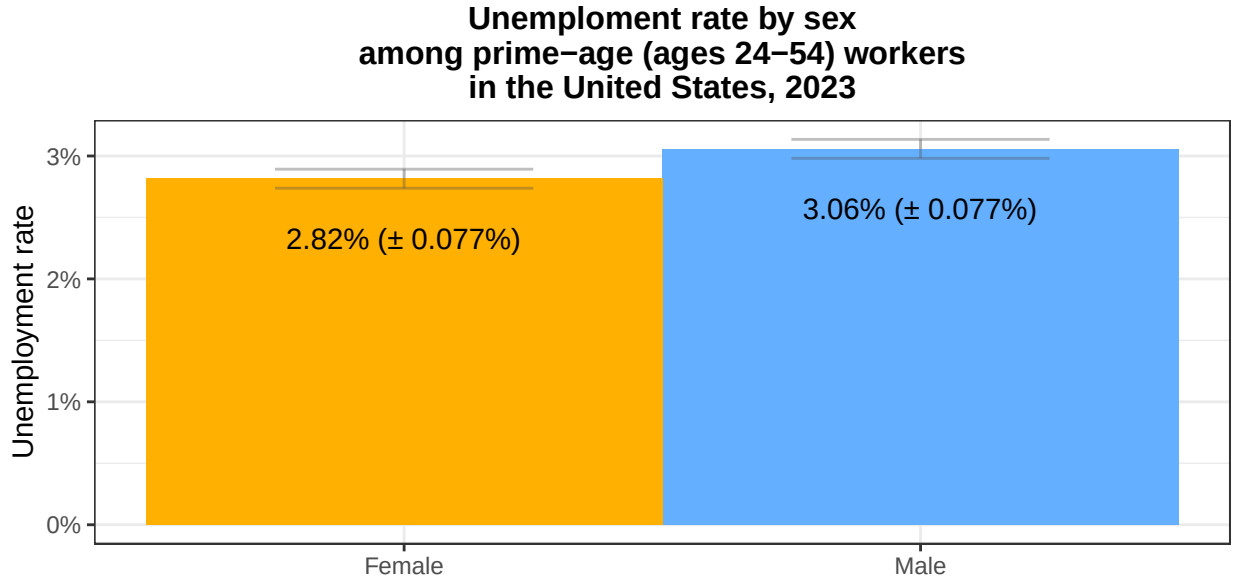
Data Source: US Census Bureau − Current Population Survey

Unemployment rate by sex

Since this statistic asks the reader to compare two proportions, I have opted for a simple bar graph with error bars corresponding with a 95% confidence interval.

```
unemp = data.frame(
  month = c(All$month, All$month),
  sex = c(rep("Male", 12), rep("Female", 12)),
  pop = c(Male$inLF, Female$inLF),
  rate = c(Male$unemployed / Male$inLF, Female$unemployed / Female$inLF)
) %>% mutate(
    SE = sqrt(rate*(1-rate)/pop),
    err = 1.96*SE)

m_unemp = sum(Male$unemployed) / sum(Male$inLF)
f_unemp = sum(Female$unemployed) / sum(Female$inLF)

male_SE = sqrt(m_unemp*(1-m_unemp)/sum(Male$inLF))
female_SE = sqrt(f_unemp*(1-f_unemp)/sum(Female$inLF))


ur.display = data.frame(
  unemployment = c(m_unemp, f_unemp),
  sex = c("Male", "Female"),
  low = c(m_unemp-1.96*male_SE, f_unemp-1.96*female_SE),
  high = c(m_unemp+1.96*male_SE, f_unemp+1.96*female_SE),
```

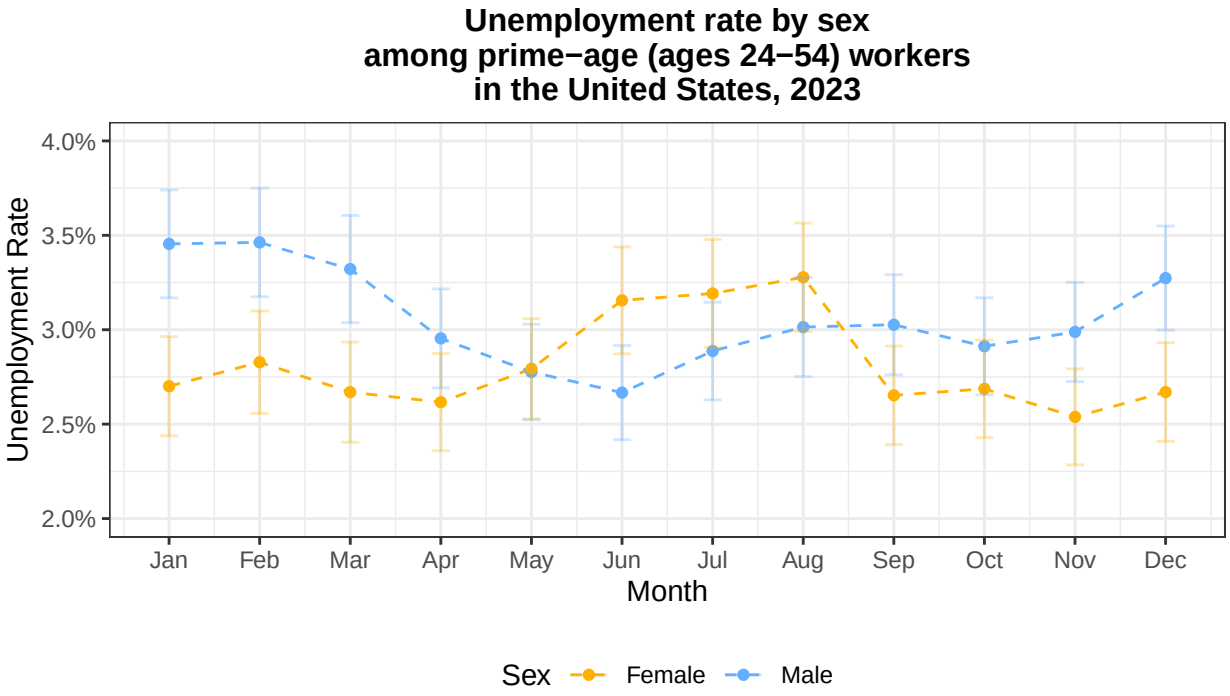
```
  err = 1.96*c(male_SE, female_SE)
)

ur.display %>% ggplot(aes(x=sex, y=unemployment, fill=sex, label=paste0(pctStyle(unemployment), " (\u00
  geom_bar(stat="identity", width=1) +
  geom_text(aes(y=(unemployment-0.004), vjust=1)) +
  geom_errorbar(aes(ymin=low, ymax=high), color="#5555555f", width=0.5) +
  scale_y_continuous("Unemployment rate", labels=scales::percent) +
  labs(title="Unemploment rate by sex\namong prime-age (ages 24-54) workers\nin the United States, 2023
       caption="Data Source: US Census Bureau - Current Population Survey microdata") +
  guides(fill="none") +
  scale_fill_manual(values=c(my_palette[5], my_palette[1])) +
  my_theme
```



Data Source: US Census Bureau − Current Population Survey microdata

And, again, I've opted to include the monthly view.

```
unemp %>%
  ggplot(aes(x=month, y=rate, col=sex, ymin=rate-err, ymax=rate+err)) +
  geom_line(linetype=2) +
  geom_point() +
  geom_errorbar(alpha=0.3, width=0.2) +
  scale_y_continuous(labels=scales::percent, limits=c(0.02,0.04)) +
  scale_color_manual(values=c(my_palette[5], my_palette[1])) +
  monthscale +
  labs(x="Month",
       y="Unemployment Rate",
       color="Sex",
       title="Unemployment rate by sex\namong prime-age (ages 24-54) workers\nin the United States, 202
       caption="Data Source: US Census Bureau - Current Population Survey") +
  my_theme +
  theme(legend.position="bottom")
```



Data Source: US Census Bureau − Current Population Survey

Although these visualizations and their prompt consider the rates for these groups independently, it feels important to convey whether the difference between these two subgroups is statistically significant. That may be beyond our scope here, but the overlapping error bars in several months encouraged a bit of extra digging. Some difference-of-proportion p-value calculations suggest that the whole-year difference (and several monthly differences) is not significant at a $p \leq 0.05$ level.

```
unemp_sig23 = propDiffSig(m_unemp,
            sum(Male$inLF),
            f_unemp,
            sum(f_unemp))

unemp_sig = unemp %>% group_by(month) %>%
  summarize(p = propDiffSig(rate[1], pop[1], rate[2], pop[2])) %>%
  mutate(Month = month.name[month]) %>% select(Month, p)
```

p for the difference in male and female unemployment rates in *all months* in 2023: 0.4990559

p for the difference in male and female unemployment rates in *each month* in 2023:

```
kbl(unemp_sig, booktabs=T, linesep="", align="c") %>% kable_styling(latex_options="hold_position") %>%
  column_spec(2, background = spec_color(log(unemp_sig$p), begin=0.4, end=0.7)) %>%
  column_spec(1, background = spec_color(unemp_sig$p > 0.05, palette=c("#fefefe","#ffffcf")))
```

Month	p
January	0.0000748
February	0.0008665
March	0.0005274
April	0.0362689
May	0.4639699
June	0.0054303
July	0.0602747
August	0.0913504
September	0.0247729
October	0.1131861
November	0.0081749
December	0.0009652

Labor force participation rate

Lastly, we are describing the proportion of the full population which participates in the labor force, grouped by race. In considering a proportion across a large number of categories, I think a bar graph does best when it is sorted by the proportion for easy navigation and comparison. That said, we also need to consider how comparable these statistics are.

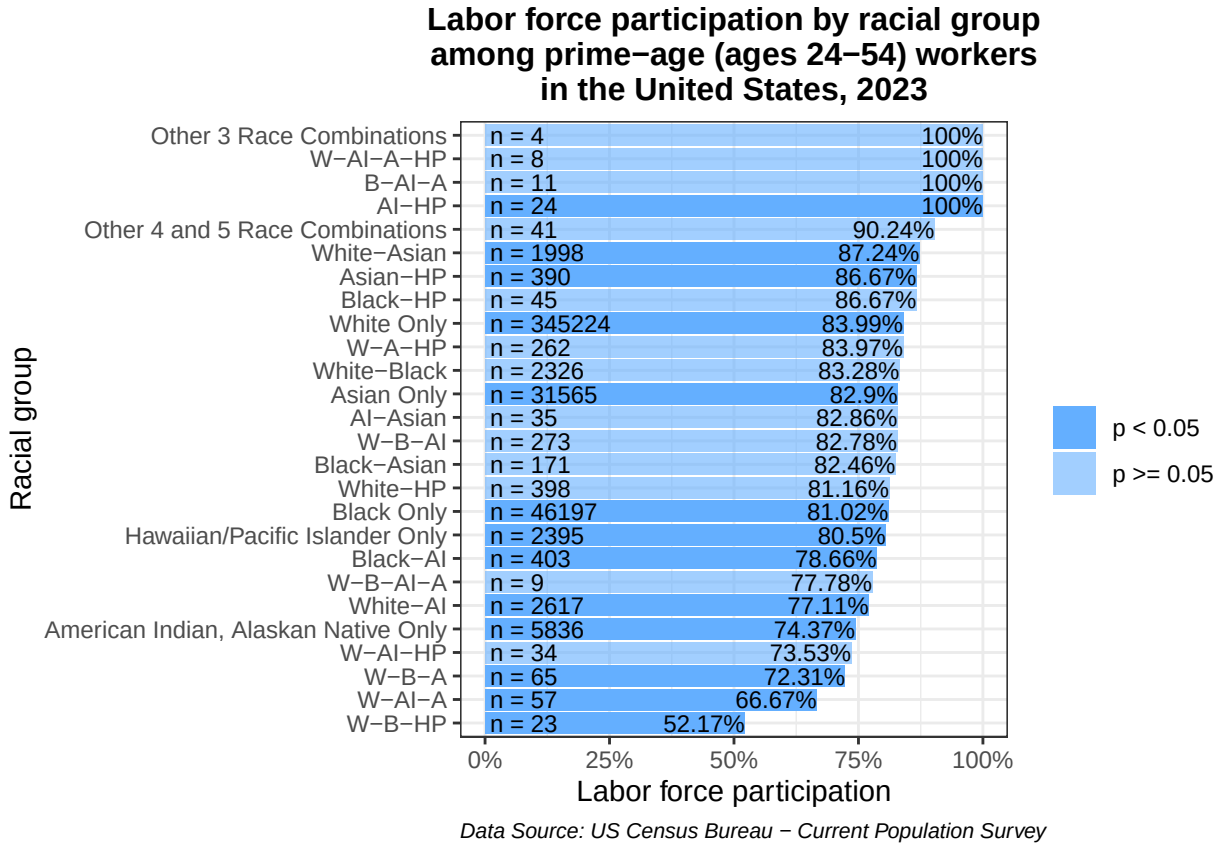
The race variable, `PTDTRACE`, creates an interesting challenge, as the 26 levels of the variable include some groups with as few as 4 members surveyed throughout 2023. This could easily produce issues of class imbalance wherein smaller racial groups are much more strongly impacted by a bias in sampling than larger groups. Even when communicating this to an *informed* general audience, it's important to convey this issue of statistical significance thoughtfully so that smaller groups can be both included and represented accurately.

Ideally, this sort of issue would be managed on the data collection end. If we were developing some sort of predictive model using these data, we would do well to account for that class imbalance through resampling. In this context, I think it's best to just convey the data limitation within the visualization.

```
lfp = prime_age %>% # labor force participation by race by month
  factorize(c('PTDTRACE', 'month')) %>%
  mutate(lfp = inLF/complete) %>%
  select(PTDTRACE, month, inLF, complete, lfp)

lfp23 = lfp %>% # labor force participation by race
  group_by(PTDTRACE) %>%
  summarize(inLF = sum(inLF), complete = sum(complete)) %>%
  mutate(LFP = inLF/complete) %>%
  arrange(LFP) %>%
  mutate(
    p.val = propDiffSig(sum(inLF)/sum(complete),
                        sum(complete),
                        LFP,
                        complete),
    race = race_coding[PTDTRACE])
```

```
lfp23 %>% mutate(X=row_number(),
                 PTDTRACE = factor(PTDTRACE, levels=PTDTRACE)) %>%
  ggplot(aes(x=PTDTRACE, y=LFP, label=pctStyle(LFP), fill=ifelse(p.val<0.05, "p < 0.05", "p >= 0.05")))
  geom_bar(stat="identity") +
  geom_text(hjust=1, size=3) +
```

```
  geom_text(aes(y=0.01, label=paste0("n = ", lfp23$complete)), hjust=0, size=3) +
  scale_y_continuous(labels=scales::percent) +
  scale_x_discrete(breaks=lfp23$PTDTRACE, labels=lfp23$race) +
  scale_fill_manual(values=c(my_palette[1], "#64afff99")) +
  labs(fill="",
       x="Racial group",
       y="Labor force participation",
       title="Labor force participation by racial group\namong prime-age (ages 24-54) workers\nin the Un
       caption="Data Source: US Census Bureau - Current Population Survey") +
  my_theme +
  coord_flip()
```



For statistical significance, I am considering whether there is sufficient evidence to reject H_0 where the proportion (labor force participation rate) in the group is *equal* to the proportion in the whole population (83.42%). With that in mind, larger samples are necessary for groups with a labor force participation rate which is closer in value to the population rate.