HW4_Mengdi

Mengdi Hao

2024-04-15

```
## Warning: package 'igraph' was built under R version 4.3.3
```

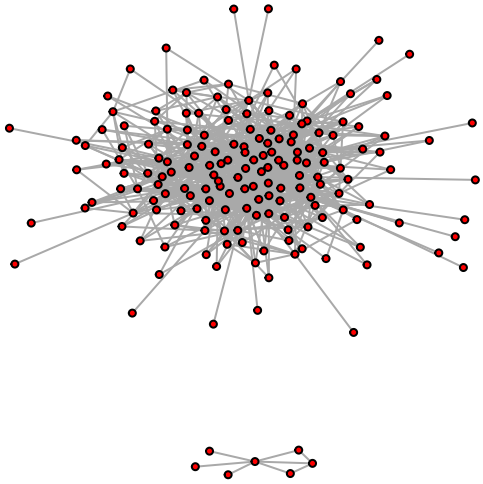
```
##
## Attaching package: 'igraph'
```

```
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
```

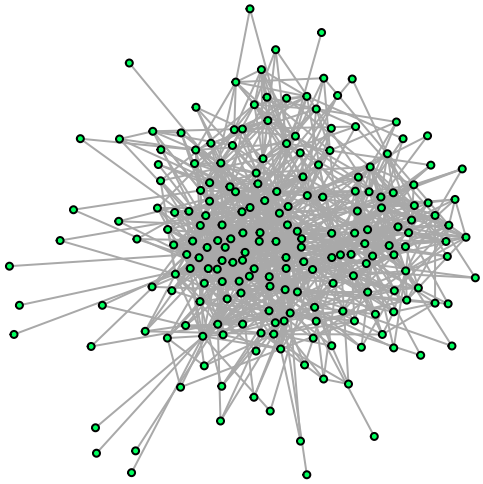
```
## The following object is masked from 'package:base':
##
##     union
```

```
## Warning: `graph.edgelist()` was deprecated in igraph 2.0.0.
## i Please use `graph_from_edgelist()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Warning: `induced.subgraph()` was deprecated in igraph 2.0.0.
## i Please use `induced_subgraph()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```
## Warning: package 'gamlr' was built under R version 4.3.3
## Loading required package: Matrix
```



Q1. I'd transform degree to create our treatment variable d. What would you do and why?

I would apply a logarithmic transformation to the degree variable for the following reasons: - Mitigating Skewness: Log transformation reduces right-skewness in count data like network degrees, making the distribution more normal and compatible with linear regression assumptions. - Linearizing Relationships: It transforms exponential relationships into linear ones, simplifying model interpretation and fitting. - Interpreting as Ratios: Post-transformation, coefficients represent percentage changes, which is useful in interpreting the result.

```
# Use log1p to avoid problems when degree = 0
hh$degree_log <- log1p(degree)
```

Q2. Build a model to predict d from x, our controls. Comment on how tight the fit is, and what that implies for estimation of a treatment effect.

The In-Sample R2 for this first stage LASSO model is 0.0817, which means around 8.17% of variations in our treatment variable is predicted by the other control variables. This means the majority of variations in the treatment variable is uncorrelated with the control variables. We could expect a significant treatment effect of the degree of connection on the probability of making loans.

```
# Two-stage LASSO
```

```r
library(gamlr)

# Convert missing values of categorical variables into reference level
hh <- naref(hh)

# Contruct the sparse desgin matrix, excluding "degree_log" and the intercept
x <- model.matrix(~ . - loan - degree_log - 1, data = hh)

# Define the treatment variable
d <- hh$degree_log

# Define the dependent variable
y <- hh$loan

# First stage LASSO: treatment variable d on control variables x
treat <- gamlr(x,d,lambda.min.ratio=1e-4)

# Predict treatment variable using control variables x
dhat <- predict(treat, x, type="response")

# Calculate IS (in-sample) R^2
cat("In-Sample R2 of the first stage model:", cor(drop(dhat),d)^2, "\n")
```

```
## In-Sample R2 of the first stage model: 0.08166857
```

Q3. Use predictions from [2] in an estimator for effect of d on loan.

The coefficient of degree_log is 0.0181, which indicates that for every percentage increase in the degree of connection, the odds of the family making a loan multiplies by 0.0181. This shows that there is a highly negative relationship between the degree of connection and the probability of making loans.

```r
# Second stage LASSO
causal <- gamlr(cbind(d,dhat,x),y,free=2,lmr=1e-4)
```

```
## 'as(<dgeMatrix>, "dgCMatrix")' is deprecated.
## Use 'as(., "CsparseMatrix")' instead.
## See help("Deprecated") and help("Matrix-deprecated").
```

```r
# Extract the treatment effect coefficient
cat("Treatment effect coefficient of degree_log:", coef(causal)["d",], "\n")
```

```
## Treatment effect coefficient of degree_log: 0.01812068
```

Q4. Compare the results from [3] to those from a straight (naive) lasso for loan on d and x. Explain why they are similar or different.

The coefficient of degree_log in the naive LASSO model is 0.1486, which also indicates a negative causal relationship between the degree of connection and the probability of making loans, but the odds multiplier increases a little bit, thus the relationship becomes less negative. This is very different from what we see in the previous two stage LASSO because this naive LASSO simply puts the treatment variable and the other control variables in the regression. This may lead to dropping some important confounding variables and leave the coefficient of the treatment variable containing confounding effect.

```r
# NAIVE lasso: directly regress y on x and d
naive <- gamlr(x = cbind(x,d), y = y, family = "binomial")

# Extract the treatment effect coefficient from the naive LASSO
cat("Treatment effect coefficient of degree_log:", coef(naive)["d",], "\n")
```

```
## Treatment effect coefficient of degree_log: 0.1485664
```

Q5. Bootstrap your estimator from [3] and describe the uncertainty.

The sampling and estimation process is conducted for 100 times. The mean of these attempts is 0.0169 and standard error is 0.0044. Relative to the estimated average, its standard error is low. This indicates that the result is fairly robust.

```r
# Bootstrap method to calculate SE
n <- nrow(x)
gamb <- c() # empty gamma

for(b in 1:100){

  ## create a matrix of resampled indices
    ib <- sample(1:n, n, replace=TRUE)

    ## create the resampled data
    xb <- x[ib,]
    db <- d[ib]
    yb <- y[ib]

    ## run the treatment regression
    # first stage
    treatb <- gamlr(xb,db,lambda.min.ratio=1e-3)
    dhatb <- predict(treatb, xb, type="response")
    # second stage
    fitb <- gamlr(cbind(db,dhatb,xb),yb,free=2)
    gamb <- c(gamb,coef(fitb)["db",])
}

# Summary statistics of 100 estimators
cat("Summary statistics of 100 estimators:", "\n")
```

```
## Summary statistics of 100 estimators:
```

```r
summary(gamb)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.002073 0.014492 0.018237 0.017184 0.020163 0.028014
```

```r
# Standard error of treatment effect using bootstrap
cat("Standard error of treatment effect:", sd(gamb), "\n")
```

```
## Standard error of treatment effect: 0.00462403
```

More: Can you think of how you'd design an experiment to estimate the treatment effect of network degree?