ProblemSet2_new

Ke-li Chiu & Diego Mamanche Castellanos

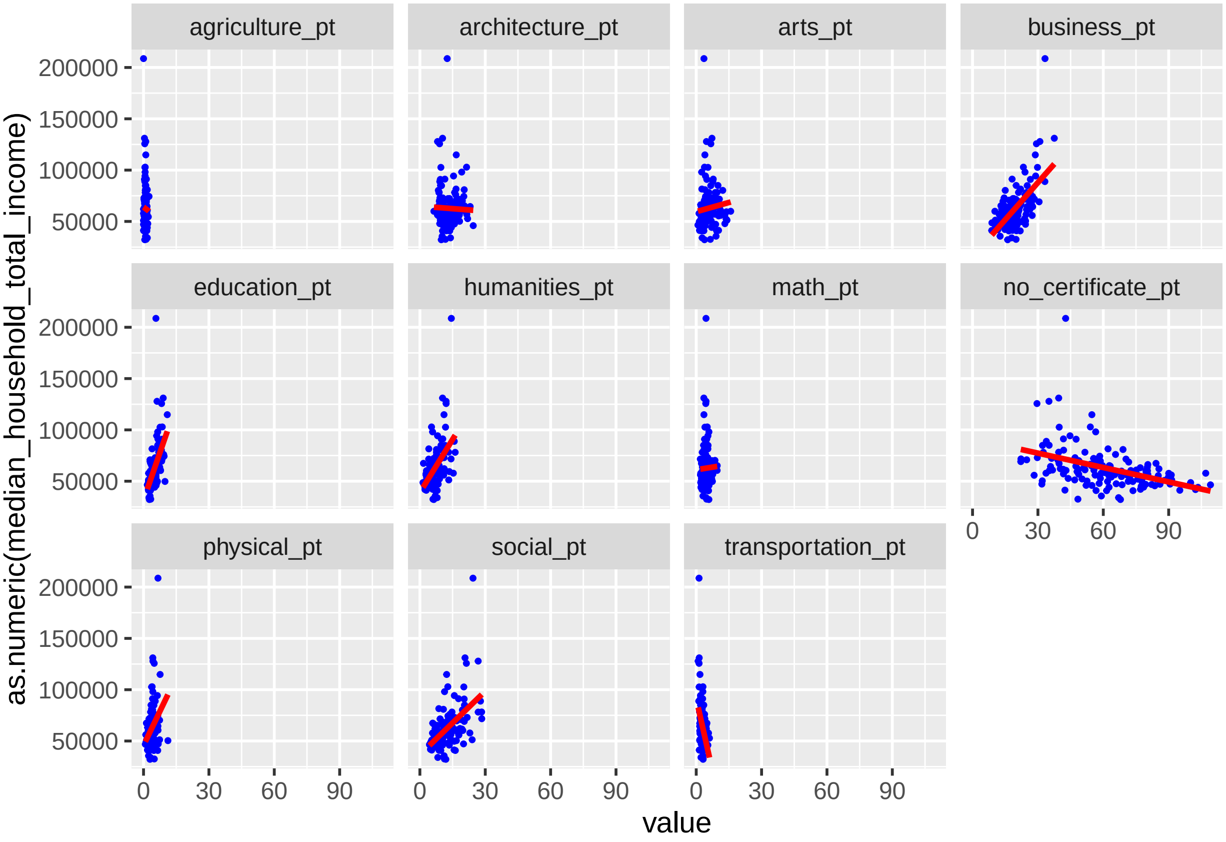
06/02/2020

Abstract

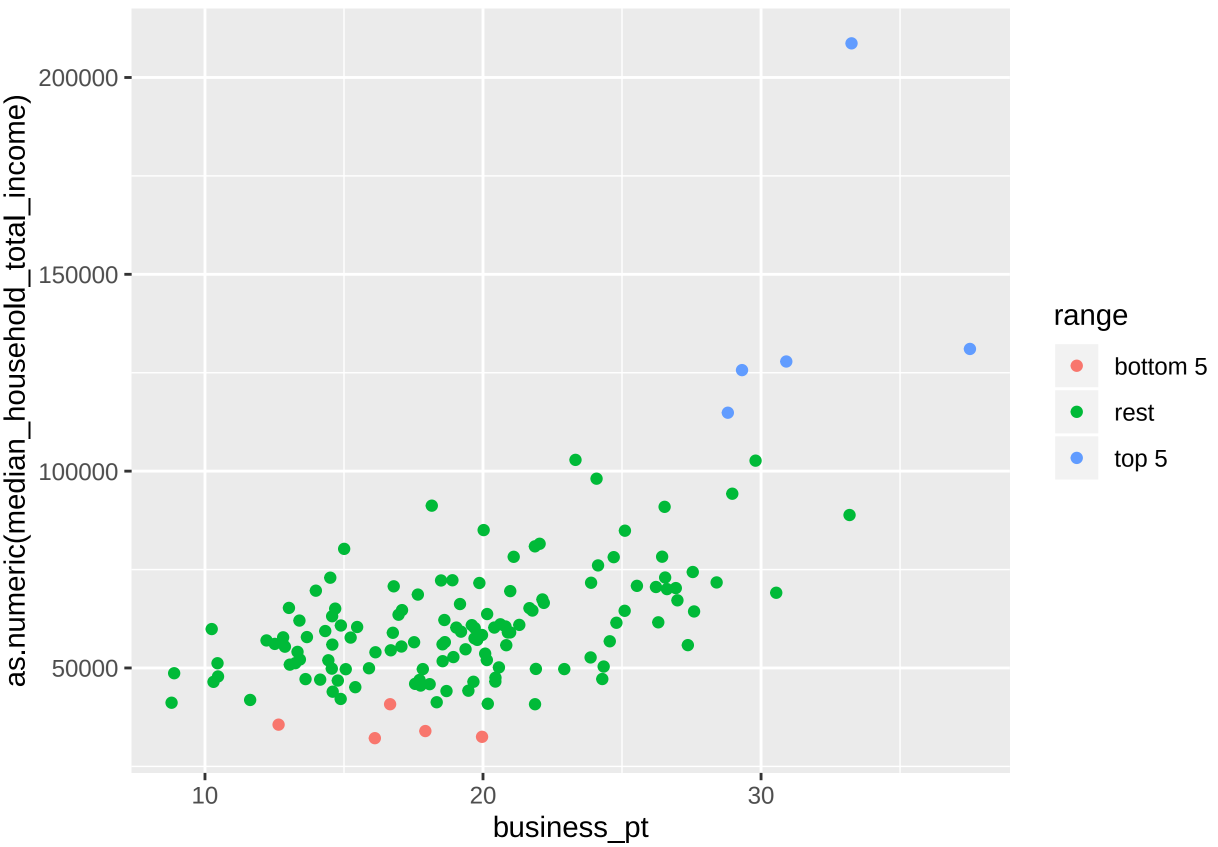
Abstract nnnnn nnnnnn

```r
# Get the resource we want from this package
neighbourhood_raw <- list_package_resources("6e19a90f-971c-46b3-852c-0c48c436d1fc") %>%
  filter(name == "Neighbourhood Data 2001, 2006, 2011.xlsx") %>%
  get_resource()
main_df_raw <- neighbourhood_raw["2011"]
main_df_raw <- as.data.frame(main_df_raw)
# Remove "X2011 prefix from all column names"
names(main_df_raw) <- substring(names(main_df_raw), 7)
```

```r
# Explore relationship between income and every major perecntage
education_percentage_only %>%
  gather(-c(median_household_total_income, total_population), key = "var", value = "value") %>%
  ggplot(aes(x = value, y = as.numeric(median_household_total_income))) +
    facet_wrap(~ var, scales = "fixed") +
    geom_point(shape=20, color="blue", size=1) +
    stat_smooth(method=lm, se=FALSE,  colour="red")
```



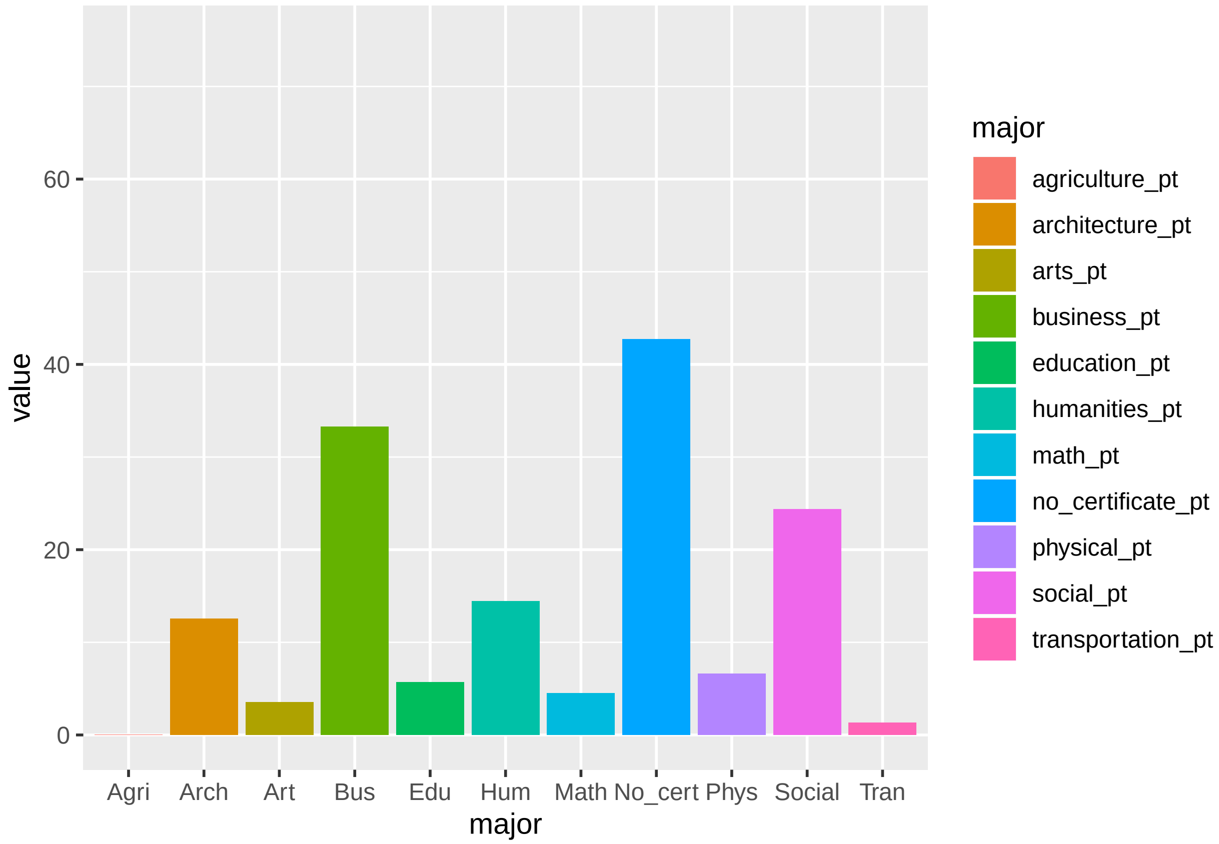
```
# Business major seem to have a stiff line, lets zoom in to it
# Get highest 5 and lowest 5 income values
income <- education_percentage_only$median_household_total_income
h5th <- sort(income)[length(income)-4]
l5th <- sort(income)[5]
# Assign income labels to the neighbourhoods
education_percentage_only$range<-ifelse(income >= h5th,"top 5",
        ifelse(income <= l5th,"bottom 5","rest"
        ))
# Plot it
ggplot(education_percentage_only, aes(x=business_pt, y=as.numeric(median_household_total_income), color
```



```r
# Linear regression model
linearMod <- lm(business_pt ~ as.numeric(median_household_total_income), data=education_percentage_only)
summary(linearMod)
```

```
##
## Call:
## lm(formula = business_pt ~ as.numeric(median_household_total_income),
##     data = education_percentage_only)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.1594 -2.9898  0.4262  2.6270  9.9781
##
## Coefficients:
##                                             Estimate Std. Error t value
## (Intercept)                                9.752e+00  1.129e+00   8.640
## as.numeric(median_household_total_income) 1.565e-04  1.705e-05   9.183
##                                            Pr(>|t|)
## (Intercept)                                1.20e-14 ***
## as.numeric(median_household_total_income) 5.31e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.31 on 139 degrees of freedom
## Multiple R-squared:  0.3776, Adjusted R-squared:  0.3731
## F-statistic: 84.32 on 1 and 139 DF,  p-value: 5.309e-16
```

```
# Plot major percentage distribution in highest income neighbourhood
education_df_highest <- education_percentage_only %>%
  filter(median_household_total_income == max(as.numeric(median_household_total_income)) )
data_plot_highest <-
  education_df_highest %>%
  pivot_longer(cols = "education_pt":"no_certificate_pt", names_to = "major")
# Make a bar chart
data_plot_highest %>%
  ggplot(aes(x = major, y = value, fill = major)) +
  geom_col()+
  ylim(0,75)+
  scale_x_discrete(
        labels=c("Agri", "Arch", "Art", "Bus","Edu", "Hum","Math", "No_cert", "Phys","Social", "Tran"))
```

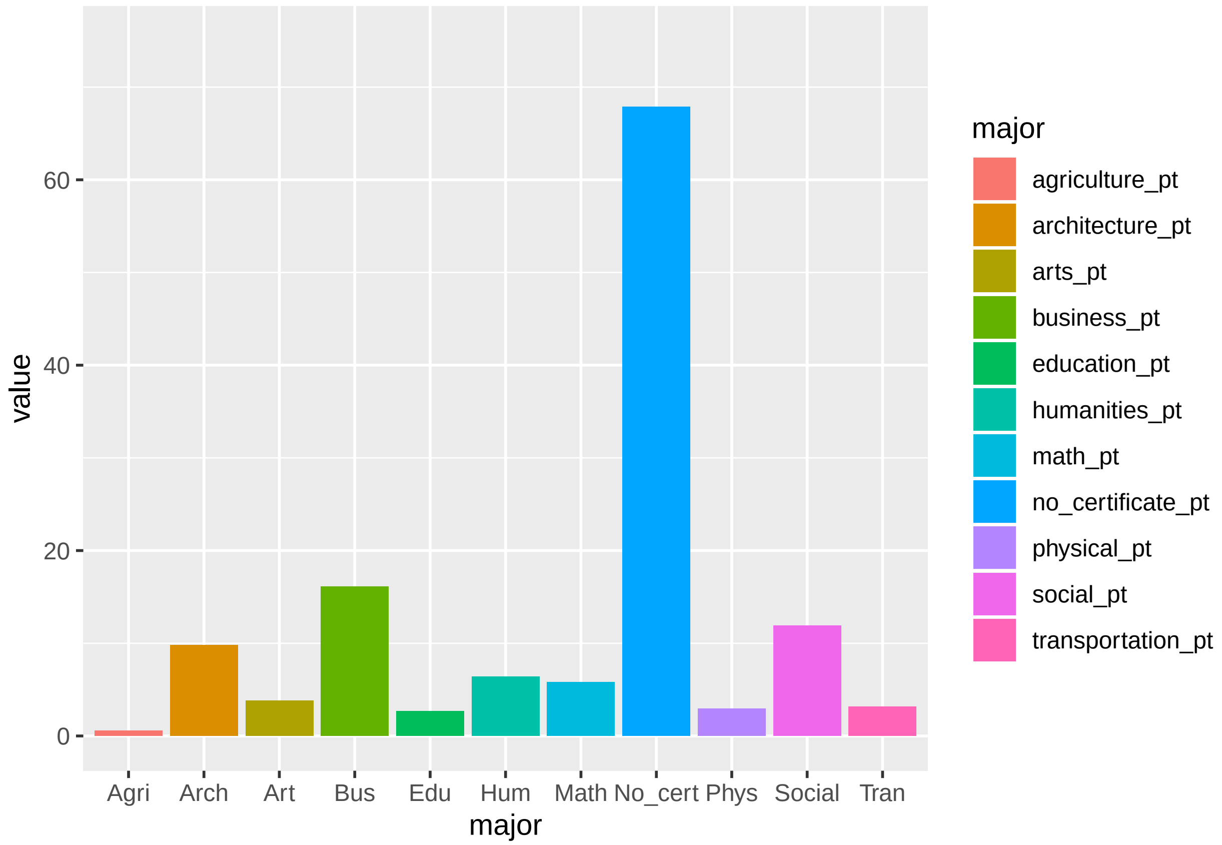


```
  #theme(axis.text.x = element_blank())
```

```
# Plot major percentage distribution in lowest income neighbourhood
education_df_lowest <- education_percentage_only %>%
   filter(median_household_total_income == min(as.numeric(median_household_total_income)) )
education_df_lowest
```

```
##   total_population median_household_total_income education_pt  arts_pt
## 1             5245                         32172     2.669209 3.813155
##   humanities_pt social_pt business_pt physical_pt  math_pt architecture_pt
## 1      6.387035  11.91611    16.11058    2.955195 5.815062        9.818875
##   agriculture_pt transportation_pt no_certificate_pt    range
## 1      0.5719733          3.145853          67.87417 bottom 5
```

```
data_plot_lowest <-
  education_df_lowest %>%
  pivot_longer(cols = "education_pt":"no_certificate_pt", names_to = "major")
# Make a bar chart
data_plot_lowest %>%
  ggplot(aes(x = major, y = value, fill = major)) +
  geom_col()+
  ylim(0,75)+
  scale_x_discrete(
        labels=c("Agri", "Arch", "Art", "Bus","Edu", "Hum","Math", "No_cert", "Phys","Social", "Tran"))
```



```
  #theme(axis.text.x = element_blank())
```