Methods for Utility Evaluation #1

Jingchen (Monika) Hu

Vassar College

Data Confidentiality

Outline

Outline

Global vs analysis-specific utility measures

- Global utility measures
  - examples?
  - pros and cons?

Global vs analysis-specific utility measures

- Global utility measures
  - examples?
  - pros and cons?
- Analysis-specifc utility measures
  - examples?
  - pros and cons?

Outline

Goals and three global utility measures

Woo et al. (2009)

- Discreminating between the original and the synthetic data using common statistical techniques.
  - Propensity score measure
  - Cluster anlaysis measure
  - Empirical CDF measure
- What are your thoughts about each measure?

Propensity score measure

- Propensity score matching is a commonly used technique.
  - estimate the effect of a treatment, policy, or other intervention
  - two groups: A (intervention) vs B (no intervention)
  - predict whether each unit has received the intervention or not
  - check how good the predictions are

Propensity score measure

- Propensity score matching is a commonly used technique.
  - estimate the effect of a treatment, policy, or other intervention
  - two groups: A (intervention) vs B (no intervention)
  - predict whether each unit has received the intervention or not
  - check how good the predictions are
- When used as a utility measure, the intervention is synthetic

Propensity score measure calculation

1. Merge the original and the synthetic datasets (recall that they have the same dimension n-by-p) by
   - stacking them together
   - resulting a merged dataset of dimension $2n$-by-p

Propensity score measure calculation

1. Merge the original and the synthetic datasets (recall that they have the same dimension n-by-p) by
   - stacking them together
   - resulting a merged dataset of dimension $2n$-by-p

2. Add an additional variable, S. For record i $(i = 1, \cdots, 2n)$
   - if it comes from the original dataset, set $S_i = 0$
   - if it comes from the synthetic dataset, set $S_i = 1$

Propensity score measure calculation

1. Merge the original and the synthetic datasets (recall that they have the same dimension n-by-p) by
   - stacking them together
   - resulting a merged dataset of dimension $2n$-by-p

2. Add an additional variable, S. For record i $(i = 1, \cdots, 2n)$
   - if it comes from the original dataset, set $S_i = 0$
   - if it comes from the synthetic dataset, set $S_i = 1$

3. For each record i $(i = 1, \cdots, 2n)$,
   - compute the probability of being in the synthetic dataset, using techniques such as logistic regression
   - this probability is the estimated propensity score, denoted as $\hat{p}_i$

Propensity score measure calculation cont'd

4. Compare the distributions of the propensity scores in the original and the synthetic datasets. Similarity can be assessed by comparisons of percentiles, as:

$$U_p = \frac{1}{2n}\sum_{i=1}^{2n}(\hat{p}_i - c)^2 \tag{1}$$

   - $2n$ is the number of records in the merged dataset
   - $\hat{p}_i$ is the estimated propensity score for unit i
   - c is the proportion of units with synthetic data in the merged dataset, typically $c = \frac{1}{2}$

Propensity score measure implications

$$U_p = \frac{1}{2n}\sum_{i=1}^{2n}(\hat{p}_i - \frac{1}{2})^2$$

- High level of similarity between the original and the synthetic data:
  - high percentage of $\hat{p}_i$ in the merged dataset close to $c = \frac{1}{2}$
  - $U_p \approx 0$
- Low level of similarity between the original and the synthetic data:
  - high percentage of $\hat{p}_i$ in the synthetic dataset close to 1 and that in the original dataset close to 0
  - $U_p \approx \frac{1}{4}$

Propensity score measure implications

$$U_p = \frac{1}{2n}\sum_{i=1}^{2n}(\hat{p}_i - \frac{1}{2})^2$$

- High level of similarity between the original and the synthetic data:
  - high percentage of $\hat{p}_i$ in the merged dataset close to $c = \frac{1}{2}$
  - $U_p \approx 0$
- Low level of similarity between the original and the synthetic data:
  - high percentage of $\hat{p}_i$ in the synthetic dataset close to 1 and that in the original dataset close to 0
  - $U_p \approx \frac{1}{4}$

In sum, the closer the value U_p is to 0, the higher the similarity level between the original and the synthetic data, indicating high utility. The closer the value U_p is to $\frac{1}{4}$, the lower the similarity level between the original and the synthetic data, indiciting low utility.

Propensity score measure example: synthetic CE sample

- Previously, we have worked with the CE sample:
  - a Bayesian simple linear regression synthesis model
  - synthesize `log(Income)` given `log(Expenditure)`
  - one synthetic dataset saved in `synthetic_one`

```
n <- dim(CEdata)[1]
synthetic_one <- synthesize(CEdata$LogExpenditure, 1, n, seed = 123)
names(synthetic_one) <- c("LogExpenditure", "LogIncome")
```

Synthetic CE sample: step 1

- Merge two datasets and add S variable

```
CEdata_twovars <- as.data.frame(cbind(CEdata$LogExpenditure,
                                      CEdata$LogIncome))
names(CEdata_twovars) <- c("LogExpenditure", "LogIncome")
merged_data <- rbind(CEdata_twovars, synthetic_one)

merged_data$S <- c(rep(0, n), rep(1, n))
```

Synthetic CE sample: step 2

- Compute propensity scores with a logistic regression
- For illustration purpose, use a logistic regression of added variable `S` given the two explanatory variables, `LogExpenditure` and `LogIncome`
- Interaction terms could be used as well

$$\log\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 LogExpenditure_i + \beta_2 LogIncome_i. \qquad (2)$$

Synthetic CE sample: step 2 cont'd

The `glm()` function is used to implement a logistic regression, with `family = "binomial"`.

```
log_reg <- glm(S ~ LogExpenditure + LogIncome, family = "binomial",
               data = merged_data)
```

Synthetic CE sample: step 2 cont'd

- The `predict()` function calculates and returns `b0 + b1*x1 + b2*x2`
  - `x1 = LogExpenditure`
  - `x2 = LogIncome`
  - `b0, b1, b2` are estimates for $\beta_0, \beta_1, \beta_2$ respectively

```
pred <- predict(log_reg, data = merged_data)
```

- Therefore in order to obtain $\hat{p}_i$, we need to use the following algebra transformation:

$$\begin{aligned} \log\left(\frac{p_i}{1-p_i}\right) &= \beta_0 + \beta_1 X_i \\ p_i &= \frac{\exp(\beta_0 + \beta_1 X_i)}{1 + \exp(\beta_0 + \beta_1 X_i)}. \end{aligned} \tag{3}$$

```
probs <- exp(pred)/(1+exp(pred))
```

Synthetic CE sample: step 3

- Calculate propensity score utility measure U_p

```
Up <- 1/(2*n)*sum((probs - 1/2)^2)
Up
```

```
## [1] 0.0001253122
```

- the calculated propensity score utility measure U_p is near 0
- the logistic regression model cannot really distinquish between the original and the synthetic datasets
- a high level of utility of our simulated synthetic data

Cluster analysis measure

- Cluster analysis is a commonly used technique
  - clustering records with similiar characteristics into the same group
  - and records clustered in different groups would share less similar characteristics
  - group characteristics (for example, the mean and standard deviation of a group-specific continuous variable) could have improved estimate
    - ★ similar records are clustered in the same group and share information
  - especially beneficial for clusters with small sample sizes

Cluster analysis measure

- Cluster analysis is a commonly used technique
  - clustering records with similiar characteristics into the same group
  - and records clustered in different groups would share less similar characteristics
  - group characteristics (for example, the mean and standard deviation of a group-specific continuous variable) could have improved estimate
    - ★ similar records are clustered in the same group and share information
  - especially beneficial for clusters with small sample sizes

- Various algorithms available for cluster analysis
  - understanding of what constitutes a cluster
  - how to efficiently find the clusters
- We can determine what features the cluster analysis should be based on when performing the cluster algorithm
  - we can choose all variables to be used for forming clusters vs only a subset

Cluster analysis measure calculation

- When used as a utility measure, we care about whether the measure can discreminate between the original and the synthetic data

1. Merge the original and the synthetic datasets (recall that they have the same dimension n-by-p) by
   - stacking them together
   - resulting a merged dataset of dimension $2n$-by-p

Cluster analysis measure calculation

- When used as a utility measure, we care about whether the measure can discreminate between the original and the synthetic data

1. Merge the original and the synthetic datasets (recall that they have the same dimension n-by-p) by
   - stacking them together
   - resulting a merged dataset of dimension $2n$-by-p
2. Add an additional variable, S. For record i $(i = 1, \cdots, 2n)$
   - if it comes from the original dataset, set $S_i = 0$
   - if it comes from the synthetic dataset, set $S_i = 1$

Cluster analysis measure calculation cont'd

3. Perform a cluster analysis on the merged dataset with a fixed number of groups, G. For each group g,
   - record the number of records clustered in this group, n_g
   - record the number of records from the original dataset is clustered in this group, n_g^S, where $n_g^S \leq n_g$

Cluster analysis measure calculation cont'd

3. Perform a cluster analysis on the merged dataset with a fixed number of groups, G. For each group g,
   - record the number of records clustered in this group, n_g
   - record the number of records from the original dataset is clustered in this group, n_g^S, where $n_g^S \leq n_g$

4. Use the following measure:

$$U_c = \frac{1}{G} \sum_{g=1}^{G} w_g \left(\frac{n_g^S}{n_g} - c\right)^2 \tag{4}$$

   - w_g is the weight assigned to cluster g (available from the clustering algorithm)
   - c is the proportion of units with synthetic data in the merged dataset, typically $c = \frac{1}{2}$

Cluster analysis measure implications

$$U_c = \frac{1}{G}\sum_{g=1}^{G} w_g(\frac{n_g^S}{n_g} - c)^2$$

- High level of similarity between the original and the synthetic data:
  - high percentage of $\frac{n_j^S}{n_j}$ in the cluster analysis close to $c = \frac{1}{2}$
  - $U_c \approx 0$
- Low level of similarity between the original and the synthetic data:
  - high percentage of $\frac{n_j^S}{n_j}$ in the cluster analysis close to either 0 or 1
  - a large value of U_c

In sum, the closer the value U_c is to 0, the higher the similarity level between the original and the synthetic data, indicating high utility. The further away the value U_c is from 0, the lower the similarity level between the original and the synthetic data, indiciting low utility.

Cluster analysis measure example: synthetic CE sample

- Previously, we have worked with the CE sample:
  - a Bayesian simple linear regression synthesis model
  - synthesize `log(Income)` given `log(Expenditure)`
  - one synthetic dataset saved in `synthetic_one`

```
n <- dim(CEdata)[1]
synthetic_one <- synthesize(CEdata$LogExpenditure, 1, n, seed = 123)
names(synthetic_one) <- c("LogExpenditure", "LogIncome")
```

Synthetic CE sample: step 1

- Merge two datasets and add S variable

```
CEdata_twovars <- as.data.frame(cbind(CEdata$LogExpenditure,
                                      CEdata$LogIncome))
names(CEdata_twovars) <- c("LogExpenditure", "LogIncome")
merged_data <- rbind(CEdata_twovars, synthetic_one)

merged_data$S <- c(rep(0, n), rep(1, n))
```

Synthetic CE sample: step 2

- Perform a cluster analysis
- For illustration purpose, we use the `hclust()` function which performs the hiererchical clustering algorithm

```
clusters <- hclust(dist(merged_data[, 1:2]), method = 'average')
```

Synthetic CE sample: step 2 cont'd

Due to the nature of hierarchical clustering algorithm, we can determine the number of groups, G, after the `hclust()` function is run. For example, if we set $G = 5$:

```
G <- 5
clusterCut <- cutree(clusters, G)
cluster_S <- as.data.frame(cbind(clusterCut, merged_data$S))
names(cluster_S) <- c("cluster", "S")
table(cluster_S)
```

```
##         S
## cluster   0   1
##       1 867 883
##       2  56  18
##       3  68  90
##       4   2   3
##       5   1   0
```

Synthetic CE sample: step 2 cont'd

We can then calculate our n_g^S, n_g and w_g for $g = 1, \cdots, G$ from `clusterCut` as follows

```
n_gS <- table(cluster_S)[, 1]
n_g <- rowSums(table(cluster_S))
w_g <- n_g / (2*n)
```

- `n_gS` contains the vector of $(n_1^S, \cdots, n_G^S)$
- `n_g` contains the vector of $(n_1, \cdots, n_G)$
- `w_g` contains the vector of $(w_1, \cdots, w_G)$ (the weights `w_g` are calculated as $\frac{n_g}{2n}$ as the percentage of records clustered in group g)

Synthetic CE sample: step 3

- Calculate cluster analysis utility measure U_c

```
Uc <- (1/G) * sum(w_g * (n_gS/n_g - 1/2)^2)
Uc
```

```
## [1] 0.0006016874
```

- the calculated cluster anlaysis utility measure U_c is near 0
- the cluster analysis algorithm clusters roughly equal numbers of records from the original data and the synthetic data, into the same group
- this means that the cluster analysis algorithm cannot really distinquish between the original and the synthetic datasets
- a high level of utility of our simulated synthetic data

Empirical CDF measure

- The empirical CDF distribution is the CDF associated with a given sample
- If two samples are similar, their empirical CDF distributions are similar
- When used as a utility measure, we care about whether the measure can discreminate between the original and the synthetic data

Empirical CDF measure calculation

1. Merge the original and the synthetic datasets (recall that they have the same dimension n-by-p) by
   - stacking them together
   - resulting a merged dataset of dimension $2n$-by-p

Empirical CDF measure calculation

1. Merge the original and the synthetic datasets (recall that they have the same dimension n-by-p) by
   - stacking them together
   - resulting a merged dataset of dimension $2n$-by-p

2. Estimate the
   - empirical CDF distribution of the original dataset, denoted as $ecdf^O$
   - empirical CDF distribution of the synthetic dataset, denoted by $ecdf^S$
   - using appropriate functions and methods

Empirical CDF measure calculation

1. Merge the original and the synthetic datasets (recall that they have the same dimension n-by-p) by
   - stacking them together
   - resulting a merged dataset of dimension $2n$-by-p

2. Estimate the
   - empirical CDF distribution of the original dataset, denoted as $ecdf^O$
   - empirical CDF distribution of the synthetic dataset, denoted by $ecdf^S$
   - using appropriate functions and methods

3. For record i $(i = 1, \cdots, 2n)$ in the merged dataset, estimate its
   - percentile under the empirical CDF distribution of the original dataset $ecdf^O$, denoted as p_i^O
   - percentile under the empirical CDF distribution of the synthetic dataset $ecdf^S$, denoted as p_i^S

Empirical CDF measure calculation cont'd

4. Use the following two measures:

- U_m:the maximum absolute difference between the empirical CDFs

$$U_m = \max_{1 \leq i \leq 2n} |p_i^O - p_i^S| \tag{5}$$

- U_a: the average squared differences between the empirical CDFs

$$U_a = \frac{1}{2n} \sum_{i=1}^{2n} (p_i^O - p_i^S)^2 \tag{6}$$

  - $2n$ is the number of records in the merged dataset

Empirical CDF measure implications

$$
\begin{aligned}
U_m &= \max_{1 \le i \le 2n} |p_i^O - p_i^S| \\
U_a &= \frac{1}{2n} \sum_{i=1}^{2n} (p_i^O - p_i^S)^2
\end{aligned}
$$

- High level of similarity between the original and the synthetic data
  - low values of U_m and U_a
- Low level of similarity between the original and the synthetic data
  - high values of U_m and U_a

In sum, the smaller the values of U_m and U_a, the higher the similarity level between the original and the synthetic data, indicating high utility. The larger the values of U_m and U_a, the lower the similarity level between the original and the synthetic data, indiciting low utility.

Empirical CDF measure example: synthetic CE sample

- Previously, we have worked with the CE sample:
  - a Bayesian simple linear regression synthesis model
  - synthesize `log(Income)` given `log(Expenditure)`
  - one synthetic dataset saved in `synthetic_one`

```
n <- dim(CEdata)[1]
synthetic_one <- synthesize(CEdata$LogExpenditure, 1, n, seed = 123)
names(synthetic_one) <- c("LogExpenditure", "LogIncome")
```

Synthetic CE sample: step 1

- Merge two datasets

```
CEdata_twovars <- as.data.frame(cbind(CEdata$LogExpenditure,
                                      CEdata$LogIncome))
names(CEdata_twovars) <- c("LogExpenditure", "LogIncome")
merged_data <- rbind(CEdata_twovars, synthetic_one)
```

Synthetic CE sample: step 2

- Estimate the two empirical CDFs
- Use the `ecdf()` function available in the `stats` R package to obtain
  - the empirical CDF of the original dataset, saved in `ecdf_orig`
  - the empirical CDF of the synthetic dataset, saved in `ecdf_syn`

```
ecdf_orig <- ecdf(CEdata_twovars[,"LogIncome"])
ecdf_syn <- ecdf(synthetic_one[,"LogIncome"])
```

- Note that here we are estimating the empirical CDFs using the `log(Income)` variable, which is synthesized in the synthetic dataset
- How to obtain empirical CDF of multivariate data?

Synthetic CE sample: step 3

- Estimate the percentiles of records in the merged dataset: $i = 1, \cdots, 2n$

```
ecdf_orig <- ecdf(CEdata_twovars[,"LogIncome"])
ecdf_syn <- ecdf(synthetic_one[,"LogIncome"])
```

```
percentile_orig <- ecdf_orig(merged_data[,"LogIncome"])
percentile_syn <- ecdf_syn(merged_data[,"LogIncome"])
```

Synthetic CE sample: step 3

- Calculate empirical CDF utility measures U_m and U_a

$$U_m = \max_{1 \le i \le 2n} |p_i^O - p_i^S|$$

$$U_a = \frac{1}{2n} \sum_{i=1}^{2n} (p_i^O - p_i^S)^2$$

```
ecdf_diff <- percentile_orig - percentile_syn
Um <- max(abs(ecdf_diff))
Um
```

```
## [1] 0.05231388
```

```
Ua <- mean(ecdf_diff^2)
Ua
```

```
## [1] 0.0007437977
```

Synthetic CE sample: step 3 cont'd

- the calculated empirical CDF utility measures U_m and U_a are small
- the empirical CDFs of the original dataset and of the synthetic dataset are similar
- this means that we cannot really distinquish between the empirical CDFs of the original and the synthetic datasets
- a high level of utility of our simulated synthetic data.

References

- Woo, M. J., Reiter, J. P., Oganian, A., and Karr, A. F. (2009). Global Measures of Data Utility for Microdata Masked for Disclosure Limitation. The Journal of Privacy and Confidentiality, 1(1), 111-124.