

# Nearest Neighbour Matching (NN) followed by a Linear Model with Difference in Differences (DD)

Michal Lapinski  
JKU Linz

---

## Abstract

The `nndd` package (<https://R-Forge.R-project.org/projects/uibk-rprog-2017/>) estimates the average treatment effect by applying nearest neighbour matching (NN) and difference in differences (DD). The nearest neighbours are matched by estimating a GLM over an individual time span. In the following a linear model is estimated with a difference in differences setup. Each estimation (NN or DD) can depend on different covariates. Simple evaluation methods of the combined estimation are provided.

*Keywords:* nearest neighbour, matching, regression, difference in differences, R.

---

## 1. Introduction

The `nndd` package can be used to estimate causal effects if a selection into treatment is observed. The background of this method is that the treatment effect wants to be estimated by difference in differences (DD). In general DD leads only to unbiased and causal impacts if the treatment assignment is random and all other identifying assumptions hold (see [Angrist and Pischke \(2008\)](#) for more details). Nevertheless, in many cases random assignment didn't occur, wasn't possible or reasonable. In natural-experiments (quasi-experiments) it is possible to observe exogenous assignments and estimate the treatment effect. However, often the assignment is not truly exogenous. In this case there are some possibilities to overcome this selection. One is to control for observed characteristic in the analysis, another is to find an instrument (IV). A third one is to construct treatment and control groups such that they are as similar as possible in the observed characteristics (matching). All three methods are not the perfect solution and can be biased due to omitted variables, influencing the treatment assignment or invalid identifying assumptions. However, the research tends to the conclusion that matching can lead to smaller bias than just controlling for observed characteristics. ([Rubin 1973](#); [Angrist and Pischke 2008](#); [Caliendo and Kopeinig 2008](#); [Imbens and Rubin 2015](#); [Huber, , Lechner, and Steinmayr 2015](#))

Summing up ([Imbens and Rubin 2015](#), 401) point out:

*"[...] in many observational studies there exists no systematically better approach for estimating the effect of a treatment on an individual unit than by finding a control unit identical on all observable aspects except on the treatment received and then comparing their outcomes."*

It is to mention that this package performs only 1:1 nearest neighbour matching with re-

placement and without truncation (NN) which is a very straight forward method, but other matching methods mostly outperform NN (see [Huber et al. \(2015\)](#)).

### Syntax of Nearest Neighbour matching (NN) in Short

At first a generalized linear model (GLM) is estimated with the treatment status ( $t$ ) as the dependent variable which is regressed on the independent variables ( $z$ ). Where  $z$  are observed variables being expected to influence the treatment status and not influencing the outcome variable of the treatment. In the next step the propensity scores are predicted for the treatment and control groups. In the following to each treated observation  $tg_i$  only one control observation  $cg_j$  is selected. In the selection procedure the control observation  $cg_j$  is matched to  $tg_i$  if it has the smallest absolute difference in the pscore among all control observation to the treated observation  $tg_i$ .

## 2. Implementation

As usual in many other regression packages for R ([R Core Team 2017](#)), the main model fitting function `nndd()` uses a formula-based interface and returns an (S3) object of class `nndd`:

```
nndd(formula, data, indexes = c("year", "firm_id", "tg", "outcome"),
      t_time, nn_time, time_ids = c("year", ""),
      link = "logit",
      subset , na.action, clustervariables,
      model = TRUE, y = TRUE, x = FALSE, displ_coefs,
      ...)
```

Actually, the `formula` has to be a two part Formula ([Zeileis and Croissant 2010](#)), specifying separate sets of outcomes ( $tg_i$ ,  $outcome_i$ ) as well as regressors  $x_i$  and  $z_i$ . For instance the formula can take a form of `tg | outcome ~ x | z` where `tg` is the response and `z` regressor variable of the GLM. The variable `outcome` is the response and `x` the control variable of the DD model. The `data` argument specifies a data frame which contains the variables occurring in the `formula` such as time and group identifiers. In addition the data has to be a panel. The argument `indexes` is a list of the group and time identifier. Last but not least `t_time` has to be specified, which defines the time of the treatment. The other arguments can be looked up in the help page of `nndd`.

A number of standard S3 methods are provided, see [Table 1](#).

Due to these methods a number of useful utilities work automatically, e.g., `AIC()`, `BIC()`, `coefstest()` (`lmtest`), `waldtest()` (`tttest`), `mtable()` (`memisc`), etc.

In addition two `summary()` S3 methods are provided. One for the class `lm` and another for the class `lmc`. Where the class `lmc` is a child of (inherits) class `lm` and implements an additional variable `clustervariables`. However, there is not construction function supportet to create a class `lmc` object yet.

## 3. Illustration

To illustrate the package's use in practice, a usual difference in difference methodology is

Method	Description
<code>print()</code>	Simple printed display with coefficients
<code>summary()</code>	A regression summary which can perform clustered standard errors; returns <code>summary.nnndd</code> object (with <code>print()</code> method)
<code>coef()</code>	Extract coefficients
<code>vcov()</code>	Associated covariance matrix
<code>predict()</code>	Different types of predictions (pscore or outcome) for new data
<code>ttest()</code>	Performs a ttest for matched treated and controls
<code>plot()</code>	Creates support plots for the NN and <code>lm.plot</code> methods for the DD estimation.
<code>waldtest()</code>	Performs the wldtest

Table 1: S3 methods provided in `nndd`.

compared to the combined methodology of `nndd`. Therefore, data on the Evaluation of the Immigration Reform and Control Act (IRCA) is used. The data is adapted data of (?).

The author used the original data to examine the effects of the Immigration Reform and Control Act on crime. The IRCA was implemented in 1986 and forbid to hire or recruit undocumented immigrants. However the IRCA also implemented a near-universal legalization of immigrants in the United States.

The theory behind a positive impact of the IRCA on crime is that an increased labour market opportunity due to IRCA increases legal work and decreases crime. The labour market opportunity is expected to increase because legal (documented) immigrants have a higher salary and lower chance to be fired. In the following crime decreases due to the increased employment.

The data consists of 31.206 observations on 21 variables. In detail it is a balanced data panel of 1.486 US counties over 21 years (the time span is 1980 till 2000). In this illustration we use some of the available variables. The chose variables are chosen with some care, however other variables might be also relevant and could improve the results. For a description of the variables and more detailed information of the data see the help page of the IRCA data or Baker (2015).

At first we create a `nndd` object. We use `year` and `county` as time and individual identifiers. `treated` is defined as the treatment variable and `v_crime` (violent crimes) as the outcome. The treatment timing is set as the year 1986. As no `nn_time` is supported, the matching occurs only on the observed values one period before treatment. Last but not least we define not to display the state fixed effect in summary statistics.

```
R> library("nndd")
R> data("IRCA", package = "nndd")
R> IRCA$StateFIPS <- factor(IRCA$StateFIPS)
R> formula <- Formula(treated | v_crime ~ unemprate + povrate + pop
+
+               + crack_index + officers_pc + income + abortions + StateFIPS
+               | unemprate + povrate + pop + crack_index +officers_pc   )
R> nndd1 <- nndd(formula = formula, data = IRCA,
+
+               index = c("county", "year"),
```

Nearest Neighbour Matching (NN) followed by a Linear Model with Difference in Differences (DD)

```
+           t_time = "1986",
+           displ_coefs = c("unemprate", "povrate", "pop" , "crack_index",
+                           "officers_pc", "income" , "abortions", "post",
+                           "treated", "post:treated") )
R> print(mnodd1)
```

Nearest Neighbour Matching (NN) followed by a Linear Model with Difference in Differences

DD was computed as follows

```
The id variable was:                county
The time variable was:              year
The outcome variable was:          v_crime
The variable identifying the treatment group was: treated
The variable categorizing the pre and post treatment period was generated as: post
```

The timing of the treatment was set as year 1986.

Coefficients in linear model (DD):

(Intercept)	unemprate	povrate	pop	crack_index
-10.324747	-0.017403	0.041067	0.052338	0.006230
officers_pc	income	abortions	StateFIPS4	StateFIPS5
70.392891	0.344500	7.967841	0.120735	-0.070991
StateFIPS6	StateFIPS8	StateFIPS9	StateFIPS13	StateFIPS15
0.080511	0.092270	-0.625594	-0.384481	-0.386130
StateFIPS16	StateFIPS17	StateFIPS18	StateFIPS19	StateFIPS22
0.252128	0.115142	-0.476853	0.016748	-0.362584
StateFIPS23	StateFIPS24	StateFIPS25	StateFIPS26	StateFIPS27
0.256896	0.334269	-0.886370	-0.336870	-0.028296
StateFIPS28	StateFIPS29	StateFIPS31	StateFIPS32	StateFIPS33
-0.167201	-0.216270	0.094629	0.306003	0.166043
StateFIPS34	StateFIPS35	StateFIPS36	StateFIPS37	StateFIPS39
0.082225	-1.398491	-0.060739	0.532583	-1.642612
StateFIPS40	StateFIPS41	StateFIPS42	StateFIPS44	StateFIPS45
-0.344928	-0.007305	-0.522948	0.320012	0.228280
StateFIPS47	StateFIPS48	StateFIPS49	StateFIPS51	StateFIPS53
-0.779264	-0.386983	-0.220564	0.371760	-0.009302
StateFIPS54	post	treated	post:treated	
0.108921	0.309161	0.351384	-0.117979	

NN was computed as follows

The time interval for Nn was:

Start time: 1985

End time: 1985

Family: binomial

Link function: logit

Summary statistics of the pscore

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Treated (1)	0.1271	0.7859	0.9826	0.8496	0.9998	1.0000
Control (0)	0.1291	0.7919	0.9927	0.8491	0.9986	0.9986

Summary statistics of the pscore difference between treated and control

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
	-0.0297402	-0.0018599	0.0011953	0.0004642	0.0014327	0.0272436

R> summary(nnidd1)

Call:

```
nnidd(formula = treated | v_crime ~ unemprate + povrate + pop +
  crack_index + officers_pc + income + abortions + StateFIPS |
  unemprate + povrate + pop + crack_index + officers_pc, data = IRCA,
  index = c("county", "year"), t_time = "1986", displ_coefs = c("unemprate",
    "povrate", "pop", "crack_index", "officers_pc", "income",
    "abortions", "post", "treated", "post:treated"), nn_time = c("1985",
    "1985"), time_ids = c("year", ""), link = "logit")
```

Residuals:

	Min	1Q	Median	3Q	Max
	-5.7818	-0.2647	0.1013	0.3684	2.6359

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
unemprate	-0.017403	0.004174	-4.169	3.08e-05 ***
povrate	0.041067	0.002311	17.771	< 2e-16 ***
pop	0.052338	0.013115	3.991	6.63e-05 ***
crack_index	0.006230	0.011838	0.526	0.59871
officers_pc	70.392891	6.999698	10.057	< 2e-16 ***
income	0.344500	0.047536	7.247	4.54e-13 ***
abortions	7.967841	5.095740	1.564	0.11793
post	0.309161	0.034894	8.860	< 2e-16 ***
treated	0.351384	0.031572	11.130	< 2e-16 ***
post:treated	-0.117979	0.031484	-3.747	0.00018 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7307 on 10955 degrees of freedom

Multiple R-squared: 0.4754, Adjusted R-squared: 0.4731

F-statistic: 206.9 on 48 and 10955 DF, p-value: < 2.2e-16

In the model nniddlc we assume that the observations are correlated within states.

Nearest Neighbour Matching (NN) followed by a Linear Model with Difference in Differences (DD)

```
R> nndd1c <- nndd(formula = formula, data = IRCA,
+                 index = c("county", "year"),
+                 t_time = "1986" ,
+                 clustervariables = "StateFIPS",
+                 displ_coefs = c("unemprate", "povrate", "pop" , "crack_index",
+                               "officers_pc", "income" , "abortions",
+                               "post", "treated", "post:treated"))
```

Next we estimate usual DD models without matching. We use all variables which were used in the nndd model as controls. We estimate again two models one normal linear regression and the other with clustered standard errors. Because there is no construction function for the class `lmc` we construct it by hand for this example. We also use the class `lmc` for the non clustered version because the summary function of class `lm` is not adapted to omit display variables.

```
R> lm1 <- lm(update(formula(formula, lhs = 2, rhs = 1),
+                  paste(paste(".",
+                              paste(formula(formula, lhs = 0, rhs = (2)),
+                                       collapse = " . + ")),
+                          "+post*treated")),
+           data = IRCA)
R> lm1$displ_coefs <- c("unemprate", "povrate", "pop" , "crack_index",
+                     "officers_pc", "income" , "abortions",
+                     "post", "treated", "post:treated")
R> class(lm1) <- c("lmc", "lm")
R> lm1c <- lm1
R> lm1c$clustervariables <- "StateFIPS"
R> class(lm1c) <- c("lmc", "lm")
```

Using a model table from **memisc** (Elff 2016) it can be easily seen, that we have different coefficients and significance across the models (see Table 2). Comparing the two models with clustered standard errors, we still see that usual DD would estimate a significant impact of IRCA on violence crime. However, nndd states a non significant impact.

```
R> mtable(lm1, nndd1, lm1c, nndd1c)
```

The difference of the sample specification is driving these results. In the nndd model we regress only on a very similar control and treatment group. We can see the similarity of the two groups in the distribution graphs of the pcores (see figure 1). Of course this only holds if pcore truly capture the selection process.

```
R> #dev.new()
R> par(mfrow = c(1,2))
R> plot(nndd1c, data = IRCA , which = c(1,2))
```

	lm1	nnddl	lm1c	nnddl1c
unemprate	0.002 (0.002)	-0.017*** (0.004)	0.002 (0.007)	-0.017 (0.015)
povrate	0.029*** (0.001)	0.041*** (0.002)	0.029*** (0.005)	0.041*** (0.010)
pop	0.109*** (0.006)	0.052*** (0.013)	0.109*** (0.023)	0.052* (0.026)
crack_index	-0.070*** (0.008)	0.006 (0.012)	-0.070 (0.041)	0.006 (0.036)
officers_pc	19.702*** (2.540)	70.393*** (7.000)	19.702 (11.611)	70.393* (29.680)
income	0.737*** (0.025)	0.345*** (0.048)	0.737*** (0.122)	0.345 (0.308)
abortions	37.700*** (3.657)	7.968 (5.096)	37.700 (24.275)	7.968 (14.217)
post	0.209*** (0.018)	0.309*** (0.035)	0.209** (0.075)	0.309** (0.098)
treated	0.159*** (0.024)	0.351*** (0.032)	0.159* (0.066)	0.351** (0.129)
post × treated	-0.216*** (0.024)	-0.118*** (0.031)	-0.216** (0.070)	-0.118 (0.262)
R-squared	0.4	0.5	0.4	0.5
adj. R-squared	0.4	0.5	0.4	0.5
sigma	0.8	0.7	0.8	0.7
F	372.4	206.9	372.4	206.9
p	0.0	0.0	0.0	0.0
Log-likelihood	-35499.2	-12137.4	-35499.2	-12137.4
Deviance	17776.6	5849.6	17776.6	5849.6
AIC	71104.3	24374.7	71104.3	24374.7
BIC	71546.8	24740.0	71546.8	24740.0
N	31206	11004	31206	11004

Table 2: Comparing results of simple DD and nnddl.

In the left graph we can see that the pscore distribution of treated (blue) and control (red) was very different before NN. Especially many control units had a pscore close to zero. After matching the distributions of the pscore look alike.

This brief illustration shows some features of the nnddl package. There are more functions such as `t.test` which evaluates the match quality of NN.

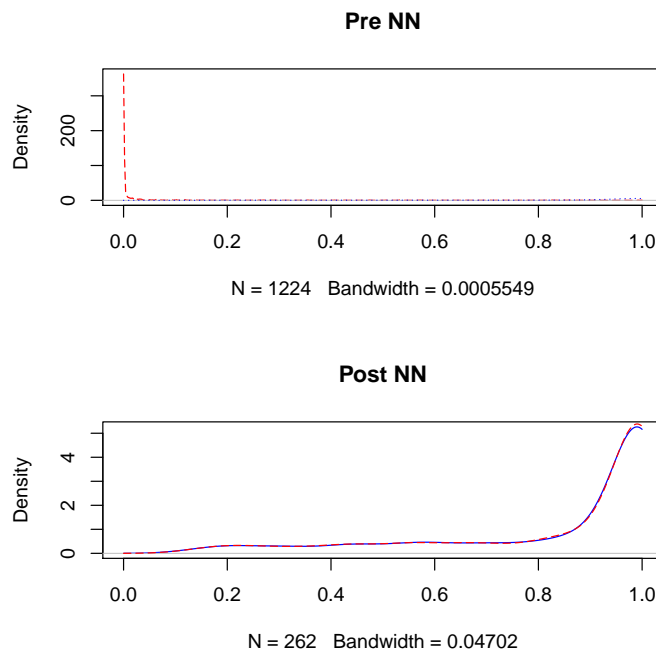


Figure 1: Pscore distribution before NN and after NN

## References

- Angrist JD, Pischke JS (2008). *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton University Press.
- Baker SR (2015). "Effects of Immigrant Legalization on Crime." *American Economic Review*, **105**(5), 210–213.
- Caliendo M, Kopeinig S (2008). "Some Practical Guidance for the Implementation of Propensity Score Matching." *Journal of Economic Surveys*, **22**(1), 31–72. ISSN 1467-6419. doi:10.1111/j.1467-6419.2007.00527.x. URL <http://dx.doi.org/10.1111/j.1467-6419.2007.00527.x>.
- Elff M (2016). *memisc: Tools for Management of Survey Data and the Presentation of Analysis Results*. R package version 0.99.8, URL <https://CRAN.R-project.org/package=memisc>.
- Huber M, , Lechner M, Steinmayr A (2015). "Radius Matching on the Propensity Score with Bias Adjustment: Tuning parameters and finite sample behaviour." *Empirical Economics*, **49**(1), 1–31. ISSN 1435-8921. doi:10.1007/s00181-014-0847-1.
- Imbens GW, Rubin DB (2015). *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. 1 edition. Cambridge University Press. ISBN 9780521885881. URL <http://amazon.com/o/ASIN/0521885884/>.



R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.

Rubin DB (1973). “Matching to Remove Bias in Observational Studies.” *Biometrics*, **29**(1), 159–183. URL <http://www.jstor.org/stable/2529684>.

Zeileis A, Croissant Y (2010). “Extended Model Formulas in R: Multiple Parts and Multiple Responses.” *Journal of Statistical Software*, **34**(1), 1–13. doi:10.18637/jss.v034.i01.

**Affiliation:**

Michal Lapinski

Department of Economics

JKU Linz

E-mail: [Michael.Lapinski@JKU.at](mailto:Michael.Lapinski@JKU.at)

URL: [www.JKU.at](http://www.JKU.at)