

# ONLY 500 Final Project

## Data Cleaning

- Unzip MERGED2014\_15\_PP.csv.zip, it is from College Scorecard webiste, not Kaggle. Since Kaggle do not have latest data.
- `mrc_table10` is data from Mobility Report Card data, they give each school a tier. Looks more useful than payscale school type. For detail, please see Codebook-MRC-Table-6.pdf.

## Read Data

```
payscale_college_type = read.csv('salaries-by-college-type.csv')
payscale_college_type = payscale_college_type[, c(1,2)]
payscale_region = read.csv('salaries-by-region.csv')
payscale_region = payscale_region[, c(1,2)]

mrc_table10 = read.csv('mrc_table10.csv')
mrc_table10 = mrc_table10[, c(1, 12)]

# Treat "NULL" and "PrivacySuppressed" as NA when read
college_scorecard = read.csv('MERGED2014_15_PP.csv', na=c("NULL", "PrivacySuppressed"))
college_scorecard = college_scorecard[, c(3, 4,6,17,377,1639,1640,1642,1643,1645,1655,1656,1657,1661,1662,1663,1664,1665,1666,1667,1668,1669,1670,1671,1672,1673,1674,1675,1676,1677,1678,1679,1680,1681,1682,1683,1684,1685,1686,1687,1688,1689,1690,1691,1692,1693,1694,1695,1696,1697,1698,1699,1700,1701,1702,1703,1704,1705,1706,1707,1708,1709,1710,1711,1712,1713,1714,1715,1716,1717,1718,1719,1720,1721,1722,1723,1724,1725,1726,1727,1728,1729,1730,1731,1732,1733,1734,1735,1736,1737,1738,1739,1740,1741,1742,1743,1744,1745,1746,1747,1748,1749,1750,1751,1752,1753,1754,1755,1756,1757,1758,1759,1760,1761,1762,1763,1764,1765,1766,1767,1768,1769,1770,1771,1772,1773,1774,1775,1776,1777,1778,1779,1780,1781,1782,1783,1784,1785,1786,1787,1788,1789,1790,1791,1792,1793,1794,1795,1796,1797,1798,1799,1800,1801,1802,1803,1804,1805,1806,1807,1808,1809,1810,1811,1812,1813,1814,1815,1816,1817,1818,1819,1820,1821,1822,1823,1824,1825,1826,1827,1828,1829,1830,1831,1832,1833,1834,1835,1836,1837,1838,1839,1840,1841,1842,1843,1844,1845,1846,1847,1848,1849,1850,1851,1852,1853,1854,1855,1856,1857,1858,1859,1860,1861,1862,1863,1864,1865,1866,1867,1868,1869,1870,1871,1872,1873,1874,1875,1876,1877,1878,1879,1880,1881,1882,1883,1884,1885,1886,1887,1888,1889,1890,1891,1892,1893,1894,1895,1896,1897,1898,1899,1900,1901,1902,1903,1904,1905,1906,1907,1908,1909,1910,1911,1912,1913,1914,1915,1916,1917,1918,1919,1920,1921,1922,1923,1924,1925,1926,1927,1928,1929,1930,1931,1932,1933,1934,1935,1936,1937,1938,1939,1940,1941,1942,1943,1944,1945,1946,1947,1948,1949,1950,1951,1952,1953,1954,1955,1956,1957,1958,1959,1960,1961,1962,1963,1964,1965,1966,1967,1968,1969,1970,1971,1972,1973,1974,1975,1976,1977,1978,1979,1980,1981,1982,1983,1984,1985,1986,1987,1988,1989,1990,1991,1992,1993,1994,1995,1996,1997,1998,1999,2000,2001,2002,2003,2004,2005,2006,2007,2008,2009,2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020,2021,2022,2023,2024,2025,2026,2027,2028,2029,2030,2031,2032,2033,2034,2035,2036,2037,2038,2039,2040,2041,2042,2043,2044,2045,2046,2047,2048,2049,2050,2051,2052,2053,2054,2055,2056,2057,2058,2059,2060,2061,2062,2063,2064,2065,2066,2067,2068,2069,2070,2071,2072,2073,2074,2075,2076,2077,2078,2079,2080,2081,2082,2083,2084,2085,2086,2087,2088,2089,2090,2091,2092,2093,2094,2095,2096,2097,2098,2099,2100,2101,2102,2103,2104,2105,2106,2107,2108,2109,2110,2111,2112,2113,2114,2115,2116,2117,2118,2119,2120,2121,2122,2123,2124,2125,2126,2127,2128,2129,2130,2131,2132,2133,2134,2135,2136,2137,2138,2139,2140,2141,2142,2143,2144,2145,2146,2147,2148,2149,2150,2151,2152,2153,2154,2155,2156,2157,2158,2159,2160,2161,2162,2163,2164,2165,2166,2167,2168,2169,2170,2171,2172,2173,2174,2175,2176,2177,2178,2179,2180,2181,2182,2183,2184,2185,2186,2187,2188,2189,2190,2191,2192,2193,2194,2195,2196,2197,2198,2199,2200,2201,2202,2203,2204,2205,2206,2207,2208,2209,2210,2211,2212,2213,2214,2215,2216,2217,2218,2219,2220,2221,2222,2223,2224,2225,2226,2227,2228,2229,2230,2231,2232,2233,2234,2235,2236,2237,2238,2239,2240,2241,2242,2243,2244,2245,2246,2247,2248,2249,2250,2251,2252,2253,2254,2255,2256,2257,2258,2259,2260,2261,2262,2263,2264,2265,2266,2267,2268,2269,2270,2271,2272,2273,2274,2275,2276,2277,2278,2279,2280,2281,2282,2283,2284,2285,2286,2287,2288,2289,2290,2291,2292,2293,2294,2295,2296,2297,2298,2299,2300,2301,2302,2303,2304,2305,2306,2307,2308,2309,2310,2311,2312,2313,2314,2315,2316,2317,2318,2319,2320,2321,2322,2323,2324,2325,2326,2327,2328,2329,2330,2331,2332,2333,2334,2335,2336,2337,2338,2339,2340,2341,2342,2343,2344,2345,2346,2347,2348,2349,2350,2351,2352,2353,2354,2355,2356,2357,2358,2359,2360,2361,2362,2363,2364,2365,2366,2367,2368,2369,2370,2371,2372,2373,2374,2375,2376,2377,2378,2379,2380,2381,2382,2383,2384,2385,2386,2387,2388,2389,2390,2391,2392,2393,2394,2395,2396,2397,2398,2399,2400,2401,2402,2403,2404,2405,2406,2407,2408,2409,2410,2411,2412,2413,2414,2415,2416,2417,2418,2419,2420,2421,2422,2423,2424,2425,2426,2427,2428,2429,2430,2431,2432,2433,2434,2435,2436,2
```

## Merge Data

## Process School Name

Normalize all school name for merging.

```
process_school_name = function(data) {
  data = sub(" \\(\\..*\\)", "", data)
  data = sub(" - ", "-", data)
  data = sub(", ", "-", data)
  data = sub("\\\\.", "", data)
  data = sub(" & ", " and ", data)
  data = sub("&", " and ", data)
  data = sub("St ", "Saint ", data)
  data
}

payscale_college_type$School.Name = process_school_name(payscale_college_type$School.Name)
nrow(payscale_college_type)

## [1] 269

payscale_region$School.Name = process_school_name(payscale_region$School.Name)
nrow(payscale_region)

## [1] 320
```

```
college_scorecard$INSTNM = process_school_name(college_scorecard$INSTNM)
nrow(college_scorecard)
```

```
## [1] 7703
```

## Merge Payscale data

```
payscale = merge(payscale_college_type, payscale_region, by="School.Name", all = FALSE)
payscale %>% group_by(School.Name) %>% filter(n() > 1)

payscale_party = payscale %>%
  filter(School.Type == "Party")

payscale = payscale %>%
  filter(School.Type != "Party") %>%
  mutate(Is.Party = School.Name %in% payscale_party$School.Name)

nrow(payscale)
```

All the duplicate rows in Payscale data is because it duplicates all Party Schools. So we split whether or not is a party school into a separate column.

## Merge with College Scorecard

```
scorecard_payscale = merge(payscale, college_scorecard, by.y = "INSTNM", by.x = "School.Name")
scorecard_payscale %>% group_by(School.Name) %>% filter(n() > 1)

scorecard_payscale = scorecard_payscale %>%
  filter( !((School.Name == 'Union College' & STABBR != 'NY') |
           (School.Name == 'Wentworth Institute of Technology' & is.na(COSTT4_A))))

nrow(scorecard_payscale)
```

Two schools are duplicate after merged with College Scorecard data. After some search, we only keep Union College in New York because that is the only one in northwest. And we only keep one Wentworth Institute of Technology because the others do not have data.

## Merge with MRC

```
data = merge(scorecard_payscale, mrc_table10, by.x = "OPEID6", by.y = "super_opeid")
data %>% group_by(School.Name) %>% filter(n() > 1)

nrow(data)
```

No duplicate in this step

## Other processing

```
names(data)[names(data) == 'tier_name'] <- 'Tier'
earning_colnames = c("MN_EARN_WNE_P6", "MD_EARN_WNE_P6", "PCT25_EARN_WNE_P6", "PCT75_EARN_WNE_P6", "SD_EARN_WNE_P6")
cost_colnames = c("COSTT4_A")
attr_colnames = c("School.Name", "School.Type", "Region", "Is.Party", "STABBR", "CONTROL", "Tier")

data$CONTROL = factor(data$CONTROL, levels = c(1,2), labels = c("Public", "Private nonprofit"))
```

We reorder the column for easy inspection. And convert CONTROL into factor.

## Accuracy and Outlier

### Accuracy & Missing value

```
summary(data)

percentmiss <- function(x){length(x[is.na(x)])/length(x)*100}

# process column first will get more records left
missing_col = apply(data, 2, percentmiss)
missing_col
delete <- which(missing_col > 5)
replace_col = data[,-delete]
dont_col = data[,delete]

missing_row = apply(replace_col, 1, percentmiss)
missing_row[missing_row > 5]
replace_row = subset(replace_col, missing_row <= 5)
dont_row = subset(replace_col, missing_row > 5)

# change to "cart" to avoid error, increase iteration to get reliable result
temp_no_miss = mice(replace_row, maxit=100, method='cart', seed=500)
no_miss = complete(temp_no_miss,1)

# combine data back
all_rows = rbind(dont_row, no_miss)
all_col = cbind(dont_col, all_rows)

data = all_col
```

There is no accuracy problem in the data. We use mice to complete the missing value for data meet 5% rule.

### Outlier

```
# pass tolerance to prevent mahalanobis think it is singular matrix
attr_index = which(colnames(no_miss) %in% attr_colnames)
mahal <- mahalanobis(no_miss[-c(attr_index)],
                     colMeans(no_miss[-c(attr_index)], na.rm=TRUE),
                     cov(no_miss[-c(attr_index)], use = "pairwise.complete.obs"),
```

```

                                tol=1e-30)
cutoff = qchisq(1-.001,ncol(no_miss[-c(attr_index)]))
print(cutoff)

## [1] 43.8202

summary(mahal < cutoff)

##      Mode   FALSE    TRUE
## logical      15     128

noout = subset(no_miss, mahal < cutoff)

no_miss[mahal >= cutoff, c("School.Name", "COSTT4_A", "MN_EARN_WNE_P6", "MD_EARN_WNE_P6", "MN_EARN_WNE_P10")]

##               School.Name COSTT4_A MN_EARN_WNE_P6
## 16                Pomona College   59730       51200
## 26          Colorado School of Mines   30154       74700
## 30                Yale University   61620       67800
## 55          Bowdoin College       59900       57100
## 56          Colby College       59110       50200
## 57          Amherst College       61544       61600
## 59    Massachusetts Institute of Technology   59020       99600
## 61          Williams College       61850       51400
## 76          Princeton University       57400       73600
## 78 New Mexico Institute of Mining and Technology   18762       49000
## 105                Reed College       59595       34800
## 108          Carnegie Mellon University       61990       84000
## 115          University of Pennsylvania       61800       91200
## 135 Washington and Lee University       58575       58900
## 143 Florida International University       18784       41500
##      MD_EARN_WNE_P6 MN_EARN_WNE_P10 MD_EARN_WNE_P10
## 16           41100           77300           58100
## 26           69200           95600           84900
## 30           56600          124400           83200
## 55           44600           83300           65500
## 56           42700           71000           58100
## 57           44100           83300           65000
## 59           82200          153600          104700
## 61           42600           89800           59000
## 76           60800          116300           74700
## 78           43500           58500           50000
## 105          30400           52700           42200
## 108          69800          103000           83600
## 115          71600          131600           85900
## 135          49900           93300           76100
## 143          38700           52000           46300

```

The outliers are more than 5% of data, so we keep them.

## ROI Analysis

```

## Break-even point calculation
data$BE_MN_P6 = data$MN_EARN_WNE_P6/data$COSTT4_A

```

```

data$BE_MD_P6 = data$MD_EARN_WNE_P6/data$COSTT4_A
data$BE_MN_P10 = data$MN_EARN_WNE_P10/data$COSTT4_A
data$BE_MD_P10 = data$MD_EARN_WNE_P10/data$COSTT4_A

## ROI USING EARNINGS of beiginig salary
data$ROI_MN_P6 = (data$MN_EARN_WNE_P6 - data$COSTT4_A) / data$COSTT4_A
data$ROI_MD_P6 = (data$MD_EARN_WNE_P6 - data$COSTT4_A) / data$COSTT4_A

## ROI USING MEAN EARNINGS of 6-year salary
data$ROI_MN_P10 = (data$MN_EARN_WNE_P10 - data$COSTT4_A) / data$COSTT4_A
data$ROI_MD_P10 = (data$MD_EARN_WNE_P10 - data$COSTT4_A) / data$COSTT4_A
roi_colnames = c("BE_MN_P6", "BE_MD_P6", "BE_MN_P10", "BE_MD_P10", "ROI_MN_P6", "ROI_MD_P6", "ROI_MN_P10", "ROI_MD_P10")

```

## Data Output

```

data_gender_p6 = melt(data[c("School.Name", "MN_EARN_WNE_MALE0_P6", "MN_EARN_WNE_MALE1_P6")], id=c("School.Name"),
levels(data_gender_p6$Gender) = c("Male", "Female")
colnames(data_gender_p6)[3] = "MN_EARN_WNE_GENDER_P6"

data_gender_p10 = melt(data[c("School.Name", "MN_EARN_WNE_MALE0_P10", "MN_EARN_WNE_MALE1_P10")], id=c("School.Name"),
levels(data_gender_p10$Gender) = c("Male", "Female")
colnames(data_gender_p10)[3] = "MN_EARN_WNE_GENDER_P10"

data_inc_p6 = melt(data[c("School.Name", "MN_EARN_WNE_INC1_P6", "MN_EARN_WNE_INC2_P6", "MN_EARN_WNE_INC3_P6")], id=c("School.Name"),
colnames(data_inc_p6)[3] = "MN_EARN_WNE_INC_P6"
levels(data_inc_p6$Family.Income.Tercile) = c("Lowest Income Tercile", "Middle Income Tercile", "Highest Income Tercile")

data_inc_p10 = melt(data[c("School.Name", "MN_EARN_WNE_INC1_P10", "MN_EARN_WNE_INC2_P10", "MN_EARN_WNE_INC3_P10")], id=c("School.Name"),
colnames(data_inc_p10)[3] = "MN_EARN_WNE_INC_P10"
levels(data_inc_p10$Family.Income.Tercile) = c("Lowest Income Tercile", "Middle Income Tercile", "Highest Income Tercile")

data = data[c(attr_colnames, cost_colnames, earning_colnames, roi_colnames)]
#write_csv(data, "data_cleaned.csv")
#write_csv(data_gender_p6, "data_gender_p6.csv")
#write_csv(data_gender_p10, "data_gender_p10.csv")
#write_csv(data_inc_p6, "data_inc_p6.csv")
#write_csv(data_inc_p10, "data_inc_p10.csv")

```

## Algorithm and Models

```

##Read the 4th dataset
salary_degree <-
  read_csv(
    'degrees-that-pay-back.csv',
    col_names = c(
      "major",
      "start_med_slry",
      "mid_car_slry",
      "percent_chng",
      "mid_car_10th",

```

```

    "mid_car_25th",
    "mid_car_75th",
    "mid_car_90th"
  ),
  col_types = "cnndnnnn", # specify column types to coerce '$' to numeric
  skip = 1 # names specified, skip header
)

#####
cleanup<-theme(panel.grid.major = element_blank(),
               panel.grid.minor = element_blank(),
               panel.background = element_blank(),
               axis.line.x = element_line(color = 'black'),
               axis.line.y = element_line(color = 'black'),
               legend.key = element_rect(fill = 'white'),
               text = element_text(size = 12))

## Plot degree v/s starting salary
p1 <- ggplot(salary_degree, aes(x = reorder(major, start_med_slry), start_med_slry)) +
  geom_col(fill = "blue", alpha = 0.5) +
  geom_col(aes(x = reorder(major, mid_car_slry), mid_car_slry), alpha = 0.3) +
  geom_text(aes(label = (start_med_slry)), size = 3, hjust = 1.1, col="yellow")+
  xlab(NULL) +
  ylab("Salary in $")+
  coord_flip() +
  ggtitle("Starting salary v/s mid career salary in $")+
  cleanup
p1

```

## Starting salary v/s mid career salary in \$



## Top 10 school

```
accent_colors_edit <- brewer.pal(n = 5, "Pastel1")[c(1:3, 5)] # keep colors consistent for plot w/o 'p'
```

```
top10_colleges <- data %>%
  select(School.Name, School.Type, MD_EARN_WNE_P8) %>%
  arrange(desc(MD_EARN_WNE_P8)) %>%
  top_n(10)
```

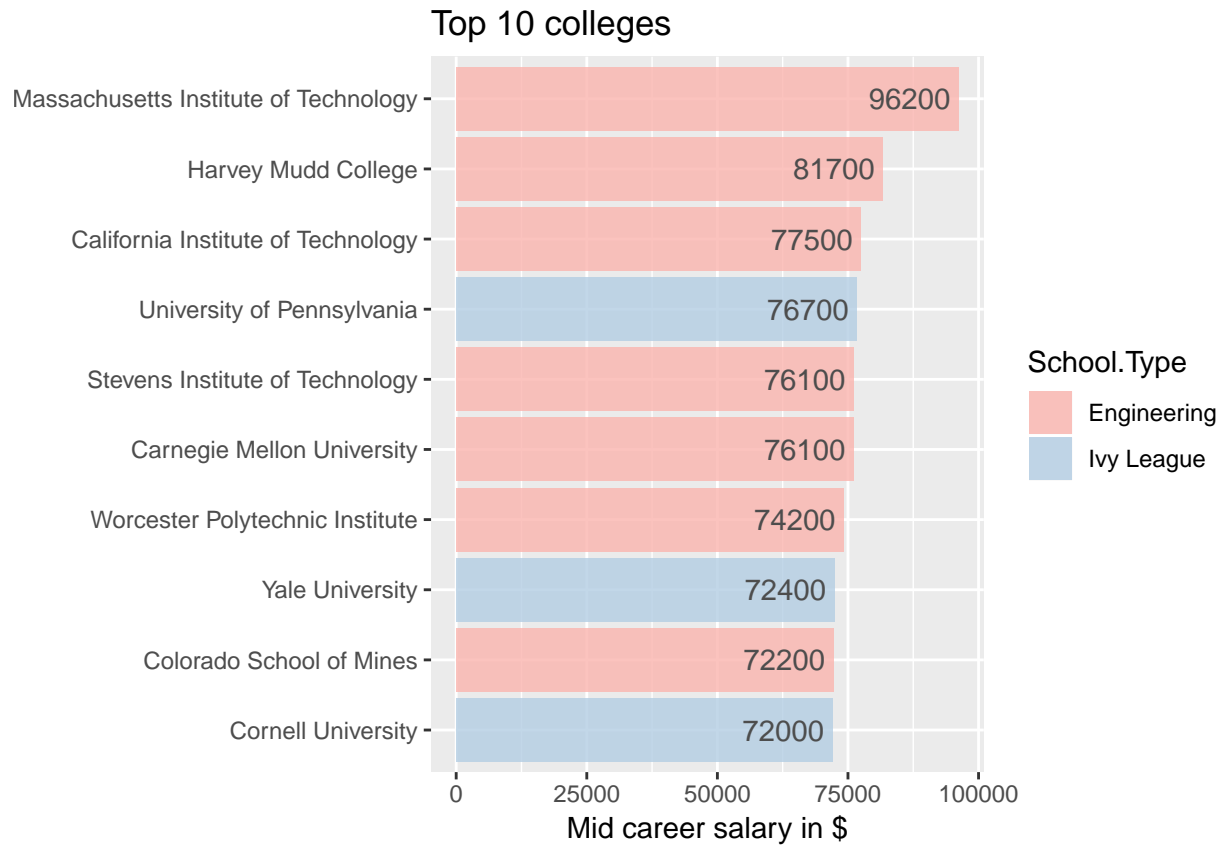
## Selecting by MD\_EARN\_WNE\_P8

```
top10_colleges
```

```
##
## 1 Massachusetts Institute of Technology Engineering 96200
## 2 Harvey Mudd College Engineering 81700
## 3 California Institute of Technology Engineering 77500
## 4 University of Pennsylvania Ivy League 76700
## 5 Stevens Institute of Technology Engineering 76100
## 6 Carnegie Mellon University Engineering 76100
## 7 Worcester Polytechnic Institute Engineering 74200
## 8 Yale University Ivy League 72400
## 9 Colorado School of Mines Engineering 72200
## 10 Cornell University Ivy League 72000
```

```
ggplot(top10_colleges, aes(reorder(School.Name, MD_EARN_WNE_P8), MD_EARN_WNE_P8, fill = School.Type)) +
  geom_col(alpha = 0.8) +
  scale_fill_manual(values = accent_colors_edit) +
```

```
geom_text(aes(label = (MD_EARN_WNE_P8)), hjust = 1.1, color = 'gray30') +
  xlab(NULL) +
  ggtitle("Top 10 colleges") +
  ylab("Mid career salary in $") +
  coord_flip()
```

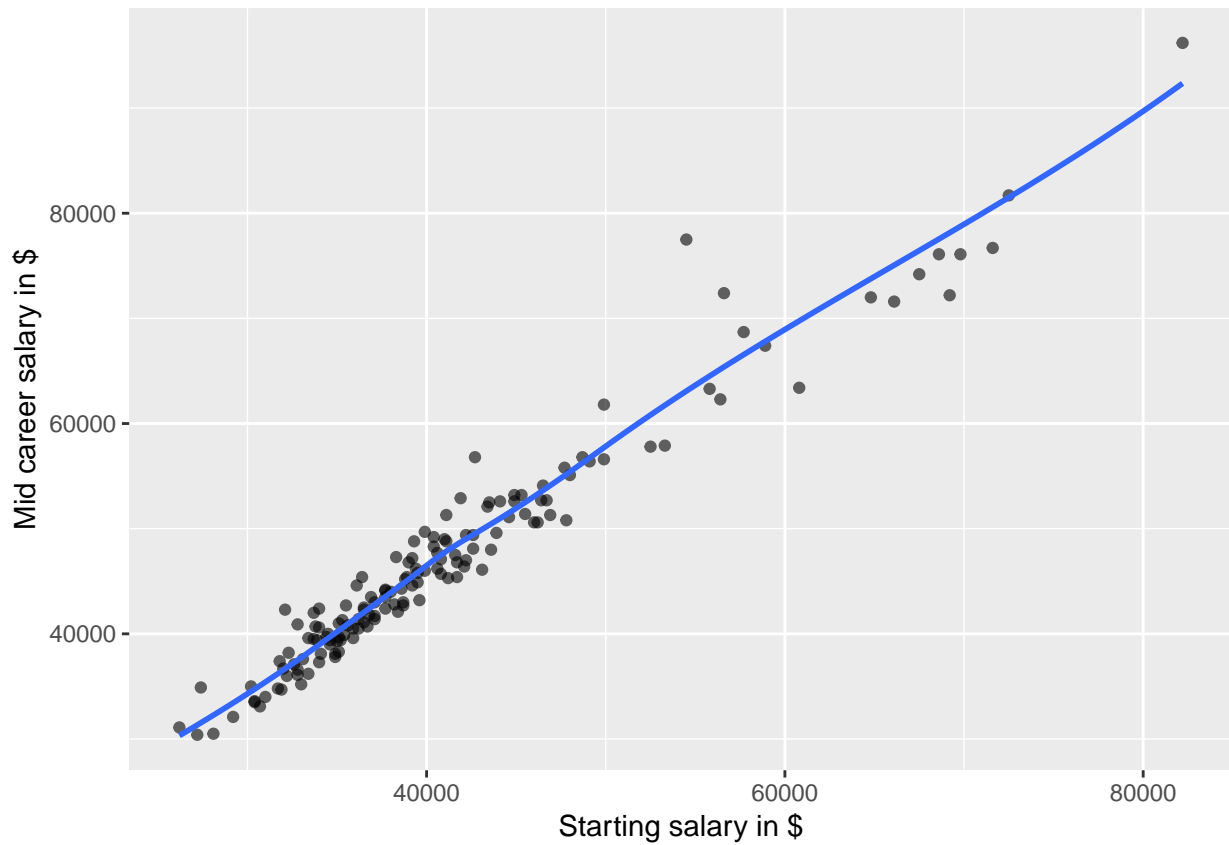


## Any correlation b/w starting salary and mid career salary

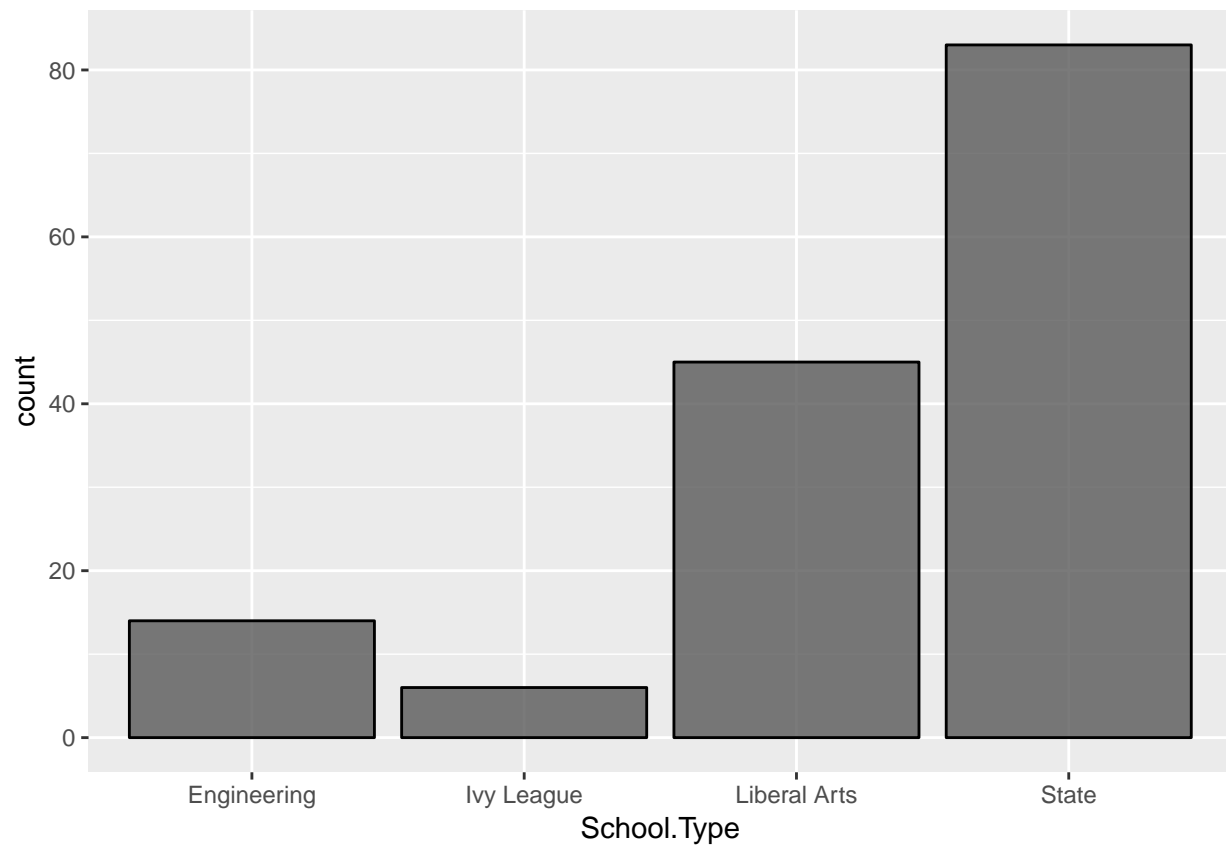
```
ggplot(data, aes(MD_EARN_WNE_P6, MD_EARN_WNE_P8)) +
  geom_point(alpha = 0.6) +
  geom_smooth(se = F) +
  xlab("Starting salary in $") +
  ylab("Mid career salary in $")
```

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'

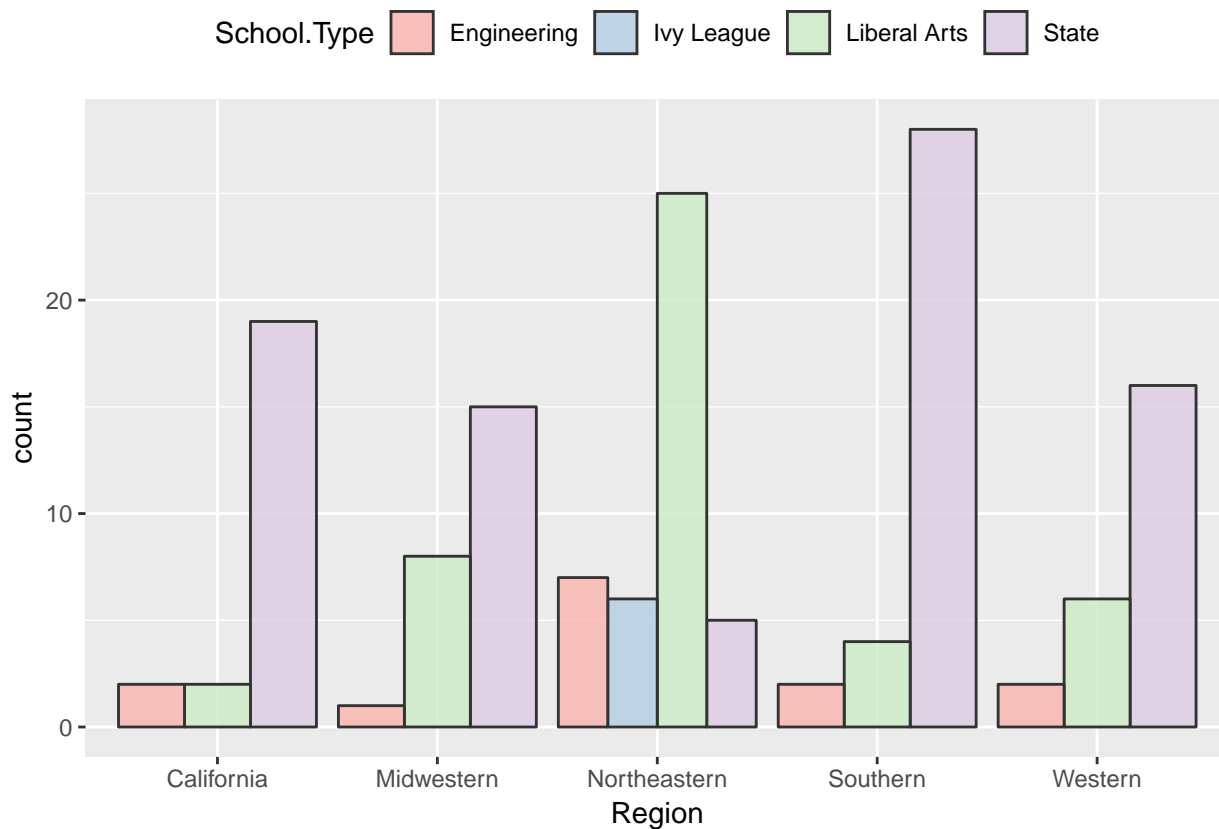




```
## Count by college type  
  
ggplot(data, aes(School.Type)) +  
  geom_bar(color = 'black', alpha = 0.8)
```



```
## Region and type
ggplot(data, aes(Region, fill = School.Type)) +
  geom_bar(position = 'dodge', alpha = 0.8, color = 'gray20') +
  scale_fill_brewer(palette = 'Pastel1') +
  theme(legend.position = "top")
```



*# How do starting salary and mid-career median salary differ over region and type?*

*#Below is a look at the mean starting and mid-career salaries over these two categories.*

```
ggplot(data, aes(reorder(Region, MN_EARN_WNE_P6), MD_EARN_WNE_P8, fill = School.Type)) +
  stat_summary(geom = 'col', position = 'dodge', alpha = 0.6) +
  stat_summary(aes(Region, MN_EARN_WNE_P6, fill = School.Type),
    geom = 'col', position = 'dodge') +
  scale_fill_brewer(palette = 'Pastell1') +
  xlab('Region') +
  ylab('Salaries in $') +
  ggtitle('Mean starting and Mid-career median salaries') +
  coord_flip()
```

## No summary function supplied, defaulting to `mean\_se()`

## No summary function supplied, defaulting to `mean\_se()`

