First 10 Graphs

Ben Goldstone

9/6/2023

College Distance Data Import

College Distance Dataset

```
collegeDistance <- read_csv("~/CSVs/CollegeDistance.csv")
```

```
## New names:
## * `` -> ...1
```

```
## Rows: 4739 Columns: 15
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (8): gender, ethnicity, fcollege, mcollege, home, urban, income, region
## dbl (7): ...1, score, unemp, wage, distance, tuition, education
```

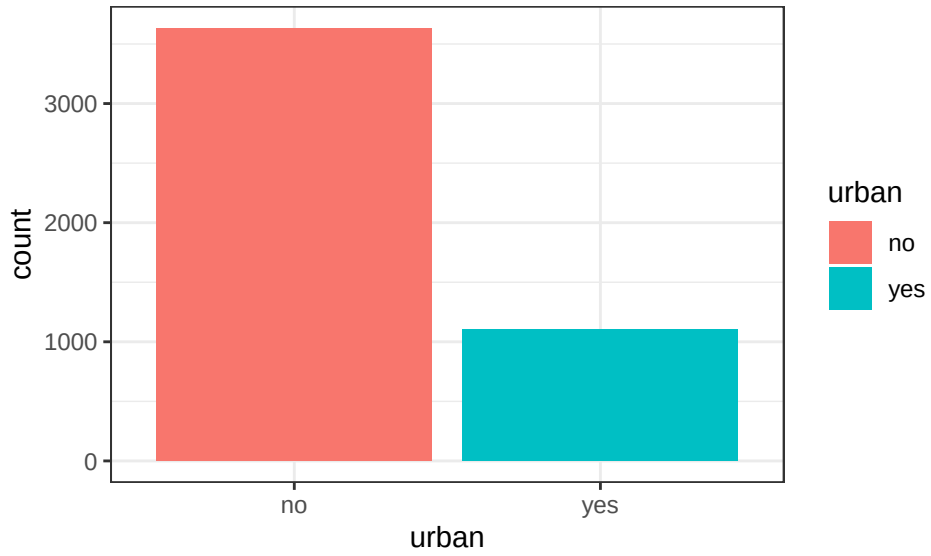
```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(collegeDistance)
```

```
## # A tibble: 6 x 15
##    ...1 gender ethnicity score fcollege mcollege home  urban unemp  wage
##   <dbl> <chr>  <chr>     <dbl> <chr>    <chr>    <chr> <chr> <dbl> <dbl>
## 1     1 male   other      39.2 yes      no       yes   yes    6.20  8.09
## 2     2 female other      48.9 no       no       yes   yes    6.20  8.09
## 3     3 male   other      48.7 no       no       yes   yes    6.20  8.09
## 4     4 male   afam       40.4 no       no       yes   yes    6.20  8.09
## 5     5 female other      40.5 no       no       no    yes    5.60  8.09
## 6     6 male   other      54.7 no       no       yes   yes    5.60  8.09
## # ... with 5 more variables: distance <dbl>, tuition <dbl>, education <dbl>,
## #   income <chr>, region <chr>
```

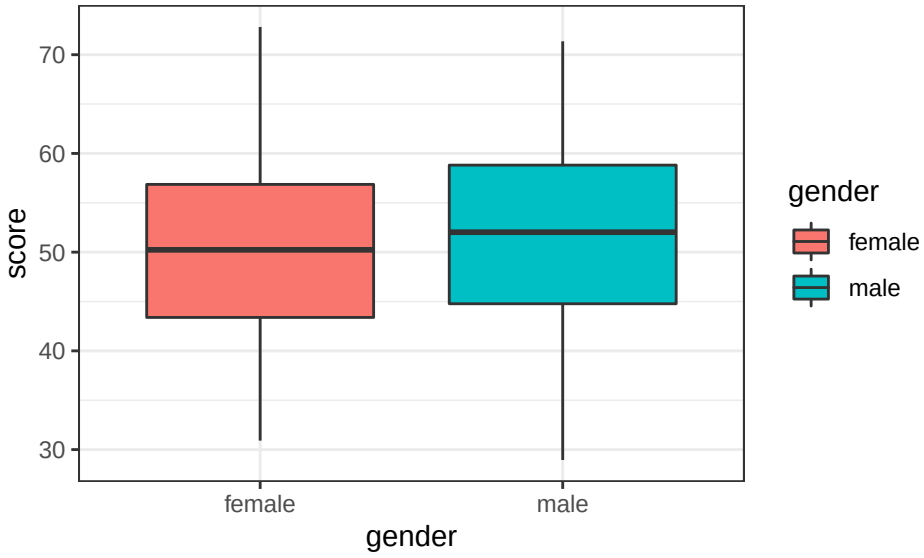
Urban College

```
gf_bar(~urban,data=collegeDistance,fill=~urban)
```



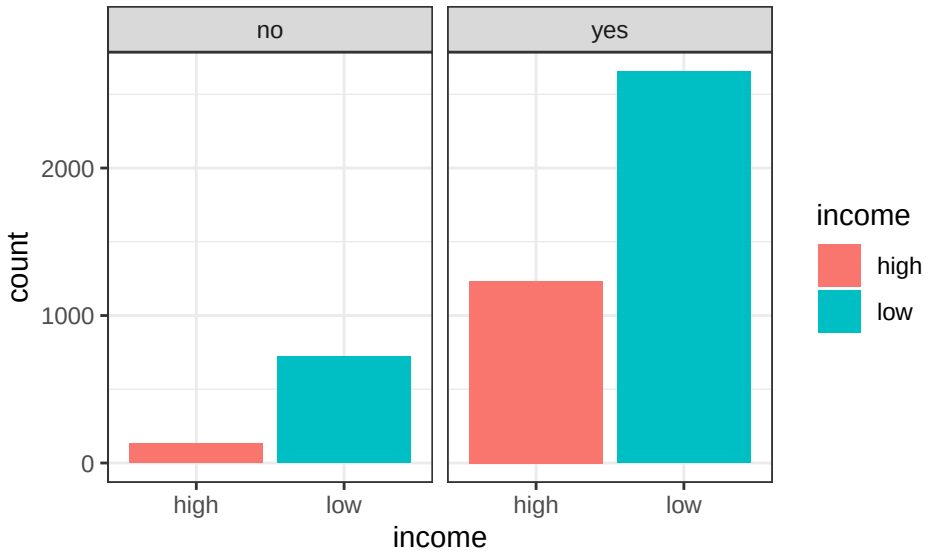
Composite test scores and gender

```
gf_boxplot(score~gender,data=collegeDistance, fill=~gender)
```



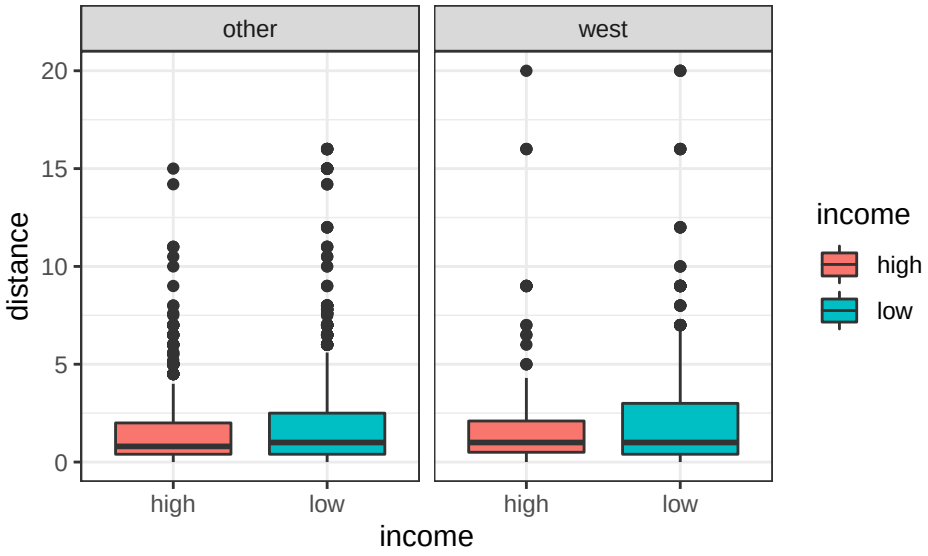
Home owner and income level

```
gf_bar(~income|home,data=collegeDistance, fill=~income)
```



Region and distance divided by income level

```
gf_boxplot(distance~income|region,data=collegeDistance, fill=~income)
```



China Income Data Import

China Income Dataset

```
ChinaIncome <- read_csv("~/CSVs/ChinaIncome.csv")
```

```
## Rows: 37 Columns: 6
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (6): year, agricultureIncome, commerceIncome, constructionIncome, indust...
```

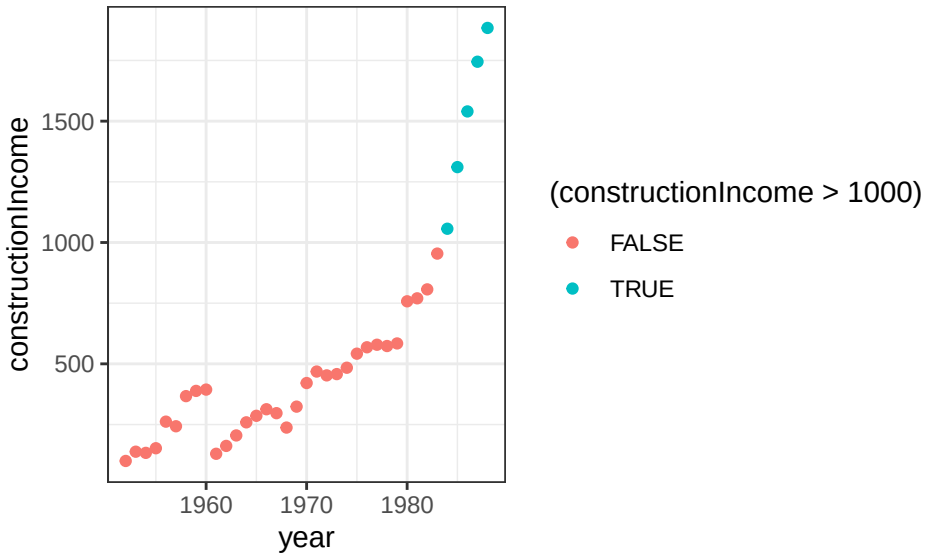
```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(ChinaIncome)
```

```
## # A tibble: 6 x 6
##    year agricultureIncome commerceIncome constructionIncome industryIncome
##   <dbl>             <dbl>          <dbl>               <dbl>          <dbl>
## 1  1952               100            100                 100            100
## 2  1953               102.           133                 138.           134.
## 3  1954               103.           136.                133.           159.
## 4  1955               112.           138.                152.           169.
## 5  1956               116.           147.                262.           219.
## 6  1957               120.           147.                243.           244.
## # ... with 1 more variable: transportIncome <dbl>
```

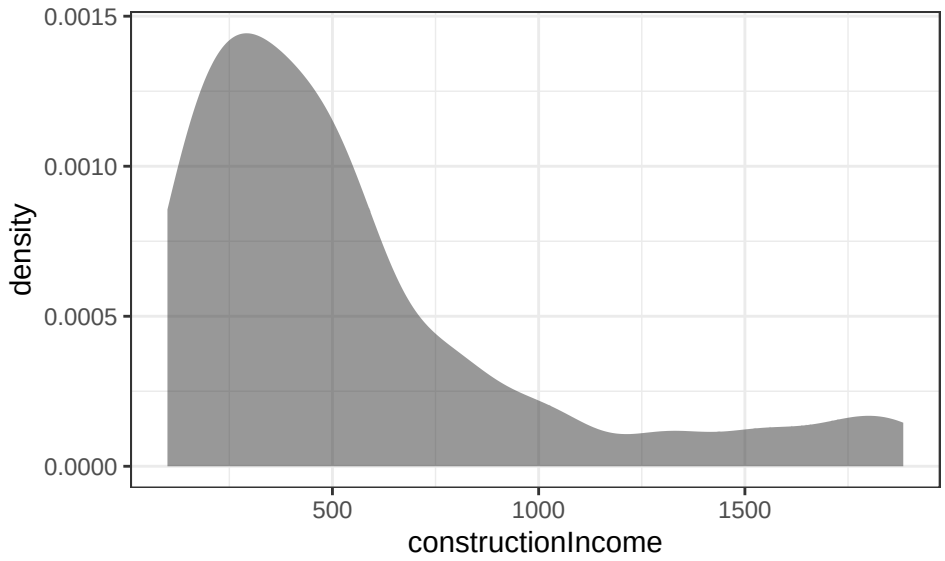
Construction Income over Time

```
gf_point(constructionIncome~year,data=ChinaIncome,color = ~ (constructionIncome > 1000))
```



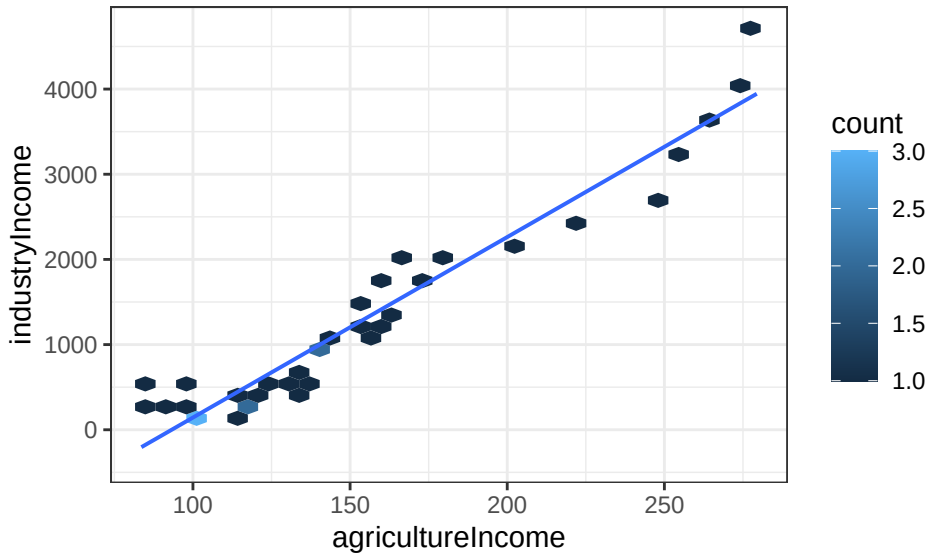
Construction Income

```
gf_density(~constructionIncome,data=ChinaIncome)
```



Industry Income and Agriculture Income

```
gf_hex(industryIncome~agricultureIncome, data=ChinaIncome) %>% gf_lm()
```



Consumer Goods Data Import

Consumer Goods Dataset

```
ConsumerGood <- read_csv("~/CSVs/ConsumerGood.csv")
```

```
## Rows: 108 Columns: 4
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (4): rownames, distribution, share, price
```

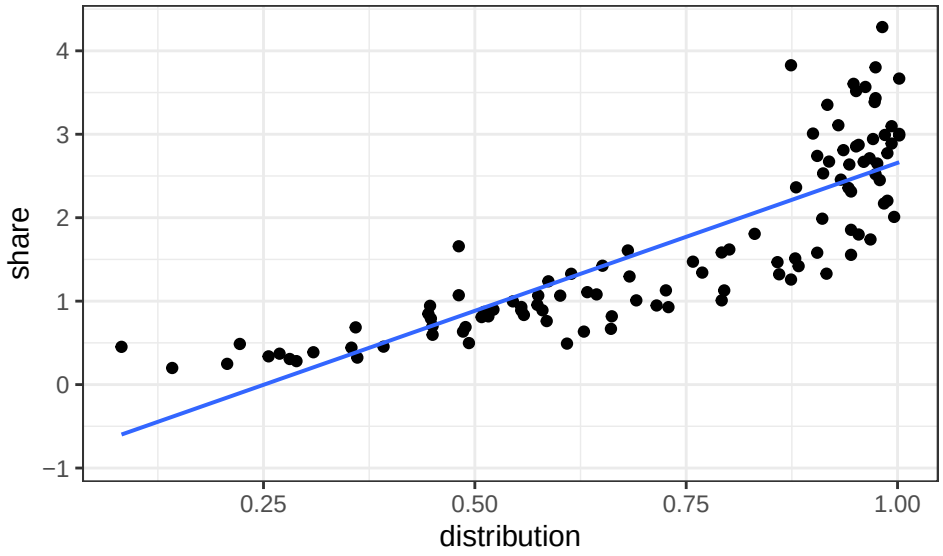
```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(ConsumerGood)
```

```
## # A tibble: 6 x 4
##   rownames distribution share price
##      <dbl>        <dbl> <dbl> <dbl>
## 1        1        0.905  2.74  106.
## 2        2        0.9    3.01  106.
## 3        3        0.988  2.20  107.
## 4        4        0.96   2.67  106.
## 5        5        0.954  2.87  106.
## 6        6        0.988  2.77  106.
```

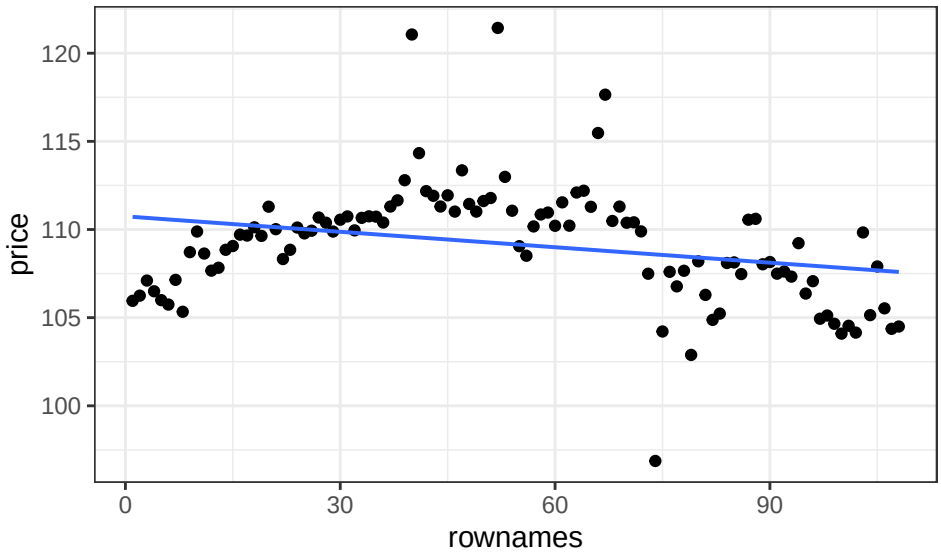
Share and distribution of consumer goods

```
gf_point(share~distribution, data=ConsumerGood) %>% gf_lm()
```



Price of Consumer Goods over time

```
gf_point(price~rownames,data=ConsumerGood) %>% gf_lm()
```



Price and distribution of Consumer Goods

```
gf_point(price~distribution,data=ConsumerGood) %>% gf_lm()
```