Omkar A. Katta January 27, 2021

Differences-in-Transports

PPHA 30545 Guillaume A. Pouliot

1 Introduction to Optimal Transport Theory **2**

1.1 Examples

1.2 Developing Intuition

1.3 Developing Theory

1.4 Aligning Intuition and Theory

2 Setting Up `Python` **8**

2.1 Installing `diftrans`

2.2 Installing Other Packages

3 Black Market for License Plates in Beijing (Daljord et al., 2021) **10**

3.1 Introduction

3.2 Understanding the Data

3.3 Clean Data of Beijing and Tianjin Car Sales

3.4 Visualize Beijing Car Sales

3.5 Compute Before-and-After Estimator

3.6 Compute Differences-in-Transports Estimator

4 Concluding Remarks **19**

4.1 Lower Bound

4.2 Simulations

4.3 Robustness Checks and Verifying Assumptions

4.4 `diftrans` package

Policymakers want to know: what is the impact of a certain policy? Clear answers to this question will impact the adoption of future policies and changes to old ones. But questions of policy evaluation and causal inference are overwhelmingly easier to ask than they are to answer.

In this homework, we will study an approach to comparing populations before and after some policy change. Unlike other most other techniques, this approach doesn't settle for looking at changes in summary statistics (e.g., the mean) of the population. Rather, this approach looks at the population as a whole, enabling you to answer questions that you would not have been able to before.

For example: how large is the black market for license plates? The defining aspect of the black market is that it is hidden from view. So as aspiring machine learning experts with a keen eye for public policy, how can we learn about something that we won't be able to see in data?

By the end of this homework, you will have the tools and understanding that is necessary for answering such an elusive question. We'll first go over some of the basics of *optimal transport theory* to understand how we can compare two distributions. Then, you will then apply this knowledge in a series of guided exercises to walk you through the analysis of Daljord et al. (2021), which answers this question by taking advantage of a policy change in Beijing, China.

1 Introduction to Optimal Transport Theory

The shortest path between two points is a straight line. To compute the distance between two points, we therefore measure the distance of this straight line. This simple idea has been a cornerstone of loss functions in machine learning techniques. For example, in linear regression, we consider the distance of the shortest path between the outcome variable of interest, y_i, and the fitted values, $\hat{y}_i$. We can the sum this distance across all i to get the total distance. This total distance is the loss function that we minimize. When the distance between y_i and $\hat{y}_i$ is measured using the sum of the squared differences (the l_2 norm), we call this "Ordinary Least Squares." If this distance is the sum of absolute values (the l_1 norm), we sometimes call this the "Median Regression."

Optimal transport theory extends this simple idea. Instead of comparing two points (e.g., y_i and $\hat{y}_i$ as in linear regression), optimal transport theory compares two *distributions*. How would we conceptualize the shortest path between two distributions? How would we then measure the distance of this path?
We will answer these questions by way of four examples.

1.1 Examples

Imagine a two-period world with three unique car prices: \$1, \$2, and \$10. The entries in the Table 1 denote how many cars were sold at each price in the "before" and "after" periods. For instance, according

to the "before" distribution of Example 2, 8 cars were sold at $1 and no cars were sold at $2. In the "after" distribution of the same example, 0 cars were sold at $1 and 8 cars were sold at $2.

Our goal will be to come up with some notion of distance between the "before" and "after" distributions in each example to quantify how much the distribution of car sales changed across the two periods. What properties do we want our notion of distance to have? One reasonable property is that if the distributions are exactly the same, then their distance should be zero. Thus, perhaps a natural way to define the distance between distributions is to look at what is the same between them and what is not. The amount of non-overlapping in our distributions will contribute to the distance. Let's start to develop our intuition from this insight.

Before we move on, notice that the total number of cars in each period is not the same: in the "before" period of Example 1, there are 8 cars in total, but in the "after" period of the same example, there are 4 cars in total. To compare the distributions, we will need to normalize them. Instead of looking at the *number* of cars sold at each price, let us consider the *proportion* of cars sold at each price. In the parlance of discrete probability, the proportion of cars sold at each price is the *mass* on each price (recall probability *mass* function). Table 2 normalizes the results in Table 1.

Table 1: Unnormalized Probability Mass Functions

	Example 1		Example 2		Example 3		Example 4	
Support	Before	After	Before	After	Before	After	Before	After
1	6	3	8	0	8	0	6	1
2	2	1	0	8	0	0	2	3
10	0	0	0	0	0	8	0	0

Table 2: Probability Mass Functions

	Example 1		Example 2		Example 3		Example 4	
Support	Before	After	Before	After	Before	After	Before	After
1	0.75	0.75	1	0	1	0	0.75	0.25
2	0.25	0.25	0	1	0	0	0.25	0.75
10	0.00	0.00	0	0	0	1	0.00	0.00

1.2 Developing Intuition

An example of two distributions that are the same is Example 1. In both periods, 75% of cars were sold at $1 while the rest were sold at $2. Hence, the distributions in each period overlaps completely, i.e., there is no change.

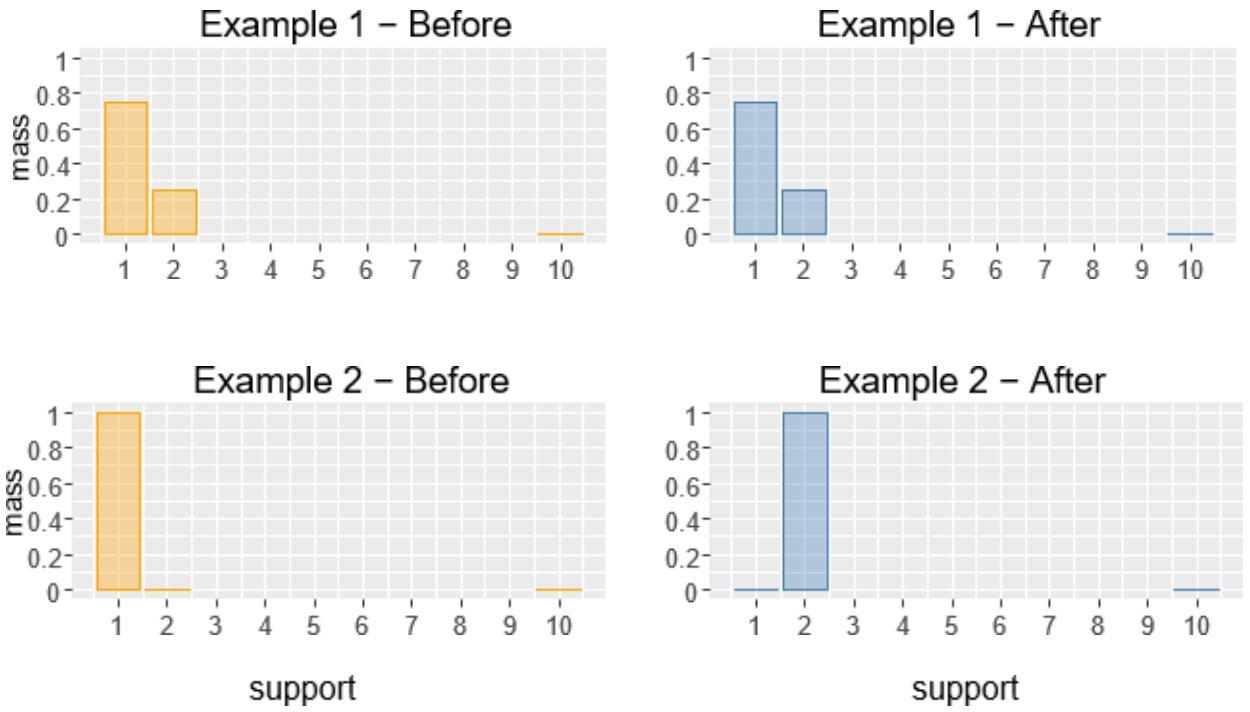
In sharp contrast, Examples 2 and 3 don't have overlap in our "before" and "after" distributions. Both examples have the same "before" distribution: all the cars were sold at $1 in the "before" period. The "after" distributions, however, are different: all the cars were sold at $2 in Example 2, but all the cars were sold at $10 in Example 3. Between these two examples, which would you say has the greatest distance between the "before" and "after" distributions? Well, the mass in Example 3 shifted from $1 to $2, whereas the mass in Example 3 shifted from $1 all the way to $10. Since the mass had to "travel" farther in Example 3, we can reasonably say that there is a greater distance between the distributions of Example 3 than the distributions of Example 2.

We can summarize what these examples teach us about distance in the following two questions:

1. What proportion of the mass moved between the "before" and "after" distributions?
2. How far did the mass move?

In our examples thus far, we had either complete overlap or no overlap at all. Example 4 provides two distributions where there is *some* overlap. Answering our two questions will require some more thought. Did the entire mass on $1 shift to $2 while the entire mass on $2 shifted to $1? Alternatively, did some of the mass stay where it was between the two distributions while the rest shifted? The partial overlap in Example 4 provides multiple choices for the "path" between the two distributions.

We want to choose the *shortest* path.



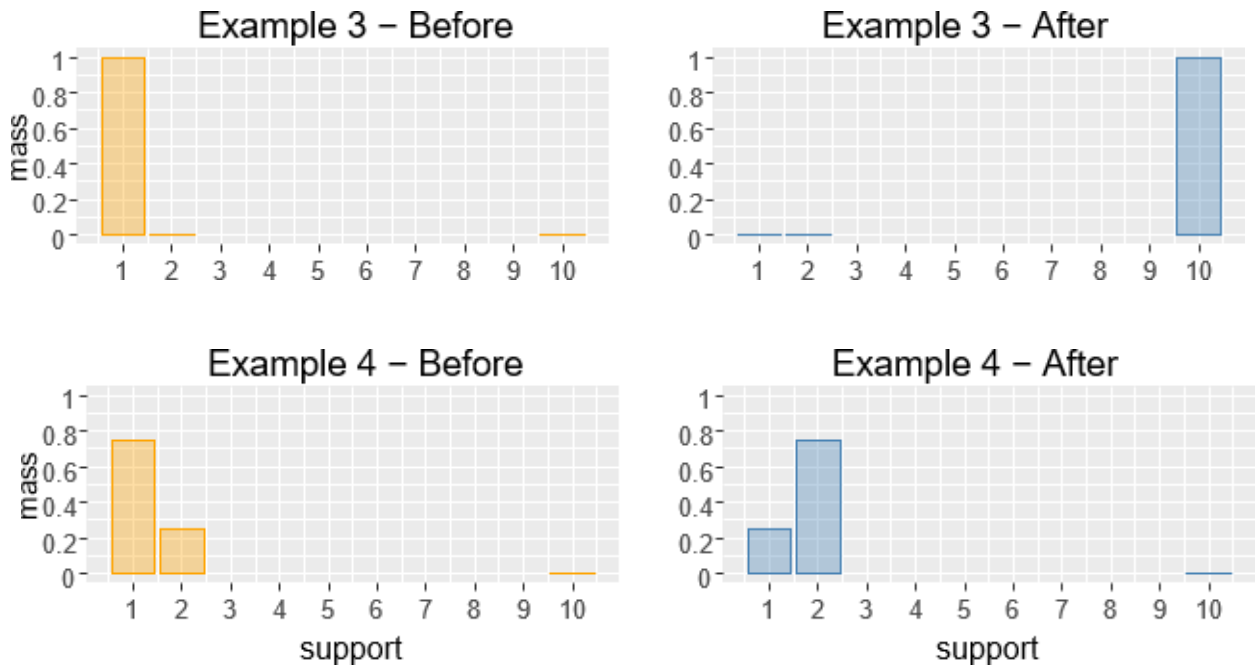


Figure 1: Before (orange) vs. After (blue) Distributions of Examples 1-4 partial overlap in Example 4

1.3 Developing Theory

To recap, we want the distance between the two distributions to reflect the proportion of mass that has been moved and how far the mass moved along the shortest path. This intuition is captured by following optimization problem:

$$\begin{aligned} \mathrm{OT}(\mathbb{P}_{\text{before}}, \mathbb{P}_{\text{after}}) = \underset{\gamma \in \Gamma}{\text{minimize}} \quad & \sum_{x_0 \in \mathcal{X}} \sum_{x_1 \in \mathcal{X}} c(x_0, x_1)\gamma(x_0, x_1) \\ \text{subject to} \quad & \sum_{x_1 \in \mathcal{X}} \gamma(x_0, x_1) = \mathbb{P}_{\text{before}}(x_0), \\ & \sum_{x_0 \in \mathcal{X}} \gamma(x_0, x_1) = \mathbb{P}_{\text{after}}(x_1) \end{aligned} \tag{1}$$

There are a lot of new symbols here, so let's break it down. We have already discussed some of these objects in the previous section. The "before" and "after" distributions are denoted by $\mathbb{P}_{\text{before}}$ and $\mathbb{P}_{\text{after}}$, respectively. The set of unique car prices in our "before" and "after" periods is $\mathcal{X}$. More formally, $\mathcal{X}$ is the union of the supports of $\mathbb{P}_{\text{before}}$ and $\mathbb{P}_{\text{after}}$.

The other objects in (1) are best explained by making an analogy to finding the shortest path between two points. Think of Γ as the set of all paths between any two points. We only want to consider the paths between two specific points, namely $\mathbb{P}_{\text{before}}$ and $\mathbb{P}_{\text{after}}$. This is the purpose of our two constraints: we limit ourselves to $\gamma \in \Gamma$ that connect $\mathbb{P}_{\text{before}}$ and $\mathbb{P}_{\text{after}}$.

A path between two points is familiar to us, but what exactly is a "path" between $\mathbb{P}_{\text{before}}$ and $\mathbb{P}_{\text{after}}$? Here, a path describes the proportion of mass that travels between any x_0 in $\mathbb{P}_{\text{before}}$ and any x_1 in $\mathbb{P}_{\text{after}}$. In this way, γ is the answer to question 1 from above. Minimizing on the space of Γ that satisfy our two constraints is therefore equivalent to finding the shortest path between $\mathbb{P}_{\text{before}}$ and $\mathbb{P}_{\text{after}}$. The last object we have left to define is the cost function $c(x_0,x_1)$, which is the unit cost of transporting mass between x_0 in $\mathbb{P}_{\text{before}}$ and x_1 in $\mathbb{P}_{\text{after}}$. The cost function answers question 2 from above.

The summand of our objective function describes the cost of transportation between x_0 in $\mathbb{P}_{\text{before}}$ and x_1 in $\mathbb{P}_{\text{after}}$ along the path $\gamma(x_0,x_1)$. We then sum over all possible pairs of values in $\mathcal{X}$ to get the total cost – this is what we are minimizing. Our objective function can therefore be called the *transport cost*. We are looking for the *optimal* transport cost, i.e., the smallest transport cost. There is a subtle nuance worth emphasizing: we are interested in the distance of the shortest path between distributions, not the shortest path itself. That is, we care not about the γ that solves the optimization problem but about the value of the objective at the minimizer.

The cost function provides some flexibility on how we value mass transfers between values of our support. For example, we could use the absolute value of their difference:

$$c(x_0,x_1) = |x_1 - x_0|.$$

Alternatively, if we wanted to exacerbate transfers between values that are far away on the support, we can consider the squared difference of x_0 and x_1 as the cost:

$$c(x_0,x_1) = (x_1 - x_0)^2.$$

For the exercises below, we will use the following cost function:

$$c(x_0,x_1,d) = 1(|x_1 - x_0| > d), \tag{2}$$

where d is some positive, user-specified number called the *bandwidth* and $1(|x_1 - x_0| > d)$ is 1 if the absolute difference between the values of our support is greater than d and 0 otherwise. Since the minimum value of our cost function is 0, $\text{OT}(\mathbb{P}_{\text{before}},\mathbb{P}_{\text{after}})$ must be at least 0.

When d = 0, this cost function doesn't care how far the mass had to travel; it only cares that the mass did travel. Since each mass transfer is valued with weight 1 or weight 0, we can write down the optimal cost as the percentage of mass that differs between our two distributions according to our shortest path.

As we increase d, we ignore mass transfers between nearby points and only care about mass that traveled between two points that are d units apart. The reason we may want to ignore small mass transfers will be made clear in Section 3.5.

Summary of Notation.

- $\mathbb{P}_{\text{before}}$, $\mathbb{P}_{\text{after}}$: the "points", i.e., distributions we are interested in
- Γ: all possible paths between *any* two points/distributions
- $\gamma \in \Gamma$ subject to our constraints: all possible paths between *our* two points/distributions of interest, namely $\mathbb{P}_{\text{before}}$ and $\mathbb{P}_{\text{after}}$
- $\gamma(x_0,x_1)$: the proportion of total mass we transfer from x_0 in $\mathbb{P}_{\text{before}}$ to x_1 in $\mathbb{P}_{\text{after}}$
- $c(x_0,x_1)$: cost per unit mass transferred between x_0 in $\mathbb{P}_{\text{before}}$ and x_1 in $\mathbb{P}_{\text{after}}$

1.4 Aligning Intuition and Theory

We have developed intuition in Section 1.2 and theory in Section 1.3. Let's see if they align by computing the optimal transport for our four examples using the theory above and our cost function (2). Recall that the distributions in Example 1 are the same, i.e., $\mathbb{P}_{\text{before}} = \mathbb{P}_{\text{after}}$. One possible bivariate distribution that satisfies the constraints in (1) is:

$$\gamma(x_0,x_1) = \mathbb{P}_{\text{before}}(x_0) \cdot 1(x_0 = x_1).$$

On the one hand, when $x_0 \neq x_1$, we have that $\gamma(x_0,x_1) = 0$. On the other hand, when $x_0 = x_1$, we have that $c(x_0,x_1,d) = 0$ for any value of d. Our objective function in (1) will therefore be 0, which is the smallest possible value for the optimal transport given our choice of cost function. Thus, $\text{OT}(\mathbb{P}_{\text{before}},\mathbb{P}_{\text{after}})$ in the case of Example 1 is 0%. Does this match with our intuition?

Let's move on to Examples 2 and 3. As in the previous example, we begin by discussing valid choices of γ before discussing the cost function. We have established that everything from our "before" distributions moved to the "after" distributions. Since $\mathcal{X}$ only has two values, there is only one path γ in Γ that satisfies the contraints in (1). For Example 2, we have that

$$\gamma(x_0, x_1) = \begin{cases} 8, & \text{for } x_0 = 1 \text{ and } x_1 = 2, \\ 0, & \text{otherwise.} \end{cases}$$

Put differently, $\gamma(x_0,x_1)$ contributes to the summation in the objective function of (1) only when $x_0 = 1$ and $x_1 = 2$. This means we only need to look at the our cost function when $x_0 = 1$ and $x_1 = 2$.
Our cost function is as follows:

$$c(1,2,d) = \begin{cases} 1, & \text{if } d < 1, \\ 0, & \text{otherwise.} \end{cases}$$

Putting all this together, we can solve (1) for Example 2:

$$\text{OT}(\mathbb{P}_{\text{before}}, \mathbb{P}_{\text{after}}) = \begin{cases} 100\%, & \text{if } d < 1, \\ 0\%, & \text{otherwise.} \end{cases}$$

Example 3 follows a very similar rationale. Verify for yourself that when d < 9, the optimal transport cost is 100% and 0 otherwise. When d is between 1 and 9, the optimal transport cost in Example 2 is less than the optimal transport cost in Example 3. Again, the theory and intuition line up.

Example 4 is more interesting. For simplicity, let d = 0 so that we only care about mass that travels. One solution to (1) is as follows:

$$\gamma = \frac{1}{4}\begin{pmatrix} 1/4 & 1/2 \\ 0 & 1/4 \end{pmatrix}.$$

As a matrix, each row and each column of γ denotes a value of $\mathcal{X}$; each entry in the matrix represents the mass transferred from the corresponding row's value of $\mathcal{X}$ corresponding to the corresponding column's value of the $\mathcal{X}$. For example, the proportion of mass that does not move is 50% (i.e., the sum of the diagonals). Meanwhile, the other 50% of the mass travels from \$1 to \$2 (see top-right). As a result, the value of our objective function with our choices for the cost function and γ is 50%.

There may be other choices for γ that satisfy the constraints of (1). However, we are interested not in γ but in OT($\mathbb{P}_{\text{before}}$,$\mathbb{P}_{\text{after}}$), which is unique. Try thinking of other choices of γ and see if you can convince yourself that our choice of γ above is the one that minimizes the transport cost.

2 Setting Up Python

2.1 Installing diftrans

With the basic theory out of the way, you may be itching to see how we can easily solve optimal transport problems. The R package called diftrans will come to our aid (Pouliot et al., 2021). However, to use this in Python, we will use a wrapper function that will enable us to easily integrate any R package in Python.

You will first need to install diftrans using RStudio. To install diftrans, you will also need to install the devtools package in R. You can install both packages with the following code in RStudio:

```
install.packages("devtools")

devtools::install_github("omkarakatta/diftrans")
```

Once you install the packages, run the following to verify if diftrans has been installed:

```
library(diftrans)
```

To import this package in Python (Jupyter notebook), first make sure to install the r-wrapper library using the following command:

```
pip install r_wrapper
```

And the import the diftrans package using the command:

```
from r_wrapper import diftrans
```

2.2 Installing Other Packages

Apart from diftrans, there are other packages that will be useful for cleaning and visualizing the data. Let's go ahead and import these packages as well.

```
Import pandas as pd        # for manipulating data

Import numpy as np         # for numeric operations

from matplotlib import pyplot as plt    # for visualization

from r_wrapper import base

from r_wrapper import stats
```

3 Black Market for License Plates in Beijing (Daljord et al., 2021)

3.1 Introduction

Only individuals with Beijing license plates are allowed to drive within city limits. Since January 2011, the Beijing government would give these license plates by way of a lottery to limit traffic and pollution in the city. Individuals who want to buy a new car must then apply to the lottery to obtain a valid license plate. Those awarded the license plate would comprise a random sample of these applicants. As a result, lottery winners are a random subset of car buyers. Even though the population of car buyers in 2011 is smaller than the population of car buyers in 2010, the random allocation of license plates makes it so that the distribution of car sales in 2010 and 2011 are not that different from one another. In Section 3.4, we will see that this is not the case; instead, we see that more expensive cars are being sold. Why do we see this shift in the distribution of car sales?

Daljord et al. (2021) explores various explanations, but ultimately comes to the conclusion that there is a black market for license plates. Since the license plates were quite valuable, lottery winners may want to sell the license plates to wealthy individuals than keep the license plates for themselves. The emerging black market for license plates would therefore mean that a larger share of wealthy individuals would obtain license plates. Since wealthy individuals tend to buy more expensive cars, a larger share of cars would be sold at higher prices, thereby explaining the upward shift in the distribution of car sales.

In order to address the issue of a black market, policymakers want to know how rampant these illicit transactions are. Unfortunately, a defining aspect of any black market is that their transactions take place under the radar; after all, no one would freely admit to buying and/or selling license plates illegally. Your task, as an aspiring machine learning expert with an interest in public policy, is help these policymakers by overcoming this challenge and by providing a credible lower bound on the size of the black market for car licenses in Beijing. This homework will guide you through the analysis of Daljord et al. (2021) using the accompanying package called diftrans to fulfill this task.

3.2 Understanding the Data

Upon attaching diftrans, you will have access to the datasets called Beijing_sample and Tianjin_sample. The relevant information in both these data sets is contained in the following columns:

- year: an integer denoting the month and year
- MSRP: manufacturers' suggested retail price, i.e., the price of the car in renminbi (RMB)[1]
- sales: the total number of cars sold

[1] The MSRP is not exactly the same as the transaction price, but we will assume it is a good proxy.

For more information on these datasets, you can look at their documentation. The code below will fetch the datasets:

```
Beijing_sample = base.get("Beijing_sample")

Tianjin_sample = base.get("Tianjin_sample")
```

3.3 Clean Data of Beijing and Tianjin Car Sales

Our first task will be to clean the datasets before feeding them to diftrans as inputs. We want to rewrite the data so they are in the same form as Table 1. The first column will be the unique MSRP values for all the years of interest (i.e., 2010 and 2011). The second column will be the total number of cars sold at each price level for that year. Here is how we clean the data for Beijing car sales in 2010:

```
# keep 2010 and 2011 data only
Beijing = Beijing_sample[(Beijing_sample['year']>= 2010) &
(Beijing_sample['year'] < 2012)]

# collect unique MSRP values
uniqueMSRP =
pd.DataFrame(Beijing.MSRP.unique()).rename(columns={0:'MSRP'})

# aggregate sales at each price for 2010 (pre-lottery)Beijing10_sales =
Beijing[(Beijing['year'] ==
2010)].groupby('MSRP').aggregate({'sales':[sum]})

Beijing10_sales =
Beijing10_sales.unstack().reset_index().rename_axis(None, axis=1)

Beijing10_sales = Beijing10_sales.drop(columns=['level_0',
'level_1']).rename(columns={0:'count'})

# merge the MSRP and sales
Beijing_pre = uniqueMSRP.merge(Beijing10_sales, how='left', on ="MSRP")

Beijing_pre[['count']] = Beijing_pre[['count']].fillna(value=0)

Beijing_pre = Beijing_pre.sort_values('MSRP') Beijing_pre.head()  #

preview data

df2 = Beijing_pre.pop('count') # uncount

Beijing_distribution_pre =
pd.DataFrame(Beijing_pre.values.repeat(df2, axis=0),
columns=Beijing_pre.columns)

df3 = Beijing_post.pop('count')
Beijing_distribution_post =
pd.DataFrame(Beijing_post.values.repeat(df3, axis=0),
columns=Beijing_post.columns)
```

Exercise 3.1. For each of the following, ensure that the first column is MSRP and the second column is count.

a. Clean data of Beijing car sales in 2011, and store the data frame in a variable called Beijing_post.
b. Clean data of Tianjin car sales in 2010 as a variable called Tianjin_pre.
c. Clean data of Tianjin car sales in 2011 as a variable called Tianjin_post.

3.4 Visualize Beijing Car Sales

As stated in the introduction, there is an upward shift in distribution of Beijing car sales from 2010 and 2011. Let's verify this statement by plotting the two distributions as histograms.

```
import seaborn as sns
fig, ax = plt.subplots()
for a in [Beijing_distribution_pre, Beijing_distribution_post]:
    sns.distplot(a/1000, ax=ax, kde=False)
plt.xlabel("MSRP(1000RMB)", size=14)
plt.ylabel("Density", size=14)
plt.title("Pre-lottery (blue) vs. Post-lottery (brown)\n Sales Distributions of Beijing Cars", size=18)
plt.legend(loc='upper right')
```

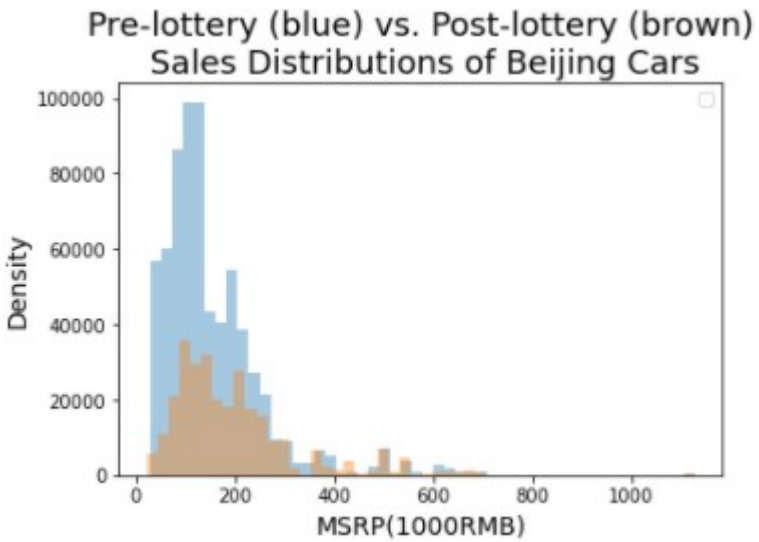


Figure 1: Pre-lottery (orange) vs. Post-lottery (blue) Sales Distributions of Beijing Cars

For ease of comparison, we overlay the two distributions on top of one another. We also normalize the histograms so the area of their respective bars sum to 1. Hence, the height of each bar tells us the proportion of cars whose prices fall in the respective bin. The post-lottery distribution is shifted to the right relative to the pre-lottery distribution, confirming the statements made in Section 3.1. Does this observation suggest that there is a black market for license plates? Perhaps this shift is not so unusual and would have happened regardless of the lottery's inception. To evaluate this concern, we may want to compare this shift with that of another city, say Tianjin, where there was no lottery during the same time period.

Exercise 3.2. The goal of this exercise is to replicate Figure 1 for Tianjin.

a. Overlay the histograms that describe the 2010 and 2011 distribution of Tianjin car sales. Be sure to normalize the histograms so the area of the bars in each histogram sum to 1.
b. Compare and contrast the shift between the Beijing distributions with the shift between the Tianjin distributions. Based on the shift in Tianjin car sales, should we be surprised to see the shift in Beijing car sales?

3.5 Compute Before-and-After Estimator

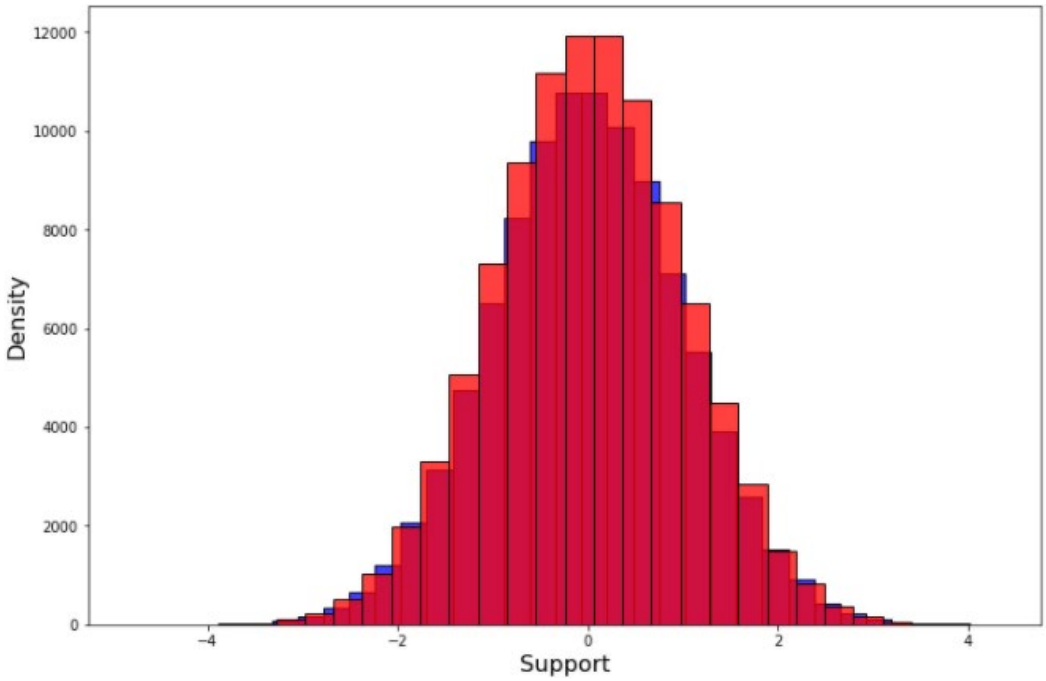
As the name suggests, our dataset, Beijing_sample, is merely a sample from the population. Hence, some or perhaps all of the differences in the empirical distributions that we plotted in Figure 1 may be a result of sampling variation. To elaborate, suppose that the true distributions of Beijing car sales in 2010 and 2011 were exactly the same. This means that the true transport cost should be 0%. However, we do not observe the true distributions – we only have access to the samples from this distribution. As a result, the distribution of our 2010 sample may differ from the distribution of our 2011 sample due to sampling variation. The optimal transport between the observed distributions would then be greater than 0%.

We will shortly discuss why this is of concern to us, but let's first see that two samples drawn from the same distribution indeed have a nonzero transport cost. Let's draw two samples from a standard normal distribution and plot their histograms in Figure 3.

```
n_observations = 100000
sample1 = np.random.normal(0, 1, n_observations)
sample2 = np.random.normal(0, 1, n_observations)
import seaborn as sns
import matplotlib.pyplot as plt


plt.figure(figsize=(12,8))
sns.histplot(data = sample1, bins = 30, color = 'blue')
sns.histplot(data = sample2, bins = 30, color = 'red')
```

```
plt.xlabel("Support", size=16)
plt.ylabel("Density", size=16)
```



The histograms of our two samples are quite similar to one another. After all, we constructed our samples so they are drawn from the same distribution. But even without seeing the code that generates the two samples, you could have guessed that they were drawn from the same distribution. As similar as these sample distributions are, they are not exactly the same. (Notice the flecks of orange peeking out.) As a result, the optimal transport cost will be nonzero.

Before we compute the optimal transport cost, let's repeat the procedure above with our distributions rather than a standard normal distribution. Let our first placebo distribution $\mathbb{P}^{(1)}_{\text{before, Beijing}}$ be sampled from $\mathbb{P}_{\text{before, Beijing}}$ with the same number of observations as $\mathbb{P}_{\text{before, Beijing}}$. Let our second placebo distribution $\mathbb{P}^{(2)}_{\text{before, Beijing}}$ also be sampled from $\mathbb{P}_{\text{before, Beijing}}$ but with the same number of observations as $\mathbb{P}_{\text{after, Beijing}}$. While $\mathbb{P}_{(1)\text{before, Beijing}}$ and $\mathbb{P}_{(2)\text{before, Beijing}}$ are sampled from the same distribution, note that their sample sizes are different! Below is the code to construct $\mathbb{P}^{(1)}_{\text{before, Beijing}}$.

```
# set the seed for reproducibility
base.set_seed(1)
# We will use the `rmultinom` function to construct our placebo.
```

```
  # Imagine the same number of cars as in 2010. (see `size` argument)

  # For each MSRP value, we will decide how many of these imaginary cars
will

  # be sold at this price. The number of of these imaginary cars to be
sold at

  # the particular MSRP value will be proportional to the actual number
of cars

  # sold in the pre-lottery distribution. (see `prob` argument) # We
only want one placebo distribution. (see `n` argument) placebo_1 <-
data.frame(MSRP = Beijing_pre['MSRP'],

  count = stats.rmultinom(n = 1, size = sum(Beijing_pre['count']),
prob = Beijing_pre['count'])

  count2 = count[:,0]

  d = {'MSRP': Beijing_pre['MSRP'], 'count' : count2}

  placebo_1 = pd.DataFrame(data=d)

  print(placebo_1)

  print(placebo_1.dtypes)
```

```
        MSRP  count
482    20800      0
374    29800     50
338    32900   3136
227    33800   3597
388    34800    539
..       ...    ...
263   703600    350
7     770000    101
314   800400      2
433   998000      6
5    1127800    325

[513 rows x 2 columns]
MSRP     int64
count    int32
dtype: object
```

Exercise 3.3.

a. Run the preceding code block so you have access to `placebo_1`.
b. Use `rmultinom` to sample observations from `Beijing_pre`. Store the resulting data frame in `placebo_2`. Be careful to draw the correct number of observations.
c. Compare `placebo_1` and `placebo_2`. Do they appear to be drawn from the same distribution?

Now that we have two placebo distributions, let us compute their optimal transport cost. If we want to compute the optimal transport between these two placebo distributions with bandwidth $d = 0$, I could use `diftrans`:

```
  placebo_at_0 = diftrans.diftrans(pre_main = placebo_1,
post_main = placebo_2, bandwidth_seq = 0)

  # The transport cost for the specified bandwidths have been
computed

  print(placebo_at_0 )
```

```
   bandwidth      main
1          0  0.011554
```

According to diftrans, the optimal transport cost between our two placebo distributions is 1%, even though both of these distributions are drawn from the same distribution![2] In other words, diftrans is picking up differences due to sampling variation as a legitimate shift in the two distributions.

Why do we care about this? Well, we do not want to confuse differences that arise from sampling variation as evidence for black market transactions. Otherwise, we will inflate the size of the black market. This is where the choice of our bandwidth d in our cost function comes in handy; recall that we can ignore shifts between the distribution that are less than d units apart. So, let's pick a bandwidth that is large enough to filter out shifts in our empirical distributions that arise from sampling bias. But we need to be careful: too large a bandwidth might mean we filter out shifts that arise due to the presence of a black market.

To figure out the appropriate bandwidth, we will adopt this rule of thumb: choose the smallest d such that the optimal transport between two placebo distributions, $\mathbb{P}^{(1)}$ and $\mathbb{P}^{(2)}$, is less than 0.05%, where $\mathbb{P}^{(1)}$ and $\mathbb{P}^{(2)}$ are drawn from the same distribution. We can then compute the optimal transport between our actual

[2] Since the placebo distributions are random draws, the optimal transport cost that you might get if you run the same code may be slightly different from our estimate. Nonetheless, it should be quite small. [3]Think about why this is a lower bound – we will go over this in Section 4.

distributions using the selected bandwidth. The procedure that we just described gives us the *before-and-after estimator* and allows us to overcome bias due to sampling.

Exercise 3.4.

a. Compute the transport cost between the two placebo distributions for different values of d from 0 to 100,000.
b. For the same values of d, compute the transport cost between the observed distributions for 2010 and 2011 Beijing car sales.
c. Plot the placebo costs and the empirical costs obtained in the previous two steps with the bandwidth as the x-axis.
d. For which values of d is the placebo cost less than 0.05%?
e. For the smallest value of d found in the previous step, what is the empirical transport cost? This estimate for the lower bound on the volume of black market transactions is what we call the *before-and-after estimate*.[3]

3.6 Compute Differences-in-Transports Estimator

The before-and-after estimator looks at the change in the distribution of Beijing car sales before and after the lottery, and takes into account that the data we work with is sampled from the population. As powerful as this estimator is, it relies on a rather strong assumption that there are no time trends. That is, the before-and-after estimator assumes that in the absense of the lottery and any black market transactions, the distribution of car sales is the same in 2011 as it is in 2010. If this assumption ends up being false, then the before-and-after estimator inflates the volume of black market trades and may not be a lower bound for the actual volume of the black market.

To relax this assumption, we would first have to measure how much of the shift is due to time trends; then, we can subtract this from the shift in Beijing car distributions. The estimator that results from this procedure is the *differences-in-transports* estimator. See if you can figure out why this is an aptly named estimator as we provide more details.

To measure the shift that results from time trends, we can use the optimal transport cost between the 2010 and 2011 distributions of Tianjin car sales, denoted $\mathbb{P}_{\text{before, Tianjin}}$ and $\mathbb{P}_{\text{after, Tianjin}}$, respectively. Why would this be a good measure of time trends? Unlike Beijing, Tianjin did not have a lottery or adopt new policies that would affect the distribution of their car sales. We can therefore argue that the changes in Tianjin's distribution of car sales from 2010 to 2011 is due to time trends. The more formal argument is that changes in Tianjin's distribution is the same as changes in Beijing's distribution if Beijing did not have a lottery or black market.

We have two tranport costs to use: one for Beijing and another for Tianjin. Rather than use diftrans twice to compute each of these transport costs, we can use diftrans once by taking advantage of its other arguments. For example, suppose we are interested in the difference-in-transports estimator with a bandwidth of 0:

```
dit_at_0 = diftrans.diftrans(pre_main = Beijing_pre,
                             post_main = Beijing_post,
                             pre_control = Tianjin_pre,
                             post_control = Tianjin_post,
                             bandwidth_seq = 0,
                             conservative = True)
print(dit_at_0)
```

```
   bandwidth      main    main2d   control      diff    diff2d
1          0  0.353123  0.353123  0.298681  0.054443  0.054443
```

You will notice that the output is different than the one for the before-and-after estimator:

- main is the Beijing transport cost
- control is the Tianjin transport cost
- diff is the difference in the Beijing and Tianjin transport cost.

The other two columns, main_2d and diff2d, are a result of using the conservative = TRUE option in diftrans. With this option, we use twice the bandwidth when computing Beijing's transport cost:

$$\text{OT}_{2d}(\mathbb{P}_{\text{before, Beijing}}, \mathbb{P}_{\text{after, Beijing}}) - \text{OT}_{d}(\mathbb{P}_{\text{before, Tianjin}}, \mathbb{P}_{\text{after, Tianjin}}). \tag{3}$$

The first term of the difference is given by main2d and the entire difference is given by diff2d. Using conservative = TRUE ensures that we find a lower bound for the volume of black market sales. You will also use conservative = TRUE in the following exercise. Also note that you've already computed Tianjin_pre and Tianjin_post from Exercise 3.1.

The results above suggest that at least 5% of car sales is a result of the black market. However, don't consider this to be our lower bound. Our discussion about sampling variation still applies here, so choosing a bandwidth of 0 as we did above is not a good choice. The exercise below walks you through the criterion we use to choose the appropriate bandwidth.

Exercise 3.5.

a. Compute the (3) for different values of d from 0 to 50,000. Unlike before, we go up to 50,000 because we are using the conservative bandwidth of $2d$ for the Beijing transport cost.
b. Using what you learned from Exercise 3.3, construct a placebo distribution that is sampled from Beijing_pre whose size is the number of Beijing cars in 2010. Call this distribution placebo_Beijing_1.

c. Construct another placebo distribution called placebo_Beijing_2 that is also sampled from Beijing_pre but is of size is the number of Beijing cars in 2011.
d. Construct a placebo distribution called placebo_Tianjin_1 that is sampled from Tianjin_pre and whose size is the number of Tianjin cars in 2010.
e. Construct a placebo distribution called placebo_Tianjin_2 that is sampled from Tianjin_pre and whose size is the number of Tianjin cars in 2011.
f. Using the four placebo distributions, compute the placebo counterpart of (3) for the same values of d that you used in part a.
g. Create a plot of the *absolute value* of the placebo differences-in-transports estimator on the y-axis and the bandwidth on the x-axis.
h. For which values of d does the absolute value of the placebo differences-in-transports estimator stay below 0.05%? Note that the absolute difference is not a monotonically decreasing object, so this difference may even increase as we increase the bandwidth. Temporary increases above the 0.05% threshold can be ignored.
i. Among all the values of d that you found in the previous step, which one yielded the largest value of (3) from part a? This is the difference-in-transports estimator.

4 Concluding Remarks

In the matter of a few exercises, you have computed the lower bound on the size of the black market for Beijing license plates using the before-and-after estimator and the differences-in-transports estimator. The black market, by its very nature, cannot be seen or measured directly, so let's review: how did you make an informed statement about something that is invisible?

You compared what you *should* have seen following a policy change in 2011 with what you *actually* saw. You should have seen no change in the distribution of car sales between 2010 and 2011 because those that won the license plates comprise a random subset of the population of car buyers. Instead, you saw a large upward shift in the distribution of car sales. In other words, a much larger share of people bought more expensive cars in 2011 than in 2010. This difference can be explained by the black market. Buyers in the black market for license plates purchase more expensive cars than the winners of the lottery. This difference in consumption behavior as evidenced by the change in the distribution of car prices tells you something about the size of the black market. So, you rephrased the question about something unobservable in terms of something that is observable: what is the difference in the distribution of car sales between 2010 and 2011?

4.1 Lower Bound

Why does the answer to the rephrased question give you a *lower bound* on the size of the black market? Well, our estimators use the difference in consumption behavior between buyers in the black market for

license plates and lottery winners as a signal for the black market: buyers in the black market purchase more expensive cars. If buyers on the black-market purchase cars that are the same price as cars that would have been bought by the winners of the lottery, then we would not be able to identify such a transaction as being illicit. So even if the 2010 and 2011 distribution of car sales were in fact the same, this does not rule out the possibility that all the sales were a result of the black market for license plates. Hence, we can only provide a lower bound on the size of the black market.

4.2 Simulations

In this homework, you used only one set of placebo distributions to determine the appropriate bandwidth for each estimator. In practice, you would use many more placebo distributions and look at the mean, standard deviation, and quantiles of the placebo costs to choose the appropriate bandwidth. Daljord et al. (2021) uses upwards of 500 simulations for each placebo distribution to choose the bandwidth. If you have some time, try repeating placebo calculations with more simulations and see how much your choice of bandwidth changes.

Once you have chosen the appropriate bandwidth, you can also create subsamples or bootstrap samples of your data, and compute the before-and-after estimator and/or the differences-in-transports estimator for each sample. Doing so will give you a distribution of estimates at the appropriate bandwidth, which will further enable you to compute standard errors for your estimate.

4.3 Robustness Checks and Verifying Assumptions

We briefly discussed the importance of the parallel-trends assumption when motivating the differences-in-transports estimator. Daljord et al. (2021) empirically justifies that this assumption holds by comparing the changes in car sales distribution between Tianjin and Beijing during 2014 and 2015. During these two years, both Beijing and Tianjin had policies in place to limit traffic congestion and pollution. While Beijing used a lottery, Tianjin used a hybrid auction and lottery. Since both cities were roughly in the same regime, we can see whether the change in the distribution at every level of the bandwidth were similar to one another, and it turns out they are quite close. In fact, we can choose a bandwidth large enough to filter out time-varying trends.

Aside from verifying assumptions, Daljord et al. (2021) performs robustness checks. For example, Daljord et al. (2021) repeats the analysis after disregarding data from December 2010. The reason for disregarding December 2010 data is that people might have changed their behaviour at the end of 2010 in anticipation of the lottery in the beginning of 2011. However, removing December 2010 has very little impact on our analysis, suggesting that anticipation of the lottery is not a threat to our results.

4.4 diftrans package

The package that you've used in this homework is quite powerful and will be useful when you use the before-and-after estimator and the differences-in-transports estimator for your own future work. However, *diftrans* is an evolving package. We hope to improve its functionality and user interface in the coming months, so be aware that the syntax may change slightly in the future.

Optimal transport theory coupled with *diftrans* provides a powerful way of analyzing changes in distribution, and Daljord et al. (2021) introduces mechanically and intuitively simple estimators that allow practitioners to do more with the data we have.

References

Daljord, Ø., Pouliot, G., Hu, M., & Xiao, J. (2021). *The black market for beijing license plates*.

Pouliot, G., Daljord, O., & Katta, O. A. (2021). *Diftrans: Compute the differences-in-transports estimator*. https://github.com/omkarakatta/diftrans